

# **A predictive model for the selectivity variables of an ammonium nitrate fluidised bed granulator**

**L Botha**



**[orcid.org/0000-0001-7971-1240](https://orcid.org/0000-0001-7971-1240)**

Dissertation submitted in fulfilment of the requirements for the degree *Master of Engineering in Chemical Engineering* at the North-West University

|                       |                     |
|-----------------------|---------------------|
| Supervisor:           | Mr AF van der Merwe |
| Assistant-supervisor: | Prof KR Uren        |
| Assistant-supervisor: | Prof G van Schoor   |

Graduation ceremony: May 2019

Student number: 23603844

## Acknowledgements

*“Many are the plans in a man’s heart, but it is the Lord’s purpose that prevails.” –  
Proverbs 19:21*

Firstly, I thank God for the intellectual ability, patience and strength to have been able to finish this journey He has chosen for me.

I thank Omnia Holdings LTD as our industry partner for the financial support they have given to me throughout my master’s degree. A big thank you to the research and development team for assistance and guidance as well as to the production team, for giving me the opportunity to obtain data from the granulation plant. Thank you to the quality control laboratory for the permission to conduct analytical analyses using their equipment.

I thank the North West University, Potchefstroom campus, for giving me the opportunity to complete my master’s degree. In that capacity, I thank the following people for their guidance and assistance:

- Mr Frikkie van der Merwe as my study leader, you have guided me to be a better chemical engineer as well as researcher and I will always be thankful for that
- Prof Kenny Uren, for your mentorship throughout my study
- Prof George van Schoor, for your mentorship throughout my study
- Tammarin Oelofse for assisting in the validation of selectivity variables
- Zandré Broodryk for assisting in sampling for data acquisition

Last, but not least, I thank my family and friends for the love and support they have given to me in this journey. You have been my pillar of strength when I needed it most.

## Abstract

A model for the inference measurement of the selectivity variables of an ammonium nitrate fluidised bed granulator was investigated. Fluidised bed granulators are complex due to numerous processes occurring in one vessel and can thus be challenging to control. Previous studies have shown that there is no statistical significant relation between the input (operational) parameters and the selectivity (quality) variables of the fluidised bed granulator, therefore the use of artificial neural networks to estimate the selectivity variables was investigated.

The objectives of the study included confirming that the selectivity variables are measured adequately and that the results of the measurements can be trusted as well as the evaluation of artificial neural network configurations as predictive models for the selectivity variables.

The selectivity variables investigated are porosity and sphericity of the ammonium nitrate product granules. The porosity measurement used by the industry partner, namely the oil absorption technique, was compared to a newly developed technique called the optical porosity measurement technique to ensure the results are credible and repeatable. The correlation between the two techniques was satisfactory, with a coefficient of determination of 0.92 and a mean absolute error of 12.7 %. The high absolute error was due to the difference in the calculations of the two techniques; the oil absorption technique calculates the porosity of the particle based on a mass of oil absorbed whereas the optical porosity measurement technique calculates porosity based on the volume of the pores. The sphericity measurement technique was tested for repeatability of the technique and it was found that the results did not vary significantly when samples were analysed multiple times.

The output values for the neural networks were the two selectivity variables investigated in the study. The prediction of the variables were done using separate neural networks. The input values for the neural networks were chosen as operating parameters that were measured online. These values are listed below, as used for phase 1 / phase 2 respectively.

- Fluidising air flow rate / fluidising air pressure
- Fluidising air temperature / fluidised bed temperature
- Liquid ammonium nitrate density / liquid ammonium nitrate concentration
- Liquid ammonium nitrate temperature
- Liquid ammonium nitrate flow rate / liquid ammonium nitrate pressure
- Atomising air flow rate

The first part of the evaluation of artificial neural networks consisted of using historical plant data that had to undergo multiple filtering stages before the data could be used for the training of a neural network. In this phase, different neuron configurations were tested using two training algorithms, namely Levenberg-Marquardt and Bayesian regularisation, on three datasets that were each assembled in their own way. The first dataset consisted of all the usable data acquired from the historical data, the second set contained only discrete data points with unique output values, while the third dataset contained data in constant time intervals of 10 minutes after each new output value was recorded. The efficiency of the constant-time dataset for use in an artificial neural network was also assessed for different time intervals as well as noise filtering of the input data. The best performing network in this phase was trained with the noise filtered constant-time data with a 10-minute time interval, using the Bayesian regularisation training algorithm with 100 neurons in the hidden layer.

Data acquisition for the second phase of this study was based on the conclusions drawn from the first phase, which include:

- Taking of samples for porosity and sphericity measurement in 10-minute intervals
- Filter input data to reduce sensor noise
- Use a large number of neurons in the hidden layer
- Use the Bayesian regularisation training algorithm

The best performing neural network for the prediction of porosity, used the Bayesian regularisation training algorithm for the training of the network and had 61 neurons in the hidden layer. The training of the neural network resulted in a correlation coefficient of 0.97 and mean-squared error of  $3.4 \times 10^{-6}$ . Subsequent simulation of the neural network with unseen resulted in a correlation coefficient of 0.90 and mean-squared error of  $1.9 \times 10^{-5}$ . The neural network testing for the estimation of sphericity was done as with phase 1 where all neuron configurations were tested using both training algorithms. The best performing neural network for the prediction of sphericity has 75 neurons in the hidden layer and uses the Bayesian regularisation training algorithm for the training of the neural network. The training of the best neural network for the prediction of sphericity resulted in a correlation coefficient of 0.98 and mean-squared error of  $1.6 \times 10^{-7}$ . The simulation of the neural network with unseen resulted in a correlation coefficient of 0.96 and mean-squared error of  $4.5 \times 10^{-7}$ . For both the porosity and sphericity data, the hyperbolic tangent activation function was found to outperform the sigmoidal and linear activation functions in any choice of neural network.

Keywords: Artificial neural networks, predictive model, selectivity variables, porosity, sphericity, fluidised bed granulation, ammonium nitrate

## Table of contents

|                                                     |              |
|-----------------------------------------------------|--------------|
| <b>Acknowledgements.....</b>                        | <b>ii</b>    |
| <b>Abstract .....</b>                               | <b>iii</b>   |
| <b>List of Tables .....</b>                         | <b>ix</b>    |
| <b>List of Figures .....</b>                        | <b>xi</b>    |
| <b>Table of Abbreviations.....</b>                  | <b>xvii</b>  |
| <b>Table of Symbols.....</b>                        | <b>xviii</b> |
| <b>Chapter 1: Introduction.....</b>                 | <b>1</b>     |
| 1.1    Background.....                              | 1            |
| 1.1.1    Fluidised bed granulation .....            | 1            |
| 1.1.2    Artificial neural networks.....            | 2            |
| 1.2    Focus of the study.....                      | 3            |
| 1.3    Objectives .....                             | 3            |
| 1.4    Scope of the study .....                     | 4            |
| <b>Chapter 2: Literature study.....</b>             | <b>5</b>     |
| 2.1    Fluidised bed granulation .....              | 5            |
| 2.1.1    Operational challenges.....                | 6            |
| 2.1.2    Types of fluidised bed granulators.....    | 6            |
| 2.1.3    Granulation growth mechanisms .....        | 7            |
| 2.1.3.1    Agglomeration.....                       | 8            |
| 2.1.3.2    Layering .....                           | 8            |
| 2.1.4    Operational parameters .....               | 8            |
| 2.1.4.1    Fluidising air flow rate .....           | 9            |
| 2.1.4.2    Fluidising air temperature .....         | 9            |
| 2.1.4.3    Liquid spray concentration/density ..... | 10           |
| 2.1.4.4    Liquid spray temperature .....           | 10           |
| 2.1.4.5    Liquid spray flow rate.....              | 10           |
| 2.1.4.6    Atomising air pressure/flow .....        | 11           |
| 2.1.4.7    Summary .....                            | 12           |
| 2.1.5    Product quality .....                      | 13           |
| 2.1.5.1    Porosity.....                            | 13           |
| 2.1.5.2    Sphericity .....                         | 16           |
| 2.1.6    Fluidised bed control .....                | 17           |
| 2.1.6.1    Soft-sensor technology .....             | 18           |
| 2.2    Artificial neural network .....              | 19           |
| 2.2.1    Types of artificial neural networks .....  | 22           |
| 2.2.1.1    Feed-forward neural networks .....       | 22           |

|                                                |                                                                   |    |
|------------------------------------------------|-------------------------------------------------------------------|----|
| 2.2.1.2                                        | Recurrent neural networks .....                                   | 23 |
| 2.2.2                                          | Activation function.....                                          | 25 |
| 2.2.3                                          | Training algorithm.....                                           | 27 |
| 2.2.3.1                                        | Levenberg-Marquardt algorithm .....                               | 31 |
| 2.2.3.2                                        | Bayesian regularisation method .....                              | 32 |
| 2.2.4                                          | Performance evaluation.....                                       | 33 |
| 2.3                                            | Chapter summary.....                                              | 35 |
| <b>Chapter 3: Experimental procedure .....</b> | <b>37</b>                                                         |    |
| 3.1                                            | Selectivity variable measurement and validation .....             | 37 |
| 3.1.1                                          | Porosity measurement.....                                         | 37 |
| 3.1.1.1                                        | Oil absorption.....                                               | 37 |
| 3.1.1.2                                        | Optical porosity measurement method .....                         | 39 |
| 3.1.1.3                                        | Comparison of methods .....                                       | 40 |
| 3.1.2                                          | Sphericity measurement and the necessity of repeatability .....   | 40 |
| 3.1.2.1                                        | Experimental setup .....                                          | 41 |
| 3.2                                            | Data collection for the predictive model .....                    | 42 |
| 3.2.1                                          | Phase 1 data collection and preparation .....                     | 42 |
| 3.2.1.1                                        | Conditional filtering .....                                       | 42 |
| 3.2.1.2                                        | Statistical filtering.....                                        | 43 |
| 3.2.1.3                                        | Data sets produced and assessed in phase 1 model development..... | 45 |
| 3.2.2                                          | Phase 2 data collection and preparation .....                     | 46 |
| 3.2.2.1                                        | Data processing .....                                             | 47 |
| 3.2.2.2                                        | Process sampling .....                                            | 47 |
| 3.2.2.3                                        | Sample analysis.....                                              | 49 |
| 3.2.2.4                                        | Time-advance determination .....                                  | 49 |
| 3.2.2.5                                        | Linear statistical sensor filtering .....                         | 50 |
| 3.3                                            | Chapter summary.....                                              | 50 |
| <b>Chapter 4: Model development .....</b>      | <b>51</b>                                                         |    |
| 4.1                                            | Predictive model based on phase 1 data.....                       | 51 |
| 4.1.1                                          | Model equipment .....                                             | 51 |
| 4.1.2                                          | Model procedure.....                                              | 51 |
| 4.2                                            | Predictive model based on phase 2 data.....                       | 54 |
| 4.2.1                                          | Model equipment .....                                             | 54 |
| 4.2.2                                          | Model procedure.....                                              | 54 |
| 4.2.2.1                                        | Porosity .....                                                    | 55 |
| 4.2.2.2                                        | Sphericity .....                                                  | 55 |
| 4.3                                            | Chapter summary.....                                              | 56 |
| <b>Chapter 5: Results and discussion .....</b> | <b>57</b>                                                         |    |
| 5.1                                            | Selectivity variables measurement and validation .....            | 57 |

|                                                                   |                                                                                    |            |
|-------------------------------------------------------------------|------------------------------------------------------------------------------------|------------|
| 5.1.1                                                             | Porosity measurement.....                                                          | 57         |
| 5.1.1.1                                                           | Oil absorption.....                                                                | 58         |
| 5.1.1.2                                                           | Optical porosity measurement (OPM) method.....                                     | 59         |
| 5.1.1.3                                                           | Comparison of porosity techniques .....                                            | 60         |
| 5.1.2                                                             | Sphericity repeatability .....                                                     | 62         |
| 5.2                                                               | Artificial neural network .....                                                    | 62         |
| 5.2.1                                                             | Phase 1 predictive model .....                                                     | 62         |
| 5.2.1.1                                                           | Data processing.....                                                               | 62         |
| 5.2.1.2                                                           | Artificial neural network training and testing on the filtered data set .....      | 66         |
| 5.2.1.3                                                           | Artificial neural network training and testing on the discrete data set .....      | 78         |
| 5.2.1.4                                                           | Artificial neural network training and testing on the constant-time data set ..... | 85         |
| 5.2.1.5                                                           | Assessing the influence of the time-interval on the constant-time data set .....   | 95         |
| 5.2.1.6                                                           | Assessing the influence of sensor filtering on constant-time data set.....         | 96         |
| 5.2.1.7                                                           | Dataset comparison.....                                                            | 103        |
| 5.2.2                                                             | Phase 2 predictive model .....                                                     | 104        |
| 5.2.2.1                                                           | Sample analysis.....                                                               | 104        |
| 5.2.2.2                                                           | Data processing.....                                                               | 105        |
| 5.2.2.3                                                           | Artificial neural network training and testing on porosity estimation data.....    | 106        |
| 5.2.2.4                                                           | Artificial neural network training and testing on sphericity estimation data ..... | 113        |
| 5.3                                                               | Chapter summary.....                                                               | 122        |
| <b>Chapter 6: Conclusion and recommendations.....</b>             |                                                                                    | <b>123</b> |
| 6.1                                                               | Conclusion .....                                                                   | 123        |
| 6.1.1                                                             | Selectivity variable measurement and validation .....                              | 123        |
| 6.1.2                                                             | Artificial neural network .....                                                    | 123        |
| 6.2                                                               | Recommendations .....                                                              | 124        |
| 6.2.1                                                             | Selectivity variable measurement and validation .....                              | 124        |
| 6.2.2                                                             | Artificial neural network .....                                                    | 125        |
| <b>Chapter 7: References .....</b>                                |                                                                                    | <b>126</b> |
| <b>Appendix A: Experimental calculations and procedures .....</b> |                                                                                    | <b>131</b> |
| A.1                                                               | Oil absorption.....                                                                | 131        |
| A.2                                                               | Critical values for confidence interval .....                                      | 135        |
| <b>Appendix B: Results .....</b>                                  |                                                                                    | <b>136</b> |
| B.1                                                               | Oil absorption.....                                                                | 136        |
| B.2                                                               | Optical porosity measurement method .....                                          | 137        |
| B.3                                                               | Sphericity .....                                                                   | 137        |
| <b>Appendix C: Phase 1 data processing.....</b>                   |                                                                                    | <b>138</b> |
| C.1                                                               | Raw data: Statistical filtering .....                                              | 138        |
| C.2                                                               | Phase 1 data graphs .....                                                          | 143        |
| C.3                                                               | Regression plots .....                                                             | 147        |

|                                                                   |            |
|-------------------------------------------------------------------|------------|
| <b>Appendix D: Experimental calculations and procedures .....</b> | <b>161</b> |
| D.1 Phase 2 porosity calculations .....                           | 161        |
| <b>Appendix E: Phase 2 data processing .....</b>                  | <b>168</b> |
| E.1 Phase 2 porosity data graphs.....                             | 168        |
| E.2 Regression plots for phase 2 porosity estimation .....        | 169        |
| E.3 Phase 2 sphericity data graphs .....                          | 174        |
| E.4 Regression plots for phase 2 sphericity estimation .....      | 175        |
| <b>Appendix F: Raw data.....</b>                                  | <b>182</b> |
| F.1 Phase 1 raw data .....                                        | 182        |
| F.2 Phase 2 raw data .....                                        | 186        |

## List of Tables

|                                                                                                                                                    |     |
|----------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Table 2.1-1: Summary of operating parameters and their effect on growth.....                                                                       | 12  |
| Table 2.1-2: Pore type descriptions .....                                                                                                          | 13  |
| Table 2.2-1: Advantages and disadvantages of error measurements .....                                                                              | 34  |
| Table 3.2-1: Down-time filtering conditions.....                                                                                                   | 43  |
| Table 3.2-2: Logical limits for data filtering.....                                                                                                | 43  |
| Table 3.2-3: Box-and-whisker diagram values .....                                                                                                  | 44  |
| Table 3.2-4: Input parameter changes from phase 1 to phase 2.....                                                                                  | 47  |
| Table 4.1-1: Training variables for Levenberg-Marquardt training algorithm .....                                                                   | 52  |
| Table 4.1-2: Training variables for Bayesian regularisation training algorithm .....                                                               | 53  |
| Table 5.1-1: Oil absorption granule porosity results.....                                                                                          | 58  |
| Table 5.1-2: Optical porosity measurement results.....                                                                                             | 60  |
| Table 5.1-3: Sphericity repeatability results .....                                                                                                | 62  |
| Table 5.2-1: Statistical filtering values .....                                                                                                    | 63  |
| Table 5.2-2: Performance of Levenberg-Marquardt algorithm on filtered data .....                                                                   | 67  |
| Table 5.2-3: Performance of Bayesian regularisation algorithm on filtered data.....                                                                | 73  |
| Table 5.2-4: Comparison of training algorithms using filtered data .....                                                                           | 78  |
| Table 5.2-5: Performance of Levenberg-Marquardt algorithm on discrete data .....                                                                   | 79  |
| Table 5.2-6: Performance of Bayesian regularisation algorithm on discrete data.....                                                                | 82  |
| Table 5.2-7: Comparison of training algorithms using discrete data.....                                                                            | 85  |
| Table 5.2-8: Performance of Levenberg-Marquardt algorithm on constant-time data.....                                                               | 86  |
| Table 5.2-9: Performance of Bayesian regularisation algorithm on constant-time data.....                                                           | 89  |
| Table 5.2-10: Comparison of training algorithms using constant-time data.....                                                                      | 93  |
| Table 5.2-11: Performance of time-interval testing for constant-time data .....                                                                    | 95  |
| Table 5.2-12: Time constants for linear sensor filtering for phase 1 .....                                                                         | 97  |
| Table 5.2-13: Comparison between normal and sensor filtered 10-minute constant-time<br>data using the Levenberg-Marquardt training algorithm ..... | 97  |
| Table 5.2-14: Comparison between normal and sensor filtered 10-minute constant-time<br>data using Bayesian regularisation training algorithm.....  | 100 |
| Table 5.2-15: Comparison of neural network performance with different datasets .....                                                               | 103 |
| Table 5.2-16: Time constants for linear sensor filtering for phase 2 .....                                                                         | 106 |
| Table 5.2-17: Performance of phase 2 porosity estimation using the Bayesian<br>regularisation training algorithm .....                             | 108 |
| Table 5.2-18: Activation function comparison using phase 2 porosity data .....                                                                     | 111 |
| Table 5.2-19: Performance of Levenberg-Marquardt algorithm on phase 2 sphericity data ...                                                          | 114 |

|                                                                                                |     |
|------------------------------------------------------------------------------------------------|-----|
| Table 5.2-20: Performance of Bayesian regularisation algorithm on phase 2 sphericity data..... | 117 |
| Table 5.2-21: Comparison of training algorithms using phase 2 sphericity data.....             | 120 |
| Table 5.2-22: Activation function comparison using phase 2 sphericity data.....                | 122 |
| Table A.1-1: Sample 1 experimental values - mass oil absorbed.....                             | 131 |
| Table A.1-2: Five-number summary for sample 1 - mass oil absorbed .....                        | 131 |
| Table A.1-3: Sample 8 experimental values - mass oil absorbed.....                             | 132 |
| Table A.1-4: Five-number summary for sample 8 - mass oil absorbed .....                        | 132 |
| Table A.1-5: Repeatability results for oil absorption - volume and mass .....                  | 133 |
| Table A.1-6: Critical t-values for oil absorption repeatability .....                          | 134 |
| Table A.1-7: Repeatability calculations and results for oil absorption.....                    | 134 |
| Table B.1-1: Constants for oil absorption calculations .....                                   | 136 |
| Table B.1-2: Oil absorption calculations .....                                                 | 136 |
| Table B.2-1: Constants for optical porosity measurement calculations .....                     | 137 |
| Table B.2-2: optical porosity measurement calculations .....                                   | 137 |
| Table B.3-1: Sphericity repeatability results.....                                             | 137 |
| Table C.1-1: Five-number summary - Fluidising air flow rate .....                              | 138 |
| Table C.1-2: Five-number summary - Fluidising air temperature .....                            | 139 |
| Table C.1-3: Five-number summary - Liquid ammonium nitrate density .....                       | 139 |
| Table C.1-4: Five-number summary - Liquid ammonium nitrate temperature .....                   | 140 |
| Table C.1-5: Five-number summary - Liquid ammonium nitrate flow rate .....                     | 141 |
| Table C.1-6: Five-number summary - Atomising air flow rate .....                               | 141 |
| Table C.1-7: Five-number summary - Oil absorption .....                                        | 142 |
| Table D.1-1: Optical porosity measurement and oil absorption values for phase 2 .....          | 161 |
| Table D.1-2: Repeatability on optical porosity measurement method on 05/09 17:10 sample .....  | 162 |
| Table D.1-3: Repeatability on optical porosity measurement method on 05/10 18:00 sample .....  | 163 |
| Table D.1-4: Repeatability on optical porosity measurement method on 05/11 05:50 sample .....  | 164 |
| Table D.1-5: Repeatability on oil absorption on 05/09 17:10 sample .....                       | 165 |
| Table D.1-6: Repeatability on oil absorption on 05/10 18:00 sample .....                       | 166 |
| Table D.1-7: Repeatability on oil absorption on 05/11 18:00 sample .....                       | 167 |
| Table F.1-1: Raw data - Layout of Excel spreadsheets.....                                      | 182 |
| Table F.2-1: Raw data - Layout of Excel spreadsheets.....                                      | 186 |

## List of Figures

|                                                                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figure 1.1-1: Fluidising bed granulator (adapted from Link & Schlünder, 1997).....                                                                              | 1  |
| Figure 1.4-1: Scope of the study.....                                                                                                                           | 4  |
| Figure 2.1-1: Granulation growth mechanisms (taken from Kok, 2016) .....                                                                                        | 7  |
| Figure 2.1-2: Fluidised bed granulator with input parameters and selectivity variables .....                                                                    | 9  |
| Figure 2.1-3: Schematic cross section of a porous solid (taken from Rouquerol <i>et al.</i> ,<br>1994).....                                                     | 13 |
| Figure 2.1-4: SEM images of agriculture grade ammonium nitrate (left) and explosive<br>grade ammonium nitrate (right) (taken from Lotspeich & Petr, 2015) ..... | 16 |
| Figure 2.2-1: Biological neuron.....                                                                                                                            | 19 |
| Figure 2.2-2: Layout of a neural network example (adapted from Dongare <i>et al.</i> , 2012).....                                                               | 20 |
| Figure 2.2-3: Example of a neural network node (adapted from Burns, 2001:348) .....                                                                             | 21 |
| Figure 2.2-4: A real-time recurrent neural network example (adapted from Cirstea <i>et al.</i> ,<br>2002).....                                                  | 23 |
| Figure 2.2-5: Hopfield neural network example (adapted from Burns, 2001: 351).....                                                                              | 24 |
| Figure 2.2-6: Cellular neural network example (adapted from Cirstea <i>et al.</i> , 2002).....                                                                  | 25 |
| Figure 2.2-7: Sigmoidal activation function (adapted from Burns, 2001:349) .....                                                                                | 26 |
| Figure 2.2-8: Hyperbolic tangent activation function (adapted from Burns, 2001:349) .....                                                                       | 26 |
| Figure 2.2-9: Linear activation function (adapted from Burns, 2001:349).....                                                                                    | 27 |
| Figure 2.2-10: Levenberg-Marquardt algorithm (taken from Yu & Wilamowski, 2011) .....                                                                           | 32 |
| Figure 3.1-1: Schematic representation of the oil absorption apparatus (taken from Kok,<br>2016:31).....                                                        | 38 |
| Figure 3.2-1: Box-and-whisker diagram illustration .....                                                                                                        | 44 |
| Figure 3.2-2: Compilation of constant-time data .....                                                                                                           | 46 |
| Figure 3.2-3: Ammonium nitrate granulator process flow diagram .....                                                                                            | 48 |
| Figure 4.1-1: Neural network configurations and topologies tested - Phase 1.....                                                                                | 52 |
| Figure 4.1-2: Division of data for neural network testing .....                                                                                                 | 54 |
| Figure 4.2-1: Neural network configurations tested – Phase 2 Porosity data .....                                                                                | 55 |
| Figure 4.2-2: Neural network configurations tested – Phase 2 Sphericity data .....                                                                              | 56 |
| Figure 5.1-1: Oil absorption results.....                                                                                                                       | 59 |
| Figure 5.1-2: Optical porosity measurement results.....                                                                                                         | 60 |
| Figure 5.1-3: Comparison of different porosity measurement techniques .....                                                                                     | 61 |
| Figure 5.1-4: Linear regression between mass-based porosity and OPM results.....                                                                                | 61 |
| Figure 5.2-1: Partial sets of filtered data .....                                                                                                               | 64 |
| Figure 5.2-2: Partial discrete data.....                                                                                                                        | 65 |

|                                                                                                                                                                              |    |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figure 5.2-3: Partial constant-time data.....                                                                                                                                | 66 |
| Figure 5.2-4: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training algorithm on filtered data .....                                      | 68 |
| Figure 5.2-5: Regression plot of the training of a 72-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data .....                         | 69 |
| Figure 5.2-6: Regression of the simulation of a 72-neuron network trained with the Levenberg-Marquardt algorithm using filtered data .....                                   | 70 |
| Figure 5.2-7: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training and filtered data; including the best-case neuron configuration ..... | 71 |
| Figure 5.2-8: Model performance of a 72-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data .....                                       | 72 |
| Figure 5.2-9: MSE-values and R-values for neuron-configurations using the Bayesian regularisation training and filtered data.....                                            | 74 |
| Figure 5.2-10: Regression plot of the training of a 113-neuron neural network trained with the Bayesian regularisation algorithm using filtered data.....                    | 75 |
| Figure 5.2-11: Regression of the simulation of a 113-neuron network trained with the Bayesian regularisation algorithm using filtered data.....                              | 76 |
| Figure 5.2-12: Model performance of a 125-neuron neural network trained with the Bayesian regularisation algorithm using filtered data.....                                  | 77 |
| Figure 5.2-13: Regression plot of the training and simulation of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using discrete data.....         | 80 |
| Figure 5.2-14: Model performance of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using discrete data.....                                      | 81 |
| Figure 5.2-15: Regression plot of the training and simulation of a 100-neuron neural network trained with the Bayesian regularisation algorithm using discrete data.....     | 83 |
| Figure 5.2-16: Model performance of a 100-neuron neural network trained with the Bayesian regularisation algorithm using discrete data.....                                  | 84 |
| Figure 5.2-17: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training and constant-time data .....                                         | 87 |
| Figure 5.2-18: Regression plots of the training and simulation of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using constant-time data .....   | 88 |
| Figure 5.2-19: Model performance of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using constant-time data.....                                  | 90 |

|                                                                                                                                                                                                |     |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figure 5.2-20: MSE-values and R-values for neuron-configurations using the Bayesian regularisation training and constant-time data.....                                                        | 91  |
| Figure 5.2-21: Regression plots of the training and simulation of a 69-neuron neural network trained with the Bayesian regularisation algorithm using constant-time data.....                  | 92  |
| Figure 5.2-22: Model performance of a 69-neuron neural network trained with the Bayesian regularisation algorithm using constant-time data.....                                                | 94  |
| Figure 5.2-23: MSE and R-values for time-interval testing for constant-time data.....                                                                                                          | 96  |
| Figure 5.2-24: Regression plots of the training and simulation of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using sensor filtered constant-time data.....     | 98  |
| Figure 5.2-25: Model performance of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using sensor filtered constant-time data.....                                   | 99  |
| Figure 5.2-26: Regression plots of the training and simulation of a 100-neuron neural network trained with the Bayesian regularisation algorithm using sensor filtered constant-time data..... | 101 |
| Figure 5.2-27: Model performance of a 100-neuron neural network trained with the Bayesian regularisation algorithm using sensor filtered constant-time data.....                               | 102 |
| Figure 5.2-28: Optical porosity measurement and oil absorption values for phase 2 .....                                                                                                        | 104 |
| Figure 5.2-29: Linear regression between oil absorption and optical porosity measurement for phase 2.....                                                                                      | 105 |
| Figure 5.2-30: Time-advance correlation coefficients for liquid AN temperature .....                                                                                                           | 106 |
| Figure 5.2-31: Partial phase 2 sensor input and porosity data .....                                                                                                                            | 107 |
| Figure 5.2-32 MSE-values and R-values for neuron-configurations using the Bayesian regularisation training algorithm and phase 2 porosity data.....                                            | 109 |
| Figure 5.2-33: Regression plots of the training and simulation of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data .....              | 110 |
| Figure 5.2-34: Model performance of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data .....                                            | 112 |
| Figure 5.2-35: Partial phase 2 sphericity data.....                                                                                                                                            | 113 |
| Figure 5.2-36: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training algorithm and phase 2 sphericity data .....                                            | 115 |
| Figure 5.2-37: Regression plots of the training and simulation of a 65-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data.....                 | 116 |

|                                                                                                                                                                |     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figure 5.2-38: Model performance of a 65-neuron neural network trained with the<br>Levenberg-Marquardt algorithm using phase 2 sphericity data.....            | 118 |
| Figure 5.2-39 MSE-values and R-values for neuron-configurations using the Levenberg-<br>Marquardt training algorithm and phase 2 sphericity data .....         | 119 |
| Figure 5.2-40: Model performance of a 75-neuron neural network trained with the<br>Bayesian regularisation algorithm using phase 2 sphericity data.....        | 121 |
| Figure A.1-1: Box-and-whisker diagram of sample 1 - mass oil absorbed .....                                                                                    | 132 |
| Figure A.1-2: Box-and-whisker diagram of sample 8 - mass oil absorbed .....                                                                                    | 133 |
| Figure A.2-1: t-critical values for confidence interval.....                                                                                                   | 135 |
| Figure C.1-1: Box-and-whisker diagram - Fluidising air flow rate .....                                                                                         | 138 |
| Figure C.1-2: Box-and-whisker diagram - Fluidising air temperature .....                                                                                       | 139 |
| Figure C.1-3: Box-and-whisker diagram - Liquid ammonium nitrate density .....                                                                                  | 140 |
| Figure C.1-4: Box-and-whisker diagram - Liquid ammonium nitrate temperature.....                                                                               | 140 |
| Figure C.1-5: Box-and-whisker diagram - Liquid ammonium nitrate flow rate .....                                                                                | 141 |
| Figure C.1-6: Box-and-whisker diagram - Atomising air flow rate .....                                                                                          | 142 |
| Figure C.1-7: Box-and-whisker diagram - Oil absorption .....                                                                                                   | 142 |
| Figure C.2-1: Phase 1 – original data graphs .....                                                                                                             | 143 |
| Figure C.2-2: Phase 1 – filtered data graphs .....                                                                                                             | 144 |
| Figure C.2-3: Phase 1 – discrete data graphs .....                                                                                                             | 145 |
| Figure C.2-4: Phase 1 – Constant-time data graphs .....                                                                                                        | 146 |
| Figure C.3-1: Regression plots of the training of a 75-neuron neural network trained with<br>the Levenberg-Marquardt algorithm using filtered data .....       | 147 |
| Figure C.3-2: Regression plot of the simulation of a 75-neuron neural network trained with<br>the Levenberg-Marquardt algorithm using filtered data .....      | 148 |
| Figure C.3-3: Regression plots of the training of a 72-neuron neural network trained with<br>the Levenberg-Marquardt algorithm using filtered data .....       | 149 |
| Figure C.3-4: Regression plots of the training of a 125-neuron neural network trained with<br>the Bayesian regularisation algorithm using filtered data.....   | 150 |
| Figure C.3-5: Regression plot of the simulation of a 125-neuron neural network trained<br>with the Bayesian regularisation algorithm using filtered data ..... | 151 |
| Figure C.3-6: Regression plots of the training of a 113-neuron neural network trained with<br>the Bayesian regularisation algorithm using filtered data.....   | 152 |
| Figure C.3-7: Regression plots of the training of a 100-neuron neural network trained with<br>the Levenberg-Marquardt algorithm using discrete data.....       | 153 |
| Figure C.3-8: Regression plots of the training of a 100-neuron neural network trained with<br>the Bayesian regularisation algorithm using discrete data .....  | 154 |

|                                                                                                                                                                           |     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figure C.3-9: Regression plots of the training of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data ...        | 155 |
| Figure C.3-10: Regression plot of the simulation of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data.....     | 156 |
| Figure C.3-11: Regression plots of the training of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data ...       | 157 |
| Figure C.3-12: Regression plots of the training of a 75-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data.....  | 158 |
| Figure C.3-13: Regression plot of the simulation of a 75-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data..... | 159 |
| Figure C.3-14: Regression plots of the training of a 69-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data.....  | 160 |
| Figure E.1-1: Phase 2 sensor inputs and porosity data.....                                                                                                                | 168 |
| Figure E.2-1: Regression plots of the training of a 50-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data .....         | 169 |
| Figure E.2-2: Regression plot of the simulation of a 50-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data .....        | 170 |
| Figure E.2-3: Regression plots of the training of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data .....         | 171 |
| Figure E.2-4: Regression plots of the training of a 61-neuron neural network using phase 2 porosity data and a sigmoidal activation function.....                         | 172 |
| Figure E.2-5: Regression plots of the training of a 61-neuron neural network using phase 2 porosity data and a linear activation function.....                            | 173 |
| Figure E.3-1: Phase 2 sensor inputs and sphericity data .....                                                                                                             | 174 |
| Figure E.4-1: Regression plots of the training of a 50-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data.....            | 175 |
| Figure E.4-2: Regression plot of the simulation of a 50-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data.....           | 176 |
| Figure E.4-3: Regression plots of the training of a 65-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data.....            | 177 |
| Figure E.4-4: Regression plots of the training of a 75-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 sphericity data.....        | 178 |

|                                                                                                                                                                        |     |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figure E.4-5: Regression plot of the simulation of a 75-neuron neural network trained with<br>the Bayesian regularisation algorithm using phase 2 sphericity data..... | 179 |
| Figure E.4-6: Regression plots of the training of a 75-neuron neural network using phase<br>2 sphericity data and a sigmoidal activation function .....                | 180 |
| Figure E.4-7: Regression plots of the training of a 75-neuron neural network using phase<br>2 sphericity data and a linear activation function .....                   | 181 |

## Table of Abbreviations

| Abbreviation | Definition                       |
|--------------|----------------------------------|
| AN           | Ammonium nitrate                 |
| ANFO         | Ammonium nitrate fuel oil        |
| ANN          | Artificial neural network        |
| BET          | Brunauer-Emmet-Teller            |
| CMC          | Carboxymethyl cellulose          |
| HDAN         | High dense ammonium nitrate      |
| MAE          | Mean absolute error              |
| MAPE         | Mean absolute percentage error   |
| MSE          | Mean-squared error               |
| PGAN         | Porous granular ammonium nitrate |
| RMSE         | Root mean-squared error          |
| SEM          | Scanning electron microscope     |
| TEM          | Transmission electron microscope |
| VTM          | Volume-to-mass                   |

## Table of Symbols

| Symbol                     | Definition                                                          | Unit              |
|----------------------------|---------------------------------------------------------------------|-------------------|
| $\alpha$                   | Linear statistical filter coefficient                               | -                 |
| $d_{50}$                   | Particle diameter                                                   | m                 |
| $e_t$ or $\epsilon$ or $E$ | Error; difference between experimental and predicted values         | -                 |
| $J_i$                      | Jacobian matrix                                                     | -                 |
| $m_i$                      | Mass                                                                | g                 |
| $N$ or $n$                 | Number of values                                                    | -                 |
| $\rho_i$                   | Density                                                             | kg/m <sup>3</sup> |
| $\pi$                      | Pi                                                                  | -                 |
| $R$                        | Correlation coefficient                                             | -                 |
| $R^2$                      | Coefficient of determination                                        | -                 |
| $\sigma_i$                 | Standard deviation                                                  | -                 |
| $t_{n-1}$                  | Critical t-value                                                    | -                 |
| $\Delta t$                 | Sampling interval for linear statistical filtering                  | min               |
| $\tau_f$                   | Time constant for linear statistical filtering                      | min               |
| $\mu$                      | Damping parameter                                                   | -                 |
| $V_i$                      | Volume                                                              | m <sup>3</sup>    |
| $\bar{x}$ or $\bar{y}$     | Average value                                                       | -                 |
| $x_i$                      | Experimental/target values                                          | -                 |
| $y_i$                      | Predicted/output values                                             | -                 |
| $Y_i$ or $Y_{i-1}$         | Current and previous filter values for linear statistical filtering | -                 |
| $w_k$                      | Weight                                                              | -                 |

## Chapter 1: Introduction

### 1.1 Background

#### 1.1.1 Fluidised bed granulation

Fluidised bed granulation is most widely used in processes that handle particulate systems; this includes industries such as chemical, pharmaceutical, agrochemical and food industries (Aleksić *et al.*, 2015; Burggraeve *et al.*, 2013; Moraga *et al.*, 2015). Patel *et al.* (2010) describes fluidised bed granulation as an intricate process consisting of three parts, namely (1) wetting and nucleation, (2) consolidation and growth, and (3) breakage and attrition.

Fluidised bed granulation can briefly be described as a particle growth process; small particles are transformed into larger particles, wherein the original particle is still present (Burggraeve *et al.*, 2013). The method is known for the fact that mixing, wetting and drying are carried out in one piece of equipment, which is also one of the main advantages of fluidised bed granulation (Moraga *et al.*, 2015). Link & Schlünder (1997) explains the process of granulation. Seed particles (1) are fluidised through a distributor plate (2) using hot gas (3). Liquid is sprayed into the granulator using nozzles, to induce growth of the particles (4). The process is shown in Figure 1.1-1.

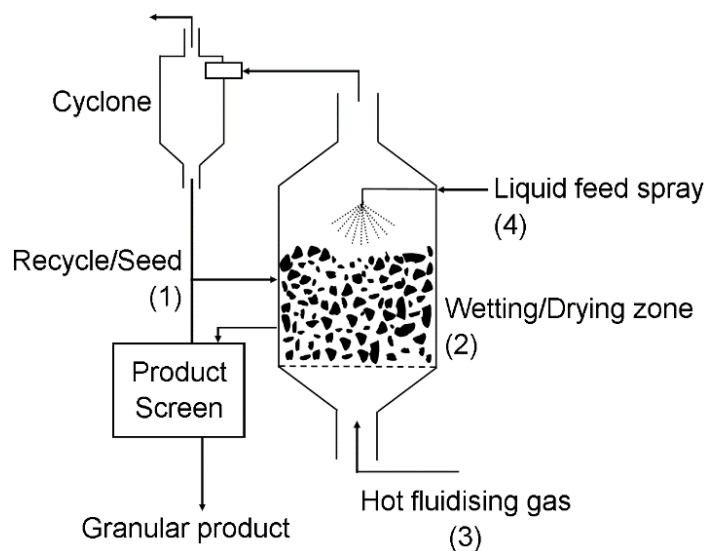


Figure 1.1-1: Fluidising bed granulator (adapted from Link & Schlünder, 1997)

Aleksić *et al.* (2015) points out that even though fluidised bed granulation is a widely-used application, it is dependent on the experience of the operator. Aleksić *et al.* (2015) point out that the optimisation of the process has proven difficult and time consuming since the product quality

depends on many factors. Artificial neural networks have been successful in overcoming this obstacle by modelling of the fluidised bed granulation process (Aleksić *et al.*, 2014).

Granulation growth mechanisms that exist in fluidised bed granulators are agglomeration, layering or a combination of the two. Agglomeration occurs when particles are joined together with liquid bridges to establish growth whereas layering occurs when one particle grows by layers forming around the particle (Kok, 2016). Aleksić *et al.* (2015) states that agglomeration is not very controllable in melt granulation. The reason for this is rapid growth due to the molten liquid being sprayed into the fluidised bed granulator and particles having more chance to be wet and coalesce. Layering as a growth mechanism is increased with a decrease in liquid binder flow rate at a constant fluidisation velocity (Moraga *et al.*, 2015). Layering also improves sphericity of granulated particles (Aleksić *et al.*, 2015).

The operating parameters of the fluidised bed granulator that will be investigated are fluidising air flow rate and temperature, liquid flow rate, temperature and concentration and atomising air pressure. The parameters are shown in Figure 2.1-2 with a fluidised bed granulator as well as the selectivity variables.

The selectivity variables are porosity and sphericity. These variables are interrelated and directly related to granule shape. The shape depends mainly on the growth mechanisms involved (Aleksić *et al.*, 2015). Layering is known to improve sphericity, whereas agglomerated particles are not as spherical.

### 1.1.2 Artificial neural networks

An artificial neural network (ANN) is defined as a non-linear function approximators. ANNs are based on biological neural systems found in human and animal brains (Azadi & Karimi-Jashni, 2016; Dongare *et al.*, 2012; Satish & Setty, 2004). ANN models consist of interconnected computational elements such as neuron layers, nodes and weights (Figure 2.2-2). The architecture of an ANN enables it to learn from examples, in most cases historical data, to recognise patterns and obtain relationships. ANNs can frequently predict unseen data sufficiently even when noisy data is used for training.

There are several types of ANNs, feed-forward networks and recurrent neural networks being the most commonly used ANNs. The most prominent difference between feed-forward and recurrent neural networks is the topology. Feed-forward neural networks consist of only input and output variables being linked by hidden layers whereas recurrent neural networks consist of outputs

being sent back as to act as inputs as well. ANNs consist of three parts, namely the input layer, hidden layers and the output layers (Dongare *et al.*, 2012). The input layer receives all input parameters of the problem. The hidden layers consist of weights that represent the relationship between the inputs and outputs of the problem. The output layer gives the outputs of the problem.

## 1.2 Focus of the study

The purpose of this study is to develop an artificial neural network model that can predict the selectivity variables in an ammonium nitrate fluidised bed granulator using various feed properties and operating parameters as inputs.

Fluidised bed granulation is challenging to control due to multiple processes occurring in one piece of process equipment. Studies concluded that there exists no statistical relation between various input parameters and quality variables of a fluidised bed granulator (Kok, 2016; Eisenberg, 2016). This observation together with prior studies conducted by Burggraeve *et al.*, 2013 serve as verification for the use of artificial intelligence as it has shown promising results in the modelling of non-linear systems. Fuzzy logic as well as artificial neural networks have been investigated as possible methods to obtain a predictive model for the control of a fluidised bed granulator, however, artificial neural networks were found to perform better than fuzzy logic control (Burggraeve *et al.*, 2013).

Inference control is implemented when a quality variable cannot be measured online and requires sampling and laboratory analysis for the quality of the product to be known. The soft sensor, in this case an artificial neural network, predicts the quality of the product online. The operator will then be able to adjust input parameters according to the prediction, if the quality of the product is not what it should be.

Soft sensors save time and money by predicting a value while the process equipment is still running, instead of having to wait for laboratory results before knowing whether the right quality is being produced.

## 1.3 Objectives

1. Ensure that the measurement of the selectivity variables is adequate for optimum quality control as well as for application in machine-learning algorithms.
2. Phase 1 artificial neural network: Evaluate neural networks for the prediction of the porosity of ammonium nitrate granules, with operational parameters listed in Section 1.1 as input parameters.

3. Phase 2 artificial neural network: Using performance data obtained in phase 1, enhance the performance of a neural network for predicting the selectivity variables, i.e. porosity and sphericity of ammonium nitrate granules.

#### 1.4 Scope of the study

The layout of this report as well as the methodology of the study is given in Figure 1.4-1.

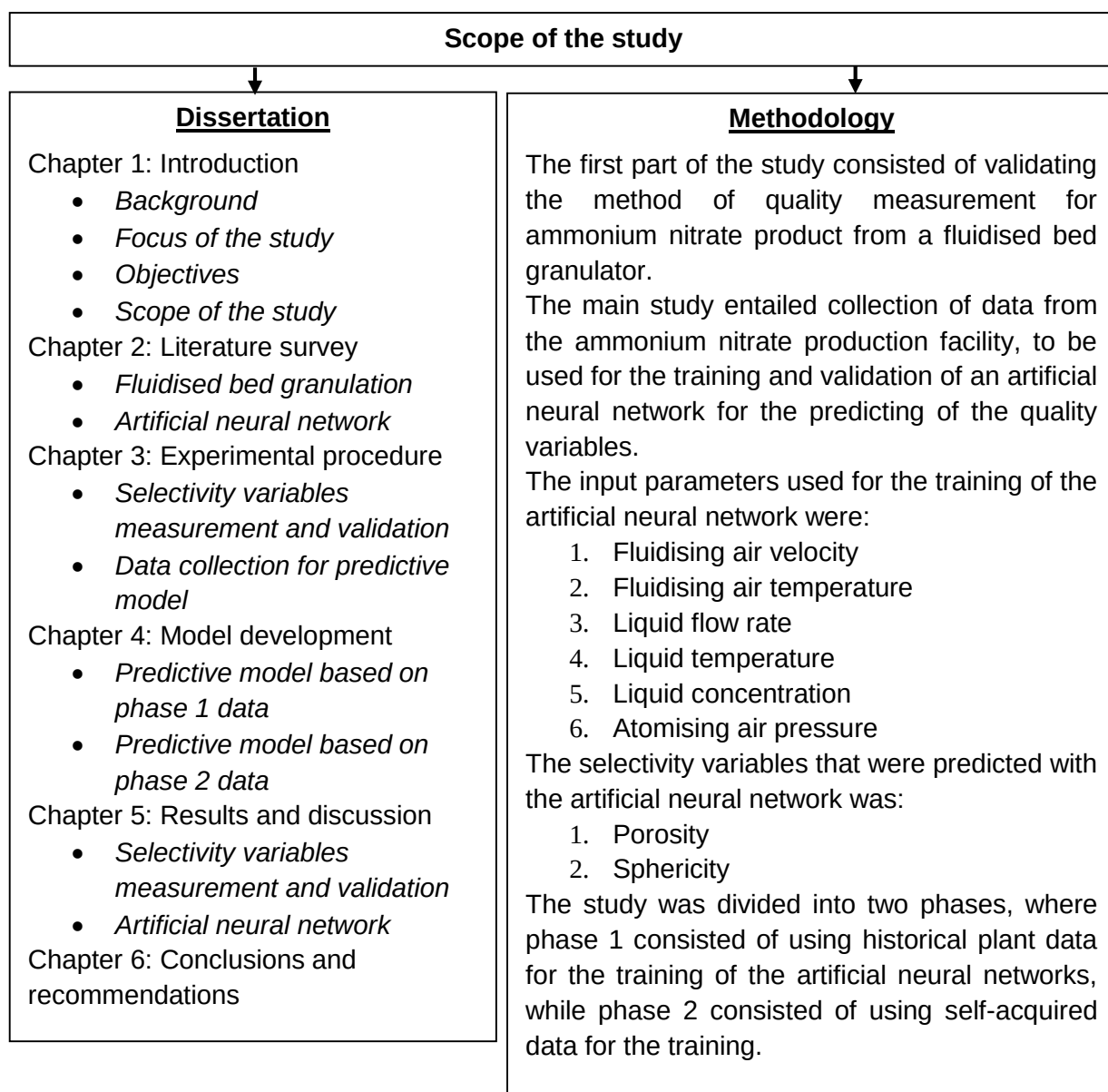


**Figure 1.4-1: Scope of the study**

## Chapter 2: Literature study

In this chapter, an in-depth literature study is done on both fluidised bed granulation as well as artificial neural network modelling.

Section 2.1 consists of the study of fluidised bed granulation. In this section the operational challenges, types of fluidised bed granulation and granulation growth are discussed. The effect of operational parameters as found by numerous authors, is also discussed along with the product quality variables being investigated. The porosity measurement techniques investigated for the first objective, is discussed under this section. The last subsection of Section 2.1 focusses on the control of fluidised bed granulators.

Section 2.2 discusses the literature obtained for the study of artificial neural networks. In this section the types of artificial neural networks, activation functions, training algorithms and performance evaluations are discussed. The two training algorithms used in the study are discussed in Section 2.2.3.

### 2.1 Fluidised bed granulation

The process of fluidised bed granulation (Figure 1.1-1) is described by several authors, usually all in consensus with one another. Fluidised bed granulation is a particle growth process in which dry particles are fluidised with hot air whilst a liquid binder (either fluid or melt) is injected into the process with a two-fluid nozzle (Becher & Schlünder, 1998; Hemati *et al.*, 2003; Link & Schlünder, 1997; Pont *et al.*, 2001; Srinivasakannan & Balasubramanian, 2003; Vengateson & Mohan, 2016).

The fluidising air's purpose in the process is to convert the particles from a static to a dynamic state. The flow rate of the air should be sufficient to allow particles to move upwards as well as downwards, creating a suspension of particles. If the fluidising air flow rate is too low, the particles will not move, whereas high fluidising air flow rates causes particles to overflow out of the granulator (Burggraave, 2013).

The liquid fluid flowing through the nozzle is usually atomised with pressurised air. The liquid binder forms droplets which either coat particles or form liquid bridges between particles. The addition of the liquid binder and the evaporation from the particles occur at the same time. The particles dry to grow by means of layering or agglomeration, as detailed in Section 2.1.3.

### 2.1.1 Operational challenges

Burggraeve (2013) lists the most common difficulties found in the granulation process; these include excessively coarse or fine granules being produced, poor fluidisation, final moisture content of particles being inconsistent and the final product being non-uniform. The formation of large agglomerates can affect the granulation process significantly, because the efficiency of the process depends on size distribution. If the agglomerates tend to grow more than they should, the process will change. Hemati *et al.* (2003) suggests sieving the product, although this will result in a larger than desired recycle ratio. Another problem that occurs in the granulation process is 'wet quenching,' which is the result of too large liquid droplet sizes being distributed into the granulator. If the liquid droplet size is much larger than the average particle size, the particles will not dry sufficiently, creating large wet lumps of particles in the fluidised bed. Dry quenching occurs when the agglomerates grow too fast and thus become too large to be fluidised (Becher & Schlünder, 1998).

In most cases, the reason for the abovementioned operational issues is that various operating parameters have an influence on the product quality. The operating parameters are also co-dependent, making it challenging to optimise a granulation process (Burggraeve, 2013).

### 2.1.2 Types of fluidised bed granulators

Burggraeve *et al.* (2013) divides granulation into wet and dry granulation, whereas Moraga *et al.* (2015) adds melt granulation to the list.

Wet granulation occurs when a binder liquid is added to the system with agitation (Burggraeve *et al.*, 2013; Moraga *et al.*, 2015). The process binds powder particles together with viscous and capillary forces. The binder is removed via evaporation during the drying part of the process so that permanent bonds form between particles. Moraga *et al.* (2015) divides the wet granulation process into two categories, based on the mixing principle; these categories include mechanical and pneumatic agitated units. Mechanical agitating units include pans, drums and high-shear granulators. Pneumatic agitating units are normally used in fluidised bed granulators.

Dry granulation is achieved by compacting the powder without the use of a binder liquid (Burggraeve *et al.*, 2013). The dry granulation method is less complicated and more cost efficient than wet granulation, however, wet granulation offers a more uniform product in terms of bulk density and compatibility. Moraga *et al.* (2015) states that dry granulation is mainly dependent on electrostatic forces to keep the powder together.

Melt granulation is like wet granulation, but in this case a melt is used to enlarge the particles (Maraga *et al.*, 2015). The binders in melt granulation can be divided into two different groups, namely in situ melt (also known as co-melt) and spray-on melt (Aleksić *et al.*, 2015; Moraga *et al.*, 2015). In situ melt is when the powders melt during granulation, whereas spray-on melt consists of spraying atomised molten liquids onto the granules. In situ melt is not viable if the seeds and binder powders have similar melting temperatures.

### 2.1.3 Granulation growth mechanisms

The growth of particles occurs either when liquid droplets attach to a particle to form a bridge with another particle which subsequently dries, or when a liquid layer forms on the surface of a single particle which subsequently dries. These two mechanisms are known as agglomeration and layering respectively. The major difference between agglomeration and layering is that layered particles are dense, spherical and strong whereas agglomerates are less spherical, more porous and dissolve more easily (Becher & Schlünder, 1998; Link & Schlünder, 1997).

In most cases both growth mechanisms occur simultaneously, with one mechanism being more dominant than the other. The dominant growth mechanism is determined by both operating and physicochemical properties of the fluidised bed granulator (Hemati *et al.*, 2003; Pont *et al.*, 2001; Srinivasakannan & Balasubramanian, 2003). The process of the growth of particles is shown in Figure 2.1-1 below with the explanation of each mechanism given in Sections 2.1.3.1 and 2.1.3.2.

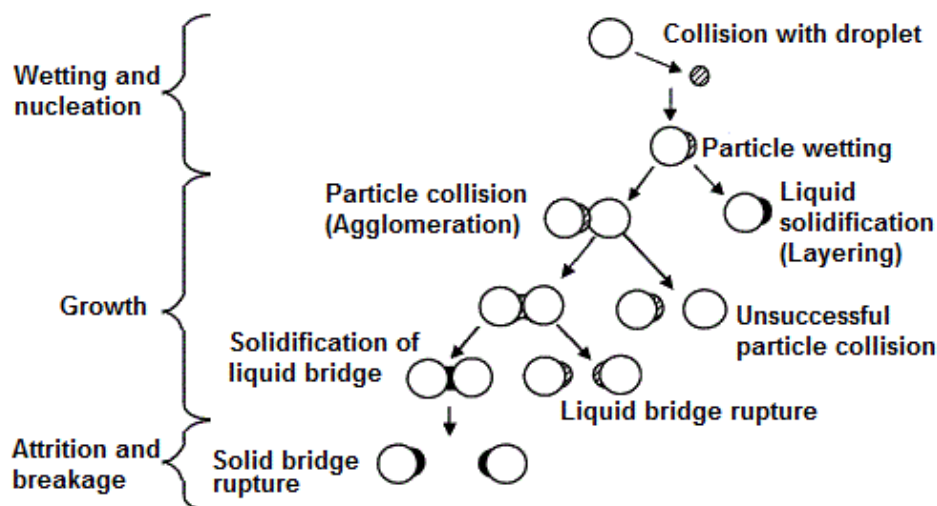


Figure 2.1-1: Granulation growth mechanisms (taken from Kok, 2016)

Srinivasakannan & Balasubramanian (2003) found that seed particle diameter has a significant influence on the growth mechanism, with smaller particles forming more agglomerates, while larger particles tend to favour layering.

### 2.1.3.1 Agglomeration

Agglomeration is the collision and adhesion of two or more particles (Becher & Schlünder, 1998; Hemati *et al.*, 2003; Pont *et al.*, 2001; Srinivasakannan & Balasubramanian, 2003). The formation of agglomerates depends on several factors, with moisture content and drying being the most significant. The moisture content and drying ability of a fluidised bed granulator will determine whether a wet particle will collide with another particle whilst still being wet, or dry before the particles could collide.

Hemati *et al.* (2003) explains that, although the main force joining particles together to form agglomerates is the liquid bridges, there are other forces that also play an initial role in the joining of the particles. These forces are Van der Waals forces, intermolecular attractive forces and electrostatic forces. Electrostatic forces are most significant as they keep the particles close to each other long enough for the particles to join.

Uncontrolled agglomeration can lead to two major problems in a fluidised bed granulator, namely wet quenching or dry quenching, as described in Section 2.1.1.

### 2.1.3.2 Layering

Layering or coating occurs when a particle grows by forming several fast-drying liquid layers on a small particle; typically, like layers in an onion (Hemati *et al.*, 2003; Pont *et al.*, 2001).

The formation of layers requires a particle to dry adequately before colliding with another particle, so that the particles do not adhere to one another whilst still colliding with liquid droplets to form a layer. Major factors that influence growth by layering are the fluidising air velocity as well as particle and droplet sizes. Inadequate particle and droplet sizes will either cause agglomeration or no growth at all (Link & Schlünder, 1997).

### 2.1.4 Operational parameters

The fluidised bed granulation process is ruled by its process parameters and their relationship with each other (Burggraeve, 2013; Moraga *et al.*, 2015; Patel *et al.*, 2010). Both the operating parameters of the granulator (Figure 2.1-2) as well as the physicochemical properties of the binder and seed particles play a role in the efficiency of the process and the quality of the product.

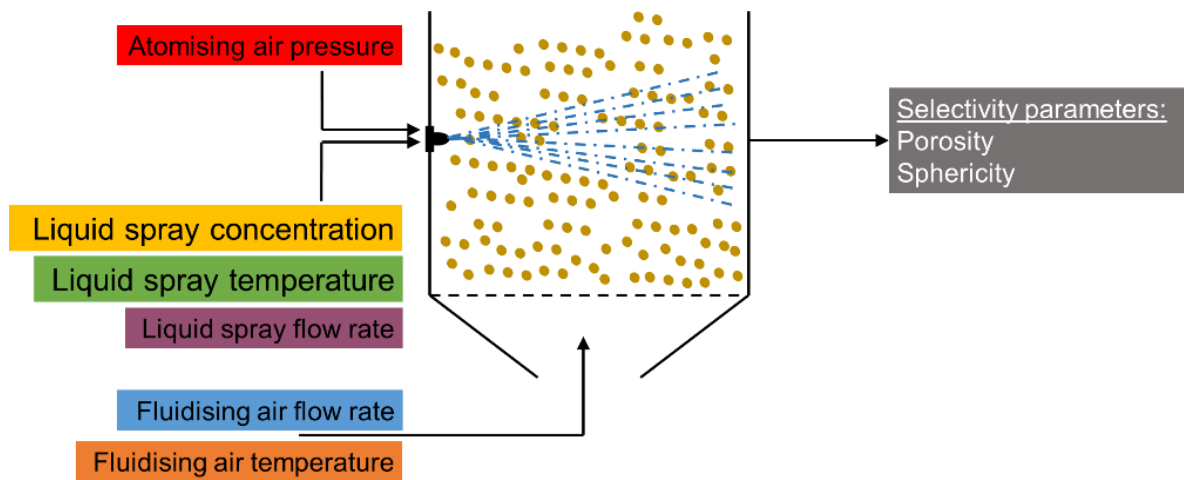


Figure 2.1-2: Fluidised bed granulator with input parameters and selectivity variables

#### 2.1.4.1 Fluidising air flow rate

Moraga *et al.* (2015) investigated the influence of fluidising air velocity (also termed fluidisation velocity) on the growth mechanism in a fluidised bed granulator and found that more agglomeration occurs at lower fluidisation velocities (Vengateson & Mohan, 2016). At higher fluidising velocity there is more heat transfer, thus causing particles to dry before colliding with each other and promoting layered growth.

Hemati *et al.* (2003) elaborates that a higher fluidisation velocity increases the drying capacity creating coated particles rather than agglomerates. A high fluidisation velocity does have a disadvantage, namely that it could cause a large amount of attrition breakage which decreases the efficiency of the fluidised bed granulator (Vengateson & Mohan, 2016).

Srinivasakannan & Balasubramanian (2003) found no significant influence of fluidising air flow rate on the particle growth, however it was discovered that a higher fluidising air flow rate decreased the chances of wet quenching in the fluidised bed.

#### 2.1.4.2 Fluidising air temperature

Becher & Schlünder (1998) explains the drying phenomenon in a fluidised bed granulation. Most granules contain sufficient heat at the start of the drying process to evaporate all the binder solvent present on the particle, thus growing by means of layering. If the bed temperature, and thus the particle temperature, is decreased, the particles cannot dry only by means of the stored heat. These particles stay wet for longer and tend to rather form liquid bridges to agglomerate with other particles.

Burggraave (2013) adds to this by stating that the drying process also depends on the ability of the fluidising air to absorb moisture, thus referring to the humidity of the fluidising air combined with its temperature.

In a study done by Moraga *et al.* (2015), layering was found to be the dominant growth mechanism in a bed operated at relatively high fluidising air temperatures. While a combination of layering and agglomeration was present when the temperature was reduced, the amount of agglomerates was still very low (Becher & Schlünder, 1998).

Coating is favoured by relatively high (but still lower than melting) fluidising air temperatures, whereas a combination of agglomeration and layering occurs when the temperature is lowered. The temperature of the bed, obtained from fluidising air temperature, does not have any significant influence on the yield of the process (Moraga *et al.*, 2015).

### **2.1.4.3 Liquid spray concentration/density**

Aleksić *et al.* (2015) notes that the content of the binder used in a fluidised bed granulator greatly influences the granule growth, with more emphasis on the size of product particles than the type of particles produced. This includes both the type of binder as well as the concentration (viscosity/density). Aleksić *et al.* (2015), Hemati *et al.* (2003) and Srinivasakannan & Balasubramanian (2003) found that higher concentrations of binder produce larger granules, but lower binder concentration produce more spherical granules.

Patel *et al.* (2010) found that higher binder concentration produces less friable granules, suggesting more layering. Moraga *et al.* (2015) explains that lower concentration binders promote the formation of liquid bridges, thus forming agglomerates. The liquid bridges formed with low concentration binders are weak and could break if the fluidising air temperature is too high.

### **2.1.4.4 Liquid spray temperature**

Liquid spray temperature is poorly described in literature with the main conclusion that its influence changes for each type of binder used in fluidised bed granulation.

### **2.1.4.5 Liquid spray flow rate**

In a study done by Hemati *et al.* (2003), sand was granulated by using different types of binders. In the study, it was found that the type of binder influences whether the binder flow rate affects the growth mechanism. The study concludes that sodium chloride (NaCl) as binder does not

influence the growth mechanism when it is varied, however when carboxymethyl cellulose (CMC) is used as binder, an increase in the flow rate increases the formation of agglomerates (Vengateson & Mohan, 2016).

Becher & Schlünder (1998) as well as Moraga *et al.* (2015) states that the formation of agglomerates depends on the liquid flow rate; they found a low, constant agglomerate fraction at low liquid flow rates with a rapid increase in the fraction of agglomerates as the liquid flow rate is increased. This is different from the conclusion of the abovementioned conclusions, because the type of binder being used plays an important role in the effect of the binder flow rate.

Srinivasakannan & Balasubramanian (2003) found that larger particles form at higher liquid flow rates while Patel *et al.* (2010) states that higher liquid flow rates improve the flowability of the particles.

Patel *et al.* (2010) declares that a low liquid binder flow rate promotes breakage as bridges that form between particles are weaker and non-cohesive. A higher liquid flow rate promotes a higher size distribution of the product particles (Moraga *et al.*, 2015).

#### **2.1.4.6 Atomising air pressure/flow**

The influence of atomising air pressure on the granulation process is interrelated to two other properties, namely the liquid flow rate and the size of seed particles.

The atomising air pressure along with the liquid flow rate determines the droplet size in the fluidised bed granulator. Higher atomising air flow rates was found to produce smaller droplets for a constant liquid binder flow rate (Chua *et al.* cited by Aleksić *et al.*, 2015; Moraga *et al.*, 2015).

Two nucleation mechanisms occur as a function of particle and droplet size (Patel *et al.*, 2010):

- Immersion mechanism:
- Liquid droplets are larger than seed particles
- Distribution mechanism:
- Seed particles are larger than liquid droplets

The distribution mechanism is mostly present in fluidised bed granulators with atomisation, promoting the formation of agglomerates (Patel *et al.*, 2010).

### 2.1.4.7 Summary

Table 2.1-1 provides a summary of the above-mentioned operating parameters and whether they promote agglomeration or layering.

**Table 2.1-1: Summary of operating parameters and their effect on growth**

| Operating parameter        | High/Low  | Promotes agglomeration (porous) | Promotes layering (spherical) | Reference                                                                                                                      |
|----------------------------|-----------|---------------------------------|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| Fluidising air flow rate   | High      |                                 |                               | -                                                                                                                              |
|                            | Low       | X                               |                               | Hemati <i>et al.</i> , 2003; Moraga <i>et al.</i> , 2015; Vengateson & Mohan, 2016                                             |
|                            | No effect |                                 |                               | Srinivasakannan & Balasubramanian, 2003                                                                                        |
| Fluidising air temperature | High      |                                 |                               | -                                                                                                                              |
|                            | Low       | X                               |                               | Becher & Schlünder, 1998; Burggraeve, 2013; Moraga <i>et al.</i> , 2015                                                        |
|                            | No effect |                                 |                               | -                                                                                                                              |
| Liquid spray concentration | High      |                                 | X                             | Aleksić <i>et al.</i> , 2015; Hemati <i>et al.</i> , 2003; Srinivasakannan & Balasubramanian, 2003; Patel <i>et al.</i> , 2010 |
|                            | Low       | X                               |                               | Moraga <i>et al.</i> , 2015                                                                                                    |
|                            | No effect |                                 |                               | -                                                                                                                              |
| Liquid spray flow rate     | High      | X                               |                               | Becher & Schlünder, 1998; Hemati <i>et al.</i> , 2003; Moraga <i>et al.</i> , 2015; Vengateson & Mohan, 2016                   |
|                            | Low       |                                 |                               | -                                                                                                                              |
|                            | No effect |                                 |                               | Hemati <i>et al.</i> , 2003                                                                                                    |
| Atomising air flow rate    | High      | X                               |                               | Patel <i>et al.</i> , 2010                                                                                                     |
|                            | Low       |                                 |                               | -                                                                                                                              |
|                            | No effect |                                 |                               | -                                                                                                                              |

## 2.1.5 Product quality

### 2.1.5.1 Porosity

Porosity is a property commonly found in most materials and porosity typically influence physical and chemical properties of a material, including strength, conductivity (thermal and electrical), translucency, density, chemical reactivity as well as interaction with fluids (Andreola *et al.*, 2000; Rouquerol *et al.*, 1994; Zou & Malzbender, 2016). The control of porosity is frequently important in the manufacturing environment as part of product quality, especially when producing products such as membranes, ceramics, industrial absorbents and catalysts (Rouquerol *et al.*, 1994).

Figure 2.1-3 shows a typical schematic representation of a porous particle. Different types of pores can exist in a particle. Based on the definition provided by Rouquerol *et al.* (1994) for the different pore types, Table 2.1-2 lists the different types of pores together with their location in a particle, with reference to the figure.

**Table 2.1-2: Pore type descriptions**

| Name          | Location  | Description                                                                          |
|---------------|-----------|--------------------------------------------------------------------------------------|
| Closed pores  | a         | Pores that are isolated from the outside                                             |
| Open pores    | b,c,d,e,f | Pores that are open to the external surface body                                     |
| Blind pores   | b,f       | Open pores which are only open at one end                                            |
| Through pores | e         | Open pores which are open at two ends of the particle                                |
| Roughness     | g         | Different from porosity in that rough edges are considered wider than they are deep. |

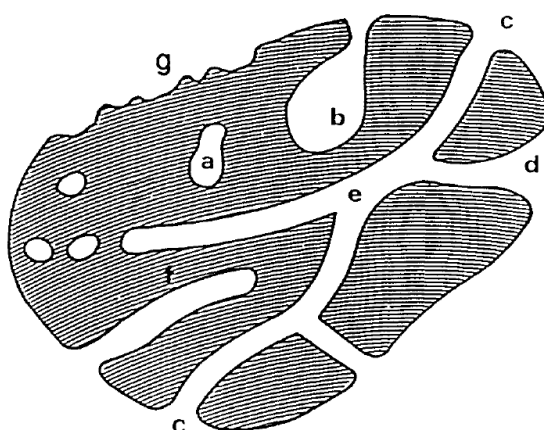


**Figure 2.1-3: Schematic cross section of a porous solid (taken from Rouquerol *et al.*, 1994)**

Rouquerol *et al.* (1994) defines porosity as the ratio of pore volume to apparent volume of a particle. Equation 2.1-1 illustrates this where  $V_p$  is the pore volume and  $V$  is the apparent volume.

$$Porosity = \frac{V_p}{V} \quad 2.1-1$$

The volumes stated above are related to density. Three different densities of a porous medium can be defined (Rouquerol *et al.*, 1994):

- True density: The density of material without any pores or external voids,
- Apparent density: The density of a material including inaccessible (closed) pores,
- Bulk density: The density of a material including all pores and external voids.

### Porosity measurement techniques

Hogekamp & Pohl (2003) states clearly in their study on porosity measurement techniques that porosity associated with agglomerated particles is as important as any other physical property such as particle size, however, it is far more difficult to measure and thus control. The control of a quality variable is dependent on the technique with which it is measured and the accuracy thereof (Hogekamp & Pohl, 2003).

The porosity of the product made in a fluidised bed granulator is often a critical variable in determining whether the particle is of good quality, for example explosive ammonium nitrate particles need to be porous to create ammonium nitrate fuel oil (ANFO) (Lotspeich & Petr, 2015).

The use of various experimental techniques to characterise porous material stemmed from the variety and the complexity of these materials. The refinement and expansion of these techniques have been extensively studied. Zou & Malzbender (2016) lists the core techniques that they used in their studies; also noting that each method has its own advantages and limitations since they all depend on different physical principles.

Liquid absorption, also referred to as oil absorption, is based on the volume of liquid that can absorb into the particle. The volume of liquid being absorbed into the particle is equal to the pores while the total volume is equal to the pore and particle volume combined.

The Brunauer-Emmet-Teller (BET) method is based on gas adsorption. In this method a gas, mostly nitrogen or carbon dioxide, is adsorbed onto the particle as well as a pore to determine a

surface area of the particle. A monolayer can form on the particle, indicating the surface area, as well as a multi-layer, indicating the volume inside the pores (Mel'gunov & Ayupov, 2017).

Mercury porosimetry is a method in which mercury is forced into pores of a material and the diameter of the pores is determined with the pressure needed to force the mercury into the pores. Porosity of a material can be determined using both mercury porosimetry and helium pycnometry. Mercury porosimetry calculates a pore volume by use of the volume of mercury displaced during the experiments while helium pycnometry calculates a true density of the sample (Mahajan & Walker, 1978).

Image analysis is considered an advanced method of determining porosity, however requires powerful computer processing as well as state of the art software for analysis. Images must be of extreme high quality to be able to distinguish between pores and solids (Andreola *et al.*, 2000). Image analysis for the calculation of porosity is also dependant on the contrast of the photo, to be able to separate 'white' pores against 'black' particles. Image analysis is also used as a way of verifying the validity of a technique such as liquid absorption. If the liquid being absorbed, is coloured, image analysis can determine whether the liquid fully absorbed into the particle or only partially due to some external factors.

#### *Oil absorption*

Oil absorption is the method currently being used at the industry partner to determine porosity of porous granular ammonium nitrate (PGAN) particles. The method was adapted in-house for the specific application. The full description of the method is given in Section 3.1.1.1.

The oil absorption method is based on liquid or dye penetration into the pores of a granule. Maria (as cited by Kok, 2016) explains that the liquid absorbed into the particles is equal to the volume of pores in the articles.

Pecht *et al.* (1994) investigated dye penetration into ceramic fragments to assess the porosity of such fragments and concluded that dye penetration can be done under pressure or in atmospheric conditions. For the test in atmospheric conditions, the ceramic bodies are generally 5 – 20 mm in size and placed in an open-air chamber filled with 1 g fuchsine dye dissolved in 1 L of 50% ethyl alcohol. The particles stay in the chamber for 5 minutes after which they are rinsed and dried. To determine porosity, the ceramic bodies are broken and visually examined for dye penetration (Pecht *et al.*, 1994).

### Microscopy

Image analysis has become more desirable with the development of stronger computer processing capabilities (Andreola *et al.*, 2000). Optical microscopy as well as scanning electron microscopy (SEM) and transmission electron microscopy (TEM) technologies make it possible to quantify morphological aspects of pores as well as size and shape factors of the particle. The images used to determine porosity must be good resolution images with noteworthy contrast to distinguish between pores and particle. The porosity of a particle is defined as the ratio of pore areas to the total visible area of the particle (Andreola *et al.*, 2000).

Lotspeich and Petr (2015) used a Quant 600 SEM analyser to determine the difference between ammonium nitrate fertiliser prills and explosive grade prills. The images obtained (Figure 2.1-4) showed significant differences in the exposed surfaces of the two different particles.

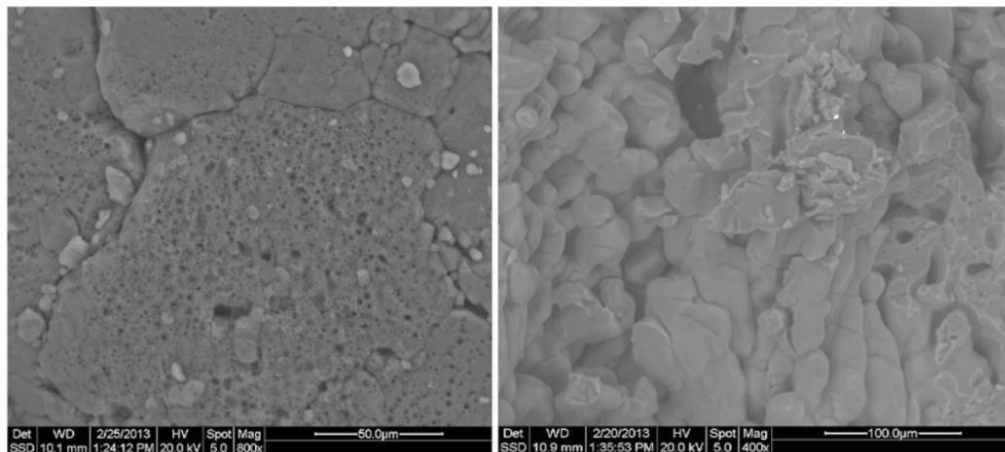


**Figure 2.1-4: SEM images of agriculture grade ammonium nitrate (left) and explosive grade ammonium nitrate (right) (taken from Lotspeich & Petr, 2015)**

The differences are clearly shown on SEM images, showing that the method of microscopy is viable in analysing properties of particles. The article found that explosive grade ammonium nitrate is bigger and more rigid on the surface than agriculture grade ammonium nitrate. Agriculture grade ammonium nitrate is porous, as with explosive grade ammonium nitrate, although the pores of agriculture grade ammonium nitrate do not extend to the surface.

#### 2.1.5.2 Sphericity

Particles with well-known shapes can easily be described by a few parameters, however irregularly shaped particles are difficult to describe. Irregular-shaped particles must always be described by more than one dimension (Rhodes, 2008:1). The dimensions that are used to

describe these shapes depend on which dimensions can be measured as well as the use of the measurable dimensions.

Ulusoy & Kursun (2011) did work on a Malvern Morphologi G3 Software analyser. The analyser captures two dimensional (2D) images of three dimensional (3D) particles and then calculates certain size and shape variables from the 2D images.

Riley *et al.* (2003) describes how size and shape factors are calculated through image analysis software. The sphericity of particles is calculated using Equation 2.1-2:

$$Sphericity = \frac{4\pi \times Area}{Perimeter^2} \quad 2.1-2$$

Here, the area is defined as the sum of the pixels in the particle image while perimeter is the sum of the pixels along the particle's boundaries.

#### 2.1.6 Fluidised bed control

The control of a fluidised bed granulator can be complicated since not only one process manufacturing mechanism is involved. Wetting, drying and mixing takes place simultaneously, causing many variables to affect one another. These variables and interactions between them directly affect the growth mechanism and thus the quality of the product (Hemati *et al.*, 2003; Srinivasakannan & Balasubramanian, 2003).

Burggraeve *et al.* (2013) divides the control of fluidised bed granulators into three categories, namely black-box, white-box and grey-box control. Black-box models use actual plant data to fit a function to the data. The function is fitted to get the best model for the data. The method is favourable as a model can be obtained quickly and then used to optimise the process control. A white-box approach is based on physical and chemical laws. The model depends on conservation principles and constitutive relations. Compared to the black-box approach, this model would take more time but have more flexibility. The grey-box approach is a combination of the white- and black-box approach. Experimental data and fundamental knowledge is combined to create a model. This method is mostly used in process system modelling.

Watano *et al.* (cited by Burggraeve *et al.*, 2013) investigated the use of fuzzy logic to control a fluidised bed granulator. The fuzzy logic was adequate to control moisture content, however not at all operating conditions.

Artificial intelligence such artificial neural networks (ANN), results in better prediction and generalisation compared to conventional methods, for the control of fluidised bed granulators. ANNs learn from examples, such as historical data, to be able to predict future values from experience (Burggraeve *et al.*, 2013).

#### **2.1.6.1 Soft-sensor technology**

Abeykoon (2018) defines soft-sensor (also known as inferential estimator or virtual sensor) technology as software based sensors that process one or more measurement simultaneously. Soft-sensors are used as an inference control measurement technique.

Soft-sensor technology is used when key process measurements can't be measured due to limitations in the process. Limitations preventing monitoring of the measurement of parameters include, but is not limited to, the inability to use physical sensors in the process instruments, physical sensors that could have disruptive effects on the process, fragility of physical sensors and complexity of processes preventing the use of physical sensors. In these cases, soft sensing is used to improve the monitoring and modelling of processes, and thus is used in advanced process control.

Soft-sensor technology is divided into two types of sensors:

1. Model-driven soft sensor
2. Data-driven soft sensor

A model-driven soft sensor is when the parameter that can't be measured on-line, is predicted using a model developed from first principles. A data-driven soft sensor is when the parameter is predicted using historical processing plant data. The data-driven approach is popular in the processing industry as it is a good representation of real process dynamics (Abeykoon, 2018).

Abeykoon (2018) lists the advantages of soft-sensing technology in the processing industry:

- Soft sensors are easy to implement on existing hardware
- Maintenance and updating of the system is easy as required
- Soft sensors are very useful when physical sensors can't be implemented
- Soft sensors have the possibility of giving real-time estimations for parameters otherwise unknown
- Soft sensors can be a economical alternative for expensive online analysers

Soft-sensing technology also bring forth a few challenges in the industry (Abeykoon, 2018):

- The design, optimisation and implementation of a soft sensor requires considerable expert knowledge about the industrial process where the sensor will be used
- Data-driven soft sensor can only perform as good as the quality of training or validation data used to develop the model used for the soft sensor
- Data-driven soft sensors are usually application specific and quite limited, depending on the data being used for training or validation

## 2.2 Artificial neural network

The human brain consists of millions of interconnected biological neurons, containing nuclei, dendrites, and axons as schematically represented in Figure 2.2-1. The axon, which is the neuron output, uses synapses to connect to other neurons. The input values from the axon cause the synapses to produce a chemical reaction which ‘fires’ if the sum of all synaptic reactions is sufficiently large. The complexity of this sensory neurons is what gives humans the ability to learn, reason, think and remember (Burns, 2001:347; Cirstea *et al.*, 2002:56).

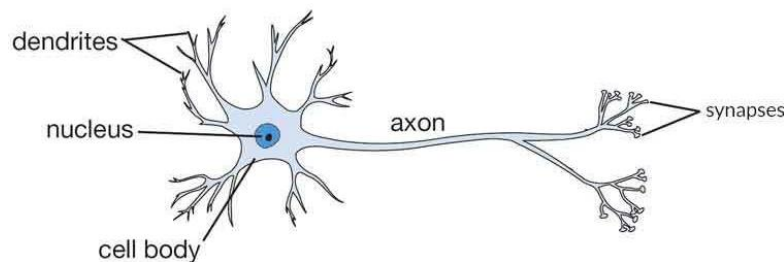


**Figure 2.2-1: Biological neuron**

Artificial neural networks (ANN) originated from an attempt to emulate the human brain's neural system to create a system that can also learn and remember from examples, and thus predict future (unseen) examples (Cirstea *et al.*, 2002:56; Zhang *et al.*, 1998). This concept falls in the domain of artificial intelligence, which in the broad field of intelligent control consists of expert systems, fuzzy logic and ANNs. Cirstea *et al.* (2002:56) classifies ANNs as the most powerful method used in artificial intelligence. The application of ANNs occur in numerous fields of the scientific community including forecasting, pattern recognition, financial modelling, control problems, to name only a few (Abraham, 2004; Zhang *et al.*, 1998).

ANNs consist of neurons, which are processing units, interconnected through weights (Dongare *et al.*, 2012; Ronco & Gawthrop, 1997). Neurons have one of two functions in ANNs, namely

aggregation and generation. The neurons aggregate its input from all other neurons as well as the environment of the neural network (such as biases). The output is then generated from these input values. Weights connect neurons in different layers to each other. Each neuron (or node) receives information from the various nodes in the previous layer through weights, while projecting information to various nodes in the following layer. The layout of a typical multilayer ANN is shown in Figure 2.2-2.

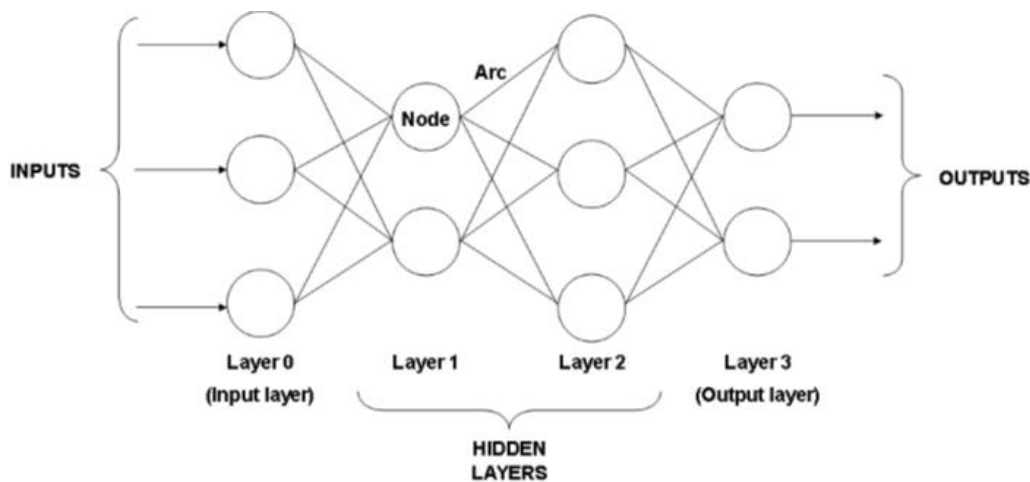


**Figure 2.2-2: Layout of a neural network example (adapted from Dongare *et al.*, 2012)**

One node is not very powerful on its own, as it generates a scalar output with only one numerical value. Several nodes constructed in an appropriate way cause the power of the system to emerge; to enable it to learn from historical data by capturing functional relationships (Azadi & Karimi-Jashni, 2016; Dongare *et al.*, 2012). The number of nodes and layers in the neural networks is strongly connected to the problem at hand and usually chosen using trial and error (Azadi & Karimi-Jashni, 2016; Dongare *et al.*, 2012; Satish & Setty, 2004), although approaches in finding an appropriate topology do exist (Zhang *et al.*, 1998). These approaches are complex and difficult to implement and does not guarantee an optimal neural network.

The first step in creating an ANN for any specific purpose, is to train the ANN. The training of the ANN is the process of obtaining appropriate weights to be able to predict the correct output for any given set of input values. The values of the weights are then stored in the arcs, which links different nodes together (Zhang *et al.*, 1998). The training is done with a sub-set of the data obtained for generating the ANN. The total data set is divided into training data and test (or validation) data; the training set is chosen to represent the likelihood of each outcome whereas the test set represents the population (Klimasauskas, 1991). As mentioned before, the training data is used to obtain the weights of the ANN, while the test data measure the ANN's ability to

predict unseen data. The training process starts with examples entering the ANN as inputs. The first hidden layer node values are calculated by weighing each input value and accumulating them at each node (Figure 2.2-3). This is achieved with the use of a specific activation function. The activation values of the nodes in the first hidden layer then becomes the inputs to the second hidden layer and the process is repeated in this layer. This continues until the output layer is reached (Zhang *et al.*, 1998). A training algorithm is used to change the values of weights until a specific performance measure is reached; usually the minimisation of an error as chosen by the operator.

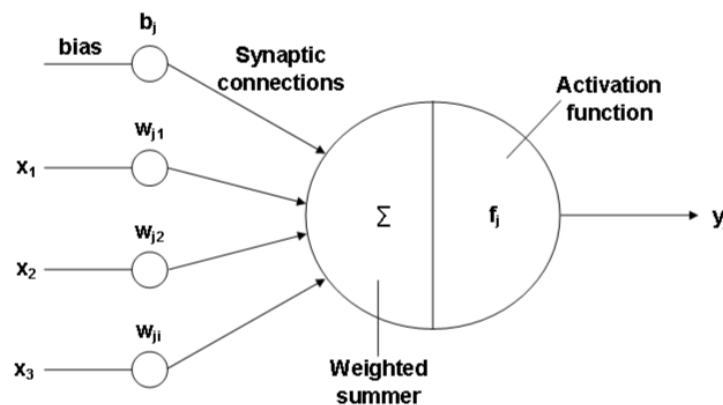


Figure 2.2-3: Example of a neural network node (adapted from Burns, 2001:348)

A key problem in obtaining an ANN for a specific application is choosing the right topology, i.e. the number of hidden layers as well as the number of nodes in each hidden layer (Abraham, 2004). Klimasauskas (1991) suggests starting with one hidden layer, as most problems are solvable with less complex topologies. The number of nodes in the hidden layer is chosen using a rule of thumb and is dependent on the number of training examples in the problem. A general rule-of-thumb is that there should be about five times more training examples than there are weights (Klimasauskas, 1991). In this study, the rule-of-thumb was not used as different topologies were investigated, including topologies that do not necessarily obey the rule-of-thumb.

A major attraction for the use of ANNs are that they don't require the plant data to be linear or non-linear, as long as the training data set covers the whole spectrum of what the plant is capable of, i.e. the minimum and maximum values of all inputs and outputs must be included in the training data set (Abraham, 2004; Burggraeve, 2013; Burns, 2001:358). This means that an artificial neural network can interpolate very well but cannot extrapolate. Zhang *et al.* (1998) proposes advantages and disadvantages of ANNs. They state the disadvantages are limitations in the research rather than problems that cannot be solved.

Several advantages of ANNs are listed below:

- ANNs consist of unique characteristics that makes them suitable for forecasting a variety of problems. The characteristics include adaptability, non-linearity and the ability to map functions (Burns, 2001:358).
- ANNs offer an alternative approach to problems, other than the traditional linear methods.
- ANNs are considered general solutions since they are model-free and data-driven.

A few limitations of ANNs are listed below:

- There is no conclusive method of determining whether ANNs are better than classical methods.
- Many factors affect the performance of an ANN, however there is no definite way of investigating these factors. The trial-and-error method is often adopted by researchers, creating many inconsistencies in the literature.
- ANNs are application specific, meaning there will not exist a universal model that can predict well in all situations.
- The method of using an ANN is considered a black-box approach and thus the ANN does not reveal anything about the relationship between inputs and outputs.
- Over-fitting is a common, generally unsolved problem with ANNs.

The optimisation of an ANN is tedious, due to the trial-and-error method being used since there is no way of identifying network structures.

### **2.2.1 Types of artificial neural networks**

Cirstea *et al.* (2002:59) classifies neural networks as either feed-forward neural networks or recurrent networks. The distinct characteristic of a feed-forward neural network is that information is only sent from one layer to the following layer, and not back to the previous layer (Dongare *et al.*, 2012). The neurons in one specific layer also do not send information to any neurons in that same layer. Recurrent neural networks are divided into networks where neurons in the same layer send information to each other or networks where some outputs are used as inputs (Cirstea *et al.*, 2002:60)

#### **2.2.1.1 Feed-forward neural networks**

The modelling of a feed-forward neural network consists of three basic elements: a set of synaptic weights, a linear combiner to sum input values and an activation function (Zhang *et al.*, 1998).

Dongare *et al.* (2012) states that feed-forward neural networks rarely consist of more than four hidden layers, whereas Ronco & Gawthrop (1997) states two hidden layers are sufficient. Cirstea *et al.* (2002:64) suggests one hidden layer when using the back-propagation algorithm as training algorithm, provided the hidden layer has sufficient number of neurons to solve the problem.

Feed-forward neural networks are mostly used in applications where the output is just a function of the current inputs, as feed-forward ANNs have no memory of past inputs (Cirstea *et al.*, 2002:59). A well-known type of feed-forward neural network that uses the back-propagation algorithm as learning algorithm is the multi-layer perceptron (Burns, 2001; Zhang *et al.*, 1998). The multi-layer perceptron is a feed-forward neural network in which all hidden layers are fully connected to each other, as seen in Figure 2.2-2 (Ronco & Gawthrop, 1997).

### 2.2.1.2 Recurrent neural networks

Recurrent networks are divided into two classes, namely networks where outputs are also used as inputs and networks where neurons in one layer are in communication with each other. Examples of recurrent neural networks where outputs are used as inputs are real-time recurrent networks (discrete and continuous) and Hopfield networks. Cellular neural networks are an example of recurrent networks where neurons in the same layer share information (Cirstea *et al.*, 2002:60). Recurrent neural networks are mainly used when communication is needed between neurons in different layers, or when the output of the neural network influences the results of the neural network.

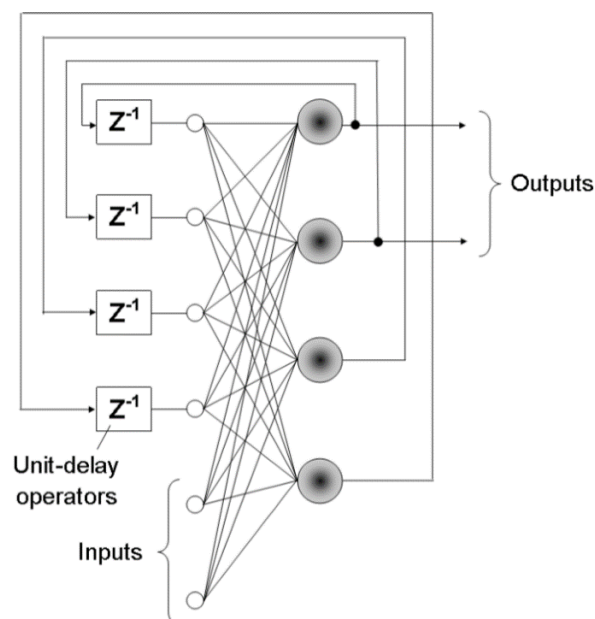


Figure 2.2-4: A real-time recurrent neural network example (adapted from Cirstea *et al.*, 2002)

Real-time recurrent neural networks (Figure 2.2-4) are used to model finite and infinite state machines. The concept of 'internal state' is implemented as a set of outputs, which is then used as future input values. The difference between discrete-time models and continuous-time models are that discrete-time models have a delay on the feedback of the outputs, whereas continuous-time models have low-pass filters. Cirstea *et al.* (2002:61) notes that discrete-time models are used for finite state machines and uses a step activation function, while continuous-time models are used in infinite state models with a sigmoidal activation function.

Hopfield neural networks contain only a single hidden layer in which each output is communicated back (as input) to all the neurons in that layer except the neuron from which it came, as can be seen in Figure 2.2-5 (Cirstea *et al.*, 2002:61). An 'energy function' is used to describe each state of a Hopfield network; the energy function decreases towards a local minimum which determines the evolution of the network's state. The network links each local minimum with a stable state, which is why Hopfield networks are ideal for optimisation problems. Hopfield networks are well known to be used for associative memory (the ability to recollect an example from a distorted version) applications as well as optimisation in power systems (Cirstea *et al.*, 2002:61; Zhang *et al.*, 1998).

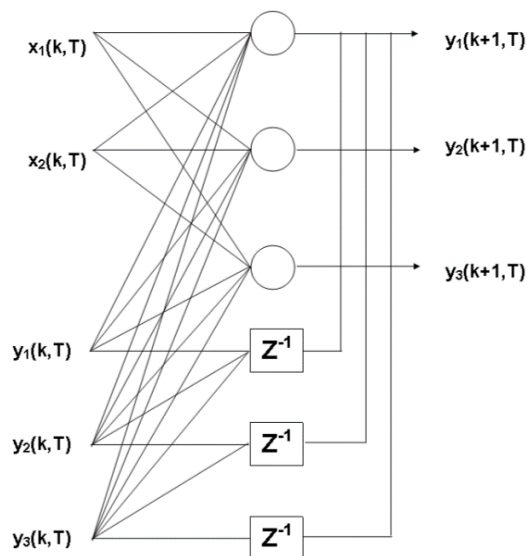


**Figure 2.2-5: Hopfield neural network example (adapted from Burns, 2001: 351)**

Cellular neural networks (Figure 2.2-6) are unique in that each neuron is only connected to the neurons close to it, thus only exchanging information with its neighbouring neurons (Cirstea *et al.*, 2002:60). Cellular neural networks are used mostly in image processing, artificial vision, biological system modelling and associative memories.

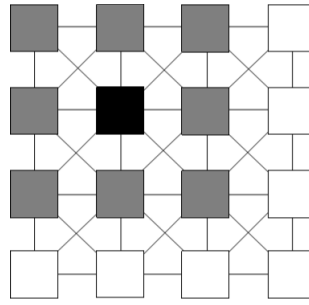


Figure 2.2-6: Cellular neural network example (adapted from Cirstea *et al.*, 2002)

The prediction of porosity and sphericity of the product of a fluidised bed granulator does not influence the next prediction and is also not so complex that communication between neurons in the hidden layers are necessary. For this study, only feed forward neural networks were tested.

### 2.2.2 Activation function

The inputs and outputs of nodes as well as the neural network itself, is correlated by using an activation/transfer function (Azadi & Karimi-Jashni, 2016; Zhang *et al.*, 1998). Azadi & Karimi-Jashni (2016) states that any differential function can theoretically act as an activation function, yet in practice it was found that there are only a few functions that are considered well-behaved. Chen and Chen (1995) detected set qualities that a function must have to be well-behaved. They state that the function must be bounded, monotonically increasing and differentiable. Both Azadi & Karimi-Jashni (2016) and Zhang *et al.* (1998) give three activation functions as used in the industry:

1. Log-sigmoid function
2. Hyperbolic tangent sigmoid function
3. Linear function

The activation function that is used depends on the application and Zhang *et al.* (1998) gives a set of heuristic rules when choosing an activation function. The log-sigmoid function is the most popular activation function, but a combination of activation functions in different layers are also used.

The sigmoid function (often referred to as the logistic function) is mathematically described by Equation 2.2-1:

$$f(x) = \frac{1}{1 + \exp(-x)} \quad 2.2-1$$

The function is illustrated in Figure 2.2-7 below:

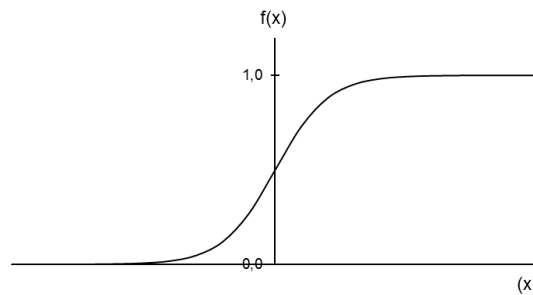


**Figure 2.2-7: Sigmoidal activation function (adapted from Burns, 2001:349)**

The saturation limits of a sigmoidal function are zero and one, which means the output varies between these values. Among the abovementioned activation functions, the sigmoid function is the most widespread choice (Zhang *et al.*, 1998). Klimasauskas (1991) suggests using the sigmoid activation function in situations that comprise learning about average behaviour such as classification problems.

The hyperbolic tangent function is described by Equation 2.2-2:

$$f(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \quad 2.2-2$$

The hyperbolic tangent function is shown in Figure 2.2-8 below:

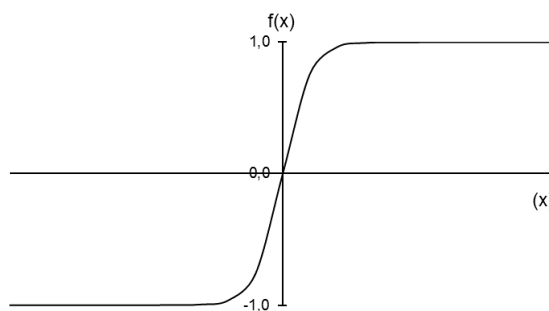


**Figure 2.2-8: Hyperbolic tangent activation function (adapted from Burns, 2001:349)**

The outputs of a neural network using the hyperbolic tangent function as activation function, vary between negative one and positive one. The hyperbolic tangent function is usually used in cases where the behaviour of the problem deviates from the average, such as forecasting problems (Klimasauskas, 1991).

The linear function is:

$$f(x) = x \quad 2.2-3$$

The linear function is given below in Figure 2.2-9:

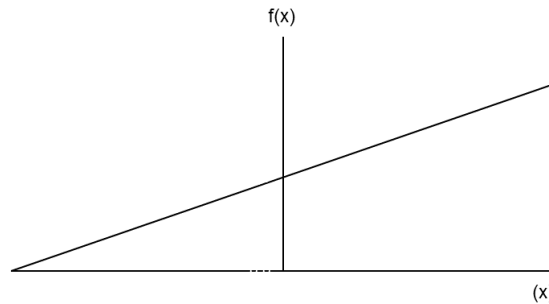


**Figure 2.2-9: Linear activation function (adapted from Burns, 2001:349)**

The linear activation function is mostly used as the activation function of the output layer, instead of being used in the hidden layers. The reason for this is that the output layer is more simplistic than the hidden layer(s) for most cases, and thus the linear activation function is sufficient for this layer (Zhang *et al.*, 1998).

Zhang *et al.*, 1998 states that most neural networks consist of a single activation function for all the nodes in a layer, although it is not unheard of to combine different activation functions for different nodes in the same or different layers. Zhang *et al.* (1998) also found that most researchers tend to use the sigmoid function in hidden layers, but there is no trend as to which activation function should be used in the output layer.

For this study, the hyperbolic tangent function will be used as activation function due to the ability to forecast complicated or random problems with no statistical significant relation between input and output variables. The study will, however, compare the hyperbolic tangent function with the other activation functions, as seen in Section 5.2.2.3.

### 2.2.3 Training algorithm

The training algorithm updates the weights in the neural network to reduce the error between obtained outputs and experimental data outputs (Azadi & Karimi-Jashni, 2016; Satish & Setty, 2004). Azadi & Karimi-Jashni (2016) names five of the most common training algorithm but warns that picking the appropriate training algorithm for a specific application in advance is proven challenging.

The training algorithms examined by Azadi & Karimi-Jashni (2016) include:

1. Levenberg-Marquardt
2. Bayesian regulation
3. Gradient descent with momentum
4. Resilient back-propagation
5. Scaled conjugate gradient

Rämet *et al.* (1996) used a modified back-propagation algorithm known as the Weigend weight eliminator (Weigend *et al.* cited by Rämet *et al.*, 1996). Zhang *et al.* (1998) states that the most commonly used training algorithm is the back-propagation algorithm, in some cases with minor modifications.

Zhang *et al.* (1998) defines the training of a neural network as the process of iteratively adjusting the weight values of the network to eventually obtain an overall minimum mean error or minimum total squared error between actual and desired outputs. The learning of the neural network is closely related to the training algorithm and can be divided into three categories, namely supervised learning, unsupervised learning and reinforced learning (Burns, 2001; Cirstea *et al.*, 2002:62; Dongare *et al.*, 2012). Supervised learning consists of a “teacher” to indicate whether the system is performing adequately, whereas unsupervised learning has no “teacher”. Unsupervised learning relies on data and the environment to guide the learning process. Reinforced learning is a combination of supervised and unsupervised learning. The neural network learns from its environment’s feedback, grading a response as good or bad and adjusting the weights accordingly. Supervised learning is considered more adequate for control problems since the input-output relationship is better defined by a ‘supervisor’ (Cirstea *et al.*, 2002:62).

Cirstea *et al.* (2002:62) classifies the training process as either constructive (Voronoi diagram algorithm and evolutionary neural networks) or non-constructive (back-propagation and particle swarm optimisation algorithms). A non-constructive training process consists of a training algorithm that only adjusts the weights and threshold levels to optimise the neural network. A constructive training process uses a training algorithm that also adjusts the architecture (or topology) of the neural network to acquire the optimum neural network.

The architecture of a neural network trained with a non-constructive training algorithm can be determined manually by the operator. Klimasauskas (1991) offers two tactics to the optimisation of the architecture, namely constructive and destructive. Constructive optimisation consists of starting with a neural network with no architecture, thus connecting the inputs directly to the

outputs. The network is then trained until the error is stable, producing the first set of weights. The weights are fixed while a single hidden unit is added to the structure. The process is repeated until the error does not change with the addition of hidden units. The destructive optimisation method consists of a similar basis with the difference that the operator starts with many hidden units, decreasing the units to a point where the error does not change or increase. If the error increases after a hidden node was removed, the hidden node is again added to obtain the optimum architecture.

Back-propagation networks are a non-constructive form of feed-forward neural networks where information is signalled forward while the error is broadcasted backwards. Back-propagation is learned using supervised learning, meaning the network is fed with examples (such as historical data) of inputs and outputs (Burns, 2001:351; Cirstea *et al.*, 2002:62). An error is calculated between the outputs in the training data set and outputs of the neural network and then the network weights are adjusted accordingly. Back propagation functions through reducing this error until the ANN has 'learned' to properly predict the data points (Dongare *et al.*, 2012; Satish & Setty, 2004).

Azadi & Karimi-Jashni (2016) states that the back-propagation algorithm is very sensitive to initial weight values, thus the values must be allocated with care. If the chosen initial weight values are inappropriate, the problem could get trapped in a local minima error.

Zhang *et al.* (1998) confirms the problem of slow convergence and lack of robustness in the back-propagation algorithm due to a poorly chosen learning rate. The learning rate is a step size that determines the rate at which weights change during the training process. The chosen learning rate could either be too small, slowing the learning process, or too large, causing oscillating behaviour in the weight space (Cirstea *et al.*, 2002:65).

Rumelhart *et al.* (cited by Zhang *et al.*, 1998) suggests adding a momentum parameter to the learning process. The momentum parameter can be used in conjunction with larger learning rates, thus decreasing process times, while still eliminating oscillations. The momentum factor does this by changing each weight in the same direction as the previous weight (Burns, 2001; Zhang *et al.*, 1998). The methodology behind using momentum, is that each weight adjustment is supplemented by a fraction of the previous weight adjustment (Cirstea *et al.*, 2002:66). The learning rate and momentum are both values between 0 and 1, with momentum more specifically being a value between 0.1 and 0.8, usually chosen simultaneously through experimentation. Tang *et al.* (cited by Zhang *et al.*, 1998) suggests using higher learning rates for low level of complexity data, while more complex data should be paired with a low learning rate and high value of momentum.

The back-propagation algorithm has been adapted to create a few new methods, each with their own appeal (Zhang *et al.*, 1998). Cirstea *et al.* (2002:65-66) lists a few ways in which the original back-propagation algorithm has been adapted:

1. The learning rate changes after each training cycle, usually starting with a large learning rate, to speed up the training, and gradually decreasing the learning rate to avoid oscillating behaviour at the end of the learning process
2. Each weight has its own learning rate, since one learning rate might not be suitable for all variables within the network
3. Each weight has its own learning rate which also varies from one training cycle to the next
4. If the error oscillates for a few iterations after one another, the learning rate of that weight is decreased
5. The momentum method (as described above) is employed

The most widely used adapted back-propagation algorithms are the quasi-Newton algorithm, the Levenberg-Marquardt algorithm and the conjugate gradient method. The conjugate gradient method employs a search direction in each iteration, to adjust each new weight value proportional to its previous values. The method converges very slowly and is computationally expensive due to the Hessian matrix that needs to be stored in memory for future calculations. The quasi-Newton training algorithm addresses this problem by assuming constant, positive values for the Hessian matrix and only tracking the inverse of the Hessian matrix without calculating and saving the matrix. The convergence of the quasi-Newton training algorithm is faster than that of the conjugate gradient algorithm, however, the algorithm tends to oscillate within a minimum due to higher learning rates. The Levenberg-Marquardt training algorithm was developed to address both the problems of the abovementioned training algorithms and can thus be a combination of the two, as explained in Section 2.2.3.1 below (Abraham, 2004).

The Bayesian regularisation training algorithm is also a widely used adaptation of the back-propagation training algorithm. The Bayesian regularisation algorithm is a relatively new adaption, compared to the Levenberg-Marquardt training algorithm. The main difference in the Bayesian regularisation algorithm is that the historical sum of weight values is also featured in the algorithm, as discussed in Section 2.2.3.2 below (Payal *et al.*, 2015).

In this study, the Levenberg-Marquardt and Bayesian regularisation training algorithms were used as they are the most commonly used training algorithms due to their separate advantages (Aqil *et al.*, 2007; Chaipimonplin, 2016; Kayri, 2016; Samadian *et al.*, 2016). Since it is a combination of two algorithms, the Levenberg-Marquardt training algorithm has an increased learning speed,

however it is prone to over-fitting. The Bayesian regularisation training algorithm solves this problem by minimising the sum of squared weights. Another advantage of the Bayesian regularisation training algorithm is that it does not require a validation dataset, which also helps to improve training of the neural network (Bui *et al.*, 2012; Chaipimonplin, 2016; Kayri, 2016).

### 2.2.3.1 Levenberg-Marquardt algorithm

The Levenberg-Marquardt algorithm was autonomously developed by Kenneth Levenberg (1944) and Donald W. Marquardt (1963). The algorithm is a combination of the quasi-Newton method and the conjugate gradient method, combining their advantages and eliminating their limitations (Marquardt, 1963; Yu & Wilamowski, 2011). The Levenberg-Marquardt algorithm can converge from most initial guess values without the guess values having to be close to real values, while still converging at a relatively good speed. The two limitations of the Levenberg-Marquardt algorithm are that it is limited to small-to-medium sized problems and it requires larger computational storage as the Jacobian matrix must be stored during calculations (Yu & Wilamowski, 2011).

Madsen *et al.* (2004) calls the Levenberg-Marquardt algorithm a damped quasi-Newton method, since it introduces a damping factor that affects the size as well as the direction of a given step. A large damping parameter is used in the case where an iteration is far from the solution while a small damping parameter is used in the final stages of a solution.

Figure 2.2-10 shows a block diagram of the training process using the Levenberg-Marquardt algorithm. The training process is summarised as follows:

1. Evaluate the error with random weight ( $w_k$ ) values
2. Update the weights using the formula for  $w_{k+1}$
3. Evaluate the new error
4. Compare the new error with the initial error
  - a. If the error increased, reset the weight vector and increase the damping parameter ( $\mu$ ) by a chosen factor; repeat from step 2
  - b. If the error decreased, keep the changes and decrease the damping parameter
5. Continue until the error value is sufficiently small

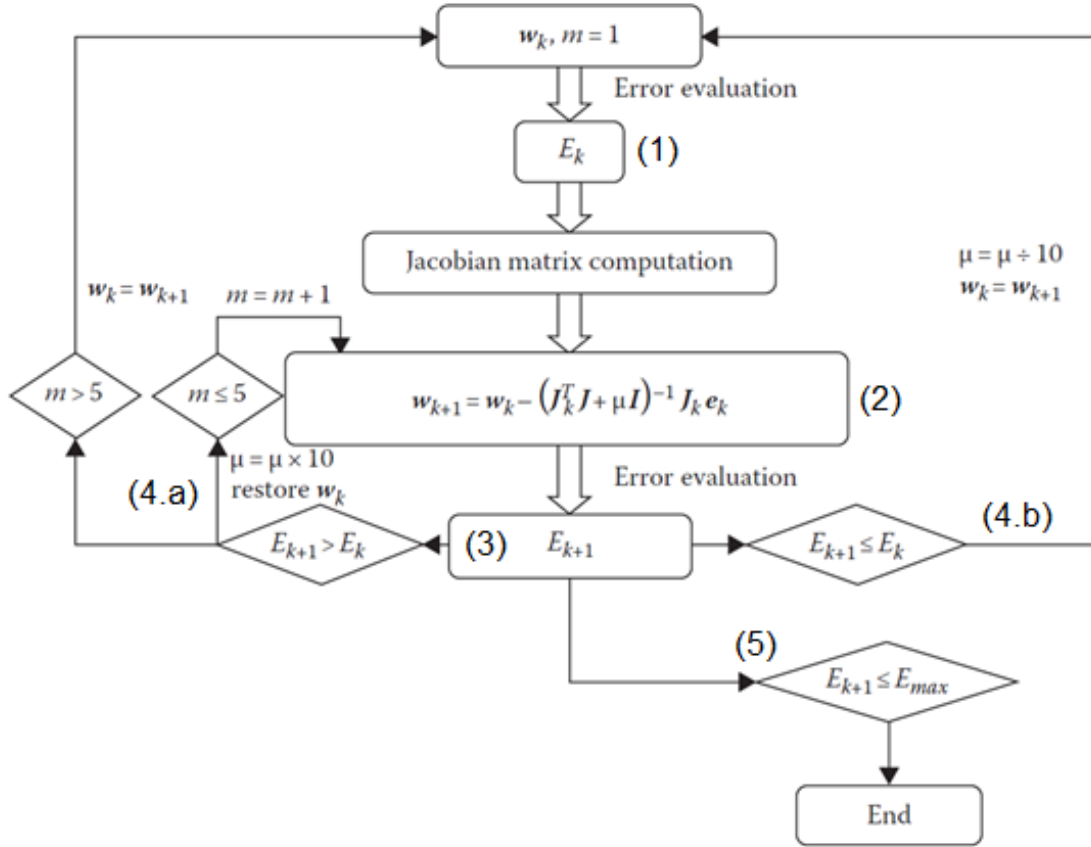


Figure 2.2-10: Levenberg-Marquardt algorithm (taken from Yu & Wilamowski, 2011)

An important parameter in the computation of the Levenberg-Marquardt training algorithm is the Jacobian matrix that must be calculated. The matrix consists of values of the first derivatives of the network errors with respect to the weights and biases of the network (Demuth & Beale, 2004).

### 2.2.3.2 Bayesian regularisation method

A major problem that occurs in neural network training using well-known back-propagation training algorithms is over-fitting. Over-fitting results in a very small error in early stages of training, however, when the neural network is presented with unseen data, a significantly large error is present. The Bayesian regularisation method pose a solution to this problem (Kaur & Salaria, 2013; Kasiviswanathan *et al.*, 2016; Khafaf & El-hag, 2018; Payal *et al.*, 2013).

Kayri (2016) explains that the Bayesian regularisation method performs better than the Levenberg-Marquardt algorithm due to its ability to reveal complex relationships in a problem. The Bayesian regularisation method is in most aspects like the Levenberg-Marquardt algorithm, particularly in the way weights and biases are updated, however a different objective function is used (Kaur & Salaria, 2013). The objective function contains a residual sum of squares and sum

of squared weights, which is used to minimise the estimation errors. The Bayesian regularisation method achieves good generalisation due to this improved objective function (Kayri, 2016). Khafaf & El-Hag (2018) also explains that the Bayesian regularisation training algorithm obtains better generalised networks when compared to the Levenberg-Marquardt training algorithm. The Bayesian regularisation training algorithm continues to try to find an optimum combination of weights and biases even after training has stopped, which is why it is often considered a better training algorithm than others.

The iterative process of the Bayesian regularisation training algorithm is given as follows (Bui *et al.*, 2012; Khafaf & El-hag, 2018):

- (i) Choose initial weight values as well as initial values for the objective function variables; the objective function variables for future steps are obtained from previous values, so only the initial values need to be chosen
- (ii) The objective function is minimised using the Levenberg-Marquardt optimisation equation
- (iii) New objective function variables are estimated for the training step
- (iv) New weight values are calculated
- (v) Steps 2 to 4 are repeated until convergence of the network

The objective function variables are what separate the Bayesian regularisation training algorithm from the Levenberg-Marquardt algorithm. The variables include the previous weight values in the optimisation of the current weight values.

#### **2.2.4 Performance evaluation**

The performance of an ANN can be measured in many ways such as training or modelling time, however, the most important measurement of performance is how well the neural network predicts unseen data, in other words, the difference between network outputs and actual output values (Satish & Setty, 2004; Zhang *et al.*, 1998).

A major problem that occurs in ANNs is overfitting. This is when the ANN performs very well during training but fails to predict accurate outputs of unseen problems (Azadi & Karimi-Jashni, 2016). Kocadağlı (cited by Azadi & Karimi-Jashni, 2016) suggests an early stopping technique to overcome the problem of overfitting.

There are a few ways to calculate this error, but the most widely used methods are:

1. Mean squared error:

$$MSE = \frac{\sum(e_t)^2}{N} \quad 2.2-4$$

2. Root mean squared error:

$$RMSE = \sqrt{MSE} \quad 2.2-5$$

3. Mean absolute error:

$$MAE = \frac{1}{N} \sum \left| \frac{e_t}{y_t} \right| \quad 2.2-6$$

4. Mean absolute percentage error:

$$MAPE = MAE \times 100 \quad 2.2-7$$

In the abovementioned formulas  $e_t$  is the difference between neural and actual outputs,  $y_t$  is the actual output value and  $N$  is the number of error terms. The mean-squared error and root mean-squared error have similar advantages and disadvantages since the calculations are derived from one another. The mean absolute error and mean absolute percentage error are also considered similar error measurements. The advantages and disadvantages of the abovementioned methods are given in Table 2.2-1 below (Azadi & Karimi-Jashni, 2016):

**Table 2.2-1: Advantages and disadvantages of error measurements**

| Error measurement | Advantages                                                                                                                                                                  | Disadvantages                                                                                                                                                   |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MSE/RMSE          | <ul style="list-style-type: none"> <li>• Measures uncertainty</li> <li>• Gives a quadratic loss function</li> </ul>                                                         | <ul style="list-style-type: none"> <li>• Emphasis is placed on larger errors more than on smaller ones</li> <li>• High sensitivity to erroneous data</li> </ul> |
| MAE/MAPE          | <ul style="list-style-type: none"> <li>• Intuitive, easy method since it is a linear measurement</li> <li>• Good method for determining the importance of errors</li> </ul> | <ul style="list-style-type: none"> <li>• Lacks statistical theory</li> <li>• Absolute measure</li> </ul>                                                        |

Payal *et al.* (2013) states that the default performance measurement for feed forward neural networks is the mean squared error calculation, which was used in this study.

An accurate performance evaluation consists of more than measurement, where one is usually an error measurement and the other is a measurement of correlation (Azadi & Karimi-Jashni, 2016). The most commonly used evaluation of correlation is the correlation coefficient, as given in Equation 2.2-8:

$$R = \left[ \left( \frac{1}{N} \right) \left( \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \right) \right] \quad 2.2-8$$

The quadratic form of the abovementioned equation is known as the coefficient of determination and is given in Equation 2.2-9 below:

$$R^2 = \left[ \left( \frac{1}{N} \right) \left( \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \right) \right]^2 \quad 2.2-9$$

In the abovementioned equations,  $x_i$  and  $y_i$  are the experimental and predicted output variables respectively,  $\bar{x}$  and  $\bar{y}$  are the averages of the experimental and predicted output variables,  $\sigma_x$  is the standard deviation of the experimental values,  $\sigma_y$  is the standard deviation of the predicted output variables and  $N$  is the number of values.

The correlation coefficient is a measurement of the strength and direction of the relationship between observed and predicted values. This is one of the biggest advantages of this performance evaluation. The disadvantages include that the results are only accurate if the relationship is linear and the value does not reveal the influence of different variables on one another (Azadi & Karimi-Jashni, 2016).

### 2.3 Chapter summary

Fluidised bed granulation has historically been known to be a process with several operational challenges, mainly due to the process being very difficult to optimise. The main purpose of a fluidised bed is granule growth, with two types of growth usually present in a fluidised bed, namely layering and agglomeration. The type of growth mechanism that dominates within a certain fluidised bed granulator determines the product characteristics and this depends mainly on

operating parameters. The effect of operating parameters on granulation can also be quite different for different applications. This makes the modelling of granulation especially challenging. In this study, two quality variables are of importance, namely porosity and sphericity. Porosity measurement techniques were investigated as a possibility for a better method of measurement. A big consequence of these operational challenges is that a fluidised bed granulator is baffling to control, especially using conventional methods. Soft-sensor technology and using artificial intelligence for the modelling of the process is suggested in most literature.

Artificial neural networks are advantageous in a complex process where the direct effect of operating parameters on quality variable are not easy to establish. The development and optimisation of artificial neural networks is not an exact science and is usually done in a 'trial-and-error' manner. Several decisions must be made in the development of a neural network for the estimation of a variable. These decisions include:

- The topology of the neural network
- The type of neural network being developed
- The activation function
- The training algorithm
- The performance evaluation used to assess the neural network

Each decision has either a set of rules, some rules-of-thumb or no rules and must be evaluated by trial-and-error.

## Chapter 3: Experimental procedure

This chapter covers the experimental procedures and experimental plan followed during the execution of this study.

Section 3.1 starts with a description of the two selectivity variables used in classifying the final ammonium nitrate granules and the ways in which each are determined. Following this, Section 3.2 consists of description of the two phases of data accumulation for the training and validation of the artificial neural networks.

### 3.1 Selectivity variable measurement and validation

#### 3.1.1 Porosity measurement

Porosity must be accurately measured to develop a control strategy for the porosity of ammonium nitrate (AN) granules. The current method used at the industry partner for determining the porosity of AN granules, is the oil absorption method. The oil absorption method is tested for validity against a method based on particle and bulk density; this method has been named the optical porosity measurement method.

A total of 8 samples, obtained from the porous granular ammonium nitrate (PGAN) plant were tested using the different measurement techniques. The samples were screened to a size range of 2,3 mm to 2,8 mm to comply with the customised operating procedure, and to ensure that the outer surface area of the particles do not largely influence measurements. The outer surface area of the particle can mainly influence the oil absorption technique, as the oil also adheres to the outer surface of the particle. This influences the measurement of oil absorbed, since larger particles will have more oil adsorbing on the outer surface compared to smaller particles. The smaller size fraction of particles also ensure that the optical porosity measurement method is accurate, since the  $d_{50}$  value that is used in the method, is more believable for a smaller fraction of particles.

##### 3.1.1.1 Oil absorption

###### Sample preparation

A smaller sample of 2.1 g was removed from each of the 8 samples from the plant. The samples were placed in separate, previously weighed containers for analysis. Samples number 1, 5 and 8 were repeated to ensure the method's repeatability.

### Equipment and procedure

A schematic representation of the oil absorption setup is shown in Figure 3.1-1.

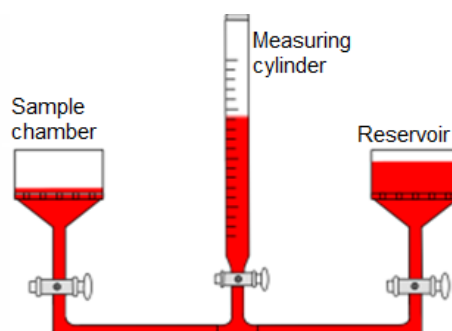


**Figure 3.1-1: Schematic representation of the oil absorption apparatus (taken from Kok, 2016:31)**

The first step in the oil absorption method, was to ensure the correct level of liquid in the measuring cylinder. The level should always be at 20.5 ml before each run to ensure constant conditions for repeatability. The measuring cylinder was filled with liquid from the reservoir. The next step is to add the 2.1 g sample into the sample chamber on top of a glass sponge found in the sample chamber. The valve below the sample chamber was opened so that the glass sponge can absorb liquid from the measuring cylinder. The granules were left in the sample chamber for 10 minutes to absorb liquid through the sponge. After 10 minutes, the valve below the sample chamber was closed and the sample was removed to be placed back into the original container. The last step was to weigh the sample with the liquid to determine how much liquid was absorbed. The weight was recorded for use in future calculations.

### Experimental calculations

Two different calculations were done on the data obtained from oil absorption. The first is a porosity percentage based on the mass of absorbed liquid, which is the method currently being used by the industrial partner, whereas the other method provided an estimate of a true porosity percentage based on the volume of absorbed liquid.

The porosity percentage based on mass was calculated using Equation 3.1-1:

$$\text{Porosity \% (mass)} = \frac{m_{\text{liq absorbed}} [g]}{m_{\text{liq absorbed}} [g] + m_{\text{granules}} [g]} \times 100 \quad 3.1-1$$

The porosity percentage by volume was calculated using Equation 3.1-2, where the volume of the pores was calculated using Equation 3.1-3 and the volume of the solid was calculated using

Equation 3.1-4. The predetermined densities used in each equation are summarised as  $\rho_{liquid} = 789 \text{ kg/m}^3$  and  $\rho_{particle} = 1725 \text{ kg/m}^3$  respectively.

$$Porosity \% (volume) = \frac{V_{pore} [m^3]}{V_{granules} [m^3]} \quad 3.1-2$$

$$V_{pore} = \frac{m_{liq \text{ absorbed}} [kg]}{\rho_{liquid} [kg/m^3]} \quad 3.1-3$$

$$V_{granules} = \frac{m_{granules} [kg]}{\rho_{particle} [kg/m^3]} \quad 3.1-4$$

### 3.1.1.2 Optical porosity measurement method

The optical porosity measurement method was developed as an alternative method for determining a volume-based granule porosity. The method relies on bulk density as well as a density based on particle sphericity, particle count and mass of the granules.

#### Sample preparation

The sample was weighed to be close to 100 g of granules. The mass was recorded for future calculations.

#### Equipment and procedure

A Haver CPA 2-1 particle size analyser was used to obtain a mean diameter, a mean sphericity and particle count of each sample. Granules were fed to the analyser where it is photographed using laser photography and these photographs were then analysed by a computer connected to the analyser. The biggest drawback of the analyser is the images are captured in 2D and not 3D.

#### Experimental calculations

The volume of a single particle (Equation 3.1-5) was calculated using the mean diameter ( $d_{50}$ ) from the particle size analyser. After this, the volume of the sample (Equation 3.1-6) was calculated using the particle count and the mean circularity from the Malvern analyser.

$$V_{particle} = \frac{4}{3}\pi \left( \frac{d_{50} [m]}{2} \right)^3 \quad 3.1-5$$

$$V_{sample} = \text{mean circularity} \times \text{particle count} \times V_{particle} [m^3] \quad 3.1-6$$

The porosity (Equation 3.1-7) was finally calculated using bulk and particle density, where bulk density was obtained using Equation 3.1-8 and particle density is fixed at  $\rho_{particle} = 1725 \text{ kg/m}^3$ .

$$\text{Porosity \% (volume)} = \left(1 - \frac{\rho_{bulk}}{\rho_{particle}}\right) \times 100 \quad 3.1-7$$

$$\rho_{bulk} = \frac{m_{granules} [kg]}{V_{sample} [m^3]} \quad 3.1-8$$

### 3.1.1.3 Comparison of methods

The porosity measurement techniques are compared based on correlation and error between values obtained with each method.

The coefficient of determination ( $R^2$ ) is obtained using a linear regression plot in Excel. The value is calculated using Equation 3.1-9 (adapted from Azadi & Karimi-Jashni, 2016).

$$R^2 = \left[ \left( \frac{1}{N} \right) \left( \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \right) \right]^2 \quad 3.1-9$$

In the equation above,  $x_i$  is the results of one method while  $y_i$  is the results of the comparing method. The variables  $\bar{x}$  and  $\bar{y}$  are average values for each method while  $\sigma_x$  and  $\sigma_y$  are the standard deviations for each method's results.

The error value is calculated as the mean absolute error (MAE) shown in Equation 3.1-10 (adapted from Azadi & Karimi-Jashni, 2016).

$$MAE = \frac{\sum |x_i - y_i|}{N} \quad 3.1-10$$

### 3.1.2 Sphericity measurement and the necessity of repeatability

Sphericity is an important quality variable in the growth of ammonium nitrate granules as it influences the flowability of the granules. The Haver CPA 2-1 particle size analyser determines a mean circularity of the granules, which is dependent on the physical orientation of the granule as it passes the detecting laser sensor. The particle size analyser works on the assumption that the 2D image taken, also represents the 3D variables of the granule. A particle that is elongated, but

falls with the top facing to the lenses, could thus be interpreted as being spherical and not cylindrical.

The purpose of this experiment is to determine whether the particle size analyser has adequate repeatability on the 2D mean circularity determination, so that this measure can be interpreted as a measure of the sphericity of a particle.

### 3.1.2.1 Experimental setup

#### Sample preparation

For the sphericity experiments, only 5 of the 8 samples were tested. The samples were not screened beforehand to ensure different particle sizes in the sample. Approximately 100 g of each sample was taken.

#### Equipment and procedure

The particle size analyser was used to obtain the mean circularity of each sample. Each sample was analysed three consecutive times by the same person in the same conditions, to ensure a representable repeatability. The samples were analysed, and mean circularity was recorded.

#### Experimental calculations

Repeatability of an analyser is based on its ability to yield the same results several times, provided the conditions are the same. The repeatability is directly related to the standard deviation from the mean. The mean (Equation 3.1-11) is calculated first, after which the variance (Equation 3.1-12) and standard deviation (Equation 3.1-13) are calculated.

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad 3.1-11$$

$$\sigma^2 = \frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n - 1} \quad 3.1-12$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n - 1}} \quad 3.1-13$$

The error is calculated using Equation 3.1-14. A  $t_{n-1}$  of 4.303 was used for confidence interval of 95% and degrees of freedom of 2. The value was obtained from Figure A.2- in Appendix A.2. The percentage error (Equation 3.1-15) is calculated as the fraction of the error to the mean.

$$Error = \frac{\sigma}{\sqrt{n}} \times t_{n-1} \quad 3.1-14$$

$$\%Error = \frac{Error}{\bar{x}} \quad 3.1-15$$

## 3.2 Data collection for the predictive model

### 3.2.1 Phase 1 data collection and preparation

Phase 1 of the model development consisted of using historical plant data to test different artificial neural network (ANN) configurations to obtain the most appropriate data set variables upon which phase 2 will be based. The data was very raw and had to be filtered several times before it could be used in the ANN testing. Three different datasets were tested to establish how data for phase 2 will be acquired.

Historical data was obtained from the PGAN production facility. The data ranged from 03:00 on 1 January 2017 to 00:00 on 15 May 2017, with a one-minute sampling interval; however, there were major sections of the data that were unusable due to shut-downs and sensor failures. The data had to be filtered to remove unusable values following a meaningful statistical filtering process as described in the subsequent sections.

The data needed to be compiled before it could be filtered and then used. The compilation consisted of adding the output data, namely oil absorption values, to the input parameter data. Each oil absorption value was added at exactly its corresponding input parameters' time stamp, and this oil absorption value was then kept constant until a next oil absorption value was recorded.

#### 3.2.1.1 Conditional filtering

##### Step 1: Removal of downtime data

The first step in raw data filtering was the removal of those sections of the data corresponding with any shut-down times. The filtering was done in Microsoft Excel, using the built-in filtering option.

Conditions that were treated as downtime are listed in Table 3.2-1:

**Table 3.2-1: Down-time filtering conditions**

|                 |
|-----------------|
| Unit Down       |
| I/O Timeout     |
| Comm Fail       |
| Values $\leq 0$ |

### Step 2: Removal of logical outliers

The second step in raw filtering was to eliminate values that were logically incorrect and attributable to measurement errors or through visible spikes. The values were chosen using logic and knowledge of the plant process engineer. Table 3.2-2 shows the limits used for data removal for each of the variables.

**Table 3.2-2: Logical limits for data filtering**

| <b>Variable</b>            | <b>Values removed</b>             |
|----------------------------|-----------------------------------|
| Fluidising air flow rate   | $\leq 500 \text{ m}^3/\text{h}$   |
| Fluidising air temperature | $\leq 80 \text{ }^\circ\text{C}$  |
| Liquid AN density          | $\leq 1.04$                       |
| Liquid AN temperature      | $\leq 100 \text{ }^\circ\text{C}$ |
| Liquid AN flow rate        | $\leq 2.3 \text{ tph}$            |
| Atomising air flow rate    | $\leq 80 \text{ kPa}$             |

#### **3.2.1.2 Statistical filtering**

Statistical filtering involved the use of a special box-and-whisker diagram wherein natural operating limits for the data was established. The box-and-whisker diagram in this case has its limits based on standard deviation from the mean instead of the usual interquartile range due to the uncertainty of the normal distribution of the data. Large sets of data usually display normal distributions, however due to the inconsequent and random nature of this data, a normal distribution is not evident and thus necessitated a special box-and-whisker analysis.

The diagram consisted of five values including the mean, two times the standard deviation above and below the mean and the minimum and maximum values. The upper and lower limits for

filtering were chosen as the two standard deviations from the mean. The box-and-whisker diagram calculations for fluidising air flow rate is illustrated in Table 3.2-3 and Figure 3.2-1.

Table 3.2-3: Box-and-whisker diagram values

|                               |       |                    |
|-------------------------------|-------|--------------------|
| <b>Minimum</b>                | 505   |                    |
| <b>-2× Standard deviation</b> | 8845  | <b>Lower limit</b> |
| <b>Mean</b>                   | 11695 |                    |
| <b>+2× Standard deviation</b> | 14544 | <b>Upper limit</b> |
| <b>Maximum</b>                | 16328 |                    |

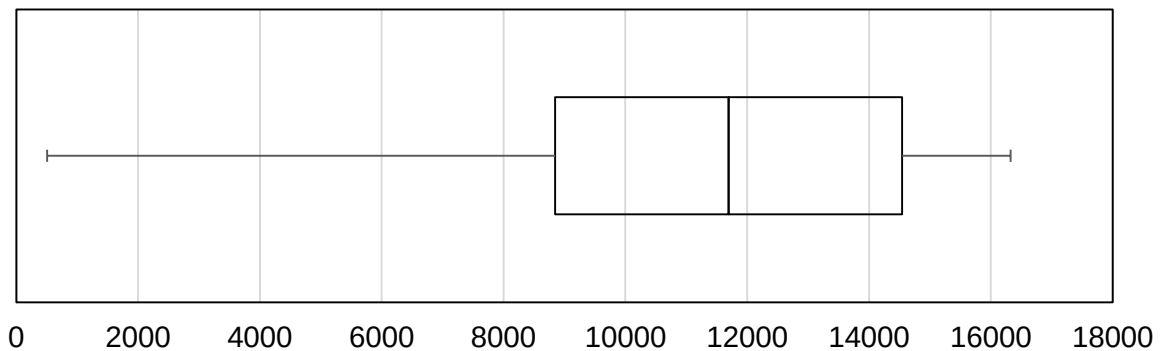


Figure 3.2-1: Box-and-whisker diagram illustration

The full data set for each input was then filtered for the upper and lower limits obtained with the box-and-whisker diagrams.

Another statistical filtering technique that was applied to the constant-time data sets that is described in the following section, is the sensor data smoothing and filtering technique provided in Perry and Green (1997:8-55). The equations used for sensor filtering is given in Equation 3.1-1 and 3.1-2.

$$Y_i = \alpha x_i + (1 - \alpha)Y_{i-1} \quad 3.2-1$$

$$\alpha = \frac{\Delta t}{\Delta t + \tau_f} \quad 3.2-2$$

In the equations above,  $y_i$  and  $y_{i-1}$  represent current and previous filter values,  $x_i$  is the unfiltered value,  $\Delta t$  is the sampling interval (in this case 1 minute) and  $\tau_f$  is the time constant. The time constant is obtained through trial and error until the noise of the data is found to be adequately filtered, but without losing the dynamics of the actual sensor signal.

### 3.2.1.3 Data sets produced and assessed in phase 1 model development

#### Filtered data

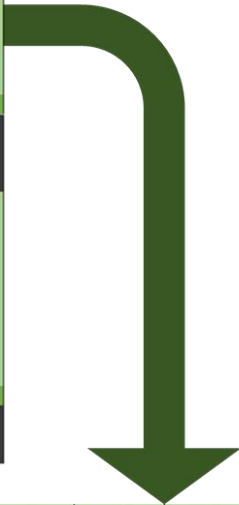
This data set is the product of the conditional and statistical filtering process. It does, however, exclude the sensor smoothing and filtering part mentioned in the previous section. This data set considers each set of input values at time stamps that were considered normal by the conditional and statistical filtering techniques. Due to irregular quality measurements conducted on the plant, some oil absorption values were kept constant for quite some time during periods wherein there is a lack of this quality variable data.

#### Discrete data

The discrete data set, which is a sub-set of the filtered data set described above, was acquired by taking both input and output data points only at a point when there was a unique oil absorption value, thus not keeping the input data points where oil absorption was not recorded. The idea is to record the input data points only at a time when the porosity value changes to assess how efficient would such a largely reduced data set be in generating a predictive model. Most of the input data points are rejected in generating this data set. It is also observed that the time-based dynamics of the process is not considered, as only a snap shot of the plant at largely spaced, irregular intervals are considered.

#### Constant-time data

The philosophy around the constant-time data sets was based on a plant running with continuous punctual quality control. The data was generated using the conditionally and statistically filtered data set in a way that the unique oil absorption values were used together with the input parameters' data of the same time stamp, and the input parameters' data of the following predetermined period. In a case where the predetermined time is e.g. 10 minutes, it created the illusion that oil absorption analysis was done every 10 minutes. The oil absorption value was then kept constant for this time. This concept is illustrated in Figure 3.2-2 below, where the dark green line represents each data point when a new oil absorption value was recorded, the light green part is the data points for which the abovementioned oil absorption value was kept constant (in this case 10 data points is 10 minutes), and the grey/black data points represent discarded data.



|     |       |     |      |     |     |     |       |
|-----|-------|-----|------|-----|-----|-----|-------|
| 107 | 11091 | 135 | 1.39 | 140 | 8.4 | 441 | 13.54 |
| 108 | 11035 | 135 | 1.39 | 140 | 8.3 | 439 | 13.54 |
| 109 | 11308 | 135 | 1.39 | 140 | 8.3 | 440 | 13.54 |
| 110 | 11231 | 135 | 1.39 | 140 | 8.5 | 439 | 14.15 |
| 111 | 11265 | 135 | 1.39 | 140 | 8.4 | 440 | 14.15 |
| 112 | 10973 | 135 | 1.39 | 140 | 8.2 | 434 | 14.15 |
| 113 | 10955 | 135 | 1.39 | 140 | 8.1 | 434 | 14.15 |
| 114 | 11092 | 135 | 1.39 | 140 | 8.3 | 438 | 14.15 |
| 115 | 10970 | 135 | 1.39 | 140 | 8.3 | 436 | 14.15 |
| 116 | 11152 | 135 | 1.39 | 140 | 8.2 | 435 | 14.15 |
| 117 | 11418 | 135 | 1.39 | 140 | 8.3 | 441 | 14.15 |
| 118 | 11224 | 135 | 1.39 | 140 | 8.1 | 436 | 14.15 |
| 119 | 11013 | 135 | 1.39 | 140 | 8.3 | 438 | 14.15 |
| 120 | 11348 | 135 | 1.39 | 140 | 8.2 | 435 | 14.15 |
| 268 | 10600 | 135 | 1.39 | 140 | 8.4 | 440 | 14.15 |
| 269 | 10733 | 135 | 1.39 | 140 | 8.4 | 442 | 14.15 |
| 270 | 10686 | 135 | 1.39 | 140 | 8.5 | 443 | 14.15 |
| 271 | 10714 | 135 | 1.39 | 140 | 8.3 | 442 | 14.15 |
| 272 | 11061 | 135 | 1.39 | 140 | 8.4 | 443 | 13.50 |
| 273 | 10505 | 135 | 1.39 | 140 | 8.2 | 438 | 13.50 |
| 274 | 10430 | 135 | 1.39 | 140 | 8.2 | 441 | 13.50 |
| 275 | 11383 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 276 | 11256 | 135 | 1.39 | 140 | 8.2 | 441 | 13.50 |
| 277 | 10896 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 278 | 10879 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 279 | 11330 | 135 | 1.39 | 140 | 8.5 | 446 | 13.50 |
| 280 | 11133 | 135 | 1.39 | 140 | 8.2 | 440 | 13.50 |
| 281 | 11211 | 135 | 1.39 | 140 | 8.1 | 437 | 13.50 |
| 282 | 11340 | 135 | 1.39 | 140 | 8.0 | 437 | 13.50 |
| 283 | 11215 | 135 | 1.39 | 140 | 8.0 | 438 | 13.50 |
| 284 | 11330 | 135 | 1.39 | 140 | 8.3 | 443 | 13.50 |
| 285 | 10872 | 135 | 1.39 | 140 | 8.3 | 443 | 13.50 |

|    |       |     |      |     |     |     |       |
|----|-------|-----|------|-----|-----|-----|-------|
| 1  | 11231 | 135 | 1.39 | 140 | 8.5 | 439 | 14.15 |
| 2  | 11265 | 135 | 1.39 | 140 | 8.4 | 440 | 14.15 |
| 3  | 10973 | 135 | 1.39 | 140 | 8.2 | 434 | 14.15 |
| 4  | 10955 | 135 | 1.39 | 140 | 8.1 | 434 | 14.15 |
| 5  | 11092 | 135 | 1.39 | 140 | 8.3 | 438 | 14.15 |
| 6  | 10970 | 135 | 1.39 | 140 | 8.3 | 436 | 14.15 |
| 7  | 11152 | 135 | 1.39 | 140 | 8.2 | 435 | 14.15 |
| 8  | 11418 | 135 | 1.39 | 140 | 8.3 | 441 | 14.15 |
| 9  | 11224 | 135 | 1.39 | 140 | 8.1 | 436 | 14.15 |
| 10 | 11013 | 135 | 1.39 | 140 | 8.3 | 438 | 14.15 |
| 11 | 11348 | 135 | 1.39 | 140 | 8.2 | 435 | 14.15 |
| 12 | 11061 | 135 | 1.39 | 140 | 8.4 | 443 | 13.50 |
| 13 | 10505 | 135 | 1.39 | 140 | 8.2 | 438 | 13.50 |
| 14 | 10430 | 135 | 1.39 | 140 | 8.2 | 441 | 13.50 |
| 15 | 11383 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 16 | 11256 | 135 | 1.39 | 140 | 8.2 | 441 | 13.50 |
| 17 | 10896 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 18 | 10879 | 135 | 1.39 | 140 | 8.3 | 441 | 13.50 |
| 19 | 11330 | 135 | 1.39 | 140 | 8.5 | 446 | 13.50 |
| 20 | 11133 | 135 | 1.39 | 140 | 8.2 | 440 | 13.50 |
| 21 | 11211 | 135 | 1.39 | 140 | 8.1 | 437 | 13.50 |
| 22 | 11340 | 135 | 1.39 | 140 | 8.0 | 437 | 13.50 |

Figure 3.2-2: Compilation of constant-time data

This filtering was also done for other time intervals including 5, 15, 20, 30 and 40 minutes to assess which predetermined period would yield the best predictive model. Sensor smoothing, as described in the previous section, was applied to the input sensor signals of these constant-time data sets to assess the influence thereof on the predictive model.

### 3.2.2 Phase 2 data collection and preparation

Phase 2 consisted of taking samples and collecting sensor data on the plant. The samples were analysed using the optical porosity measurement method (described in section 3.1.1.2) to produce granular porosity as well as sphericity using the method described in section 3.1.2. These selectivity variable values were used together with the plant sensor data to train the artificial neural networks.

### 3.2.2.1 Data processing

The input data was obtained from the data historian for the period 12:30 on 9 May 2018 to 07:00 on 11 May 2018. During this time, the granulator parameters were changed to produce high density ammonium nitrate (HDAN) instead of regular porous granular ammonium nitrate (PGAN). The input parameters used in phase 2 differ slightly from the input parameters used in phase 1 due to some of the sensor data not being available as before, however, each parameter that was changed to something else, is very well related to its replacement. Table 3.2-4 shows the difference in input parameters between phase 1 and phase 2.

**Table 3.2-4: Input parameter changes from phase 1 to phase 2**

| <b>Phase 1</b>             | <b>Phase 2</b>            |
|----------------------------|---------------------------|
| Fluidising air flow rate   | Fluidising air pressure   |
| Fluidising air temperature | Fluidised bed temperature |
| Liquid AN density          | Liquid AN concentration   |
| Liquid AN temperature      | Liquid AN temperature     |
| Liquid AN flow rate        | Liquid AN pressure        |
| Atomising air flow rate    | Atomising air flow rate   |

The output data was obtained by taking samples in 10-minute intervals. The samples were analysed using the optical porosity measurement method, after which approximately 10% of the samples were subjected to the oil absorption measurement technique (described in Section 3.1.1.1) for comparison. All the data was then compiled into one spreadsheet for processing.

The plant had some downtime during the trial period, necessitating the removal of the downtime values. A time-advance needed to be calculated to see how long it takes for the granules to travel from the point of granulation to the sampling point. After the final dataset was obtained, three of the six input parameters were filtered with the sensor smoothing technique proposed by Perry and Green (1997:8-55) to remove typical sensor noise.

### 3.2.2.2 Process sampling

Figure 3.2-3 shows a process flow diagram of the granulator setup. Seed ammonium nitrate particles are fed into the granulator (point C in the figure) using a screw conveyor (A). The seed particles consist of new seed material as well as recycled material, which consisted of both undersize product (J) and crushed oversize (K) material. Air is fed into the granulator using a fan

(N) after it is heated using a heat exchanger (B). The air goes through a perforated plate before fluidising the seed particles. Granulation occurs when the dry seed particles comes into contact with a liquid binding agent, which, in this case, is liquid ammonium nitrate. The ammonium nitrate spray enters the granulator through nozzles (E) on the side of the granulator. This means the granulator is operated as a side-spray granulator. The liquid ammonium nitrate is combined with atomising air in the nozzles to form an atomised spray. The granulator is divided into two sections, namely the spray section and the drying section (D). There are no nozzles in the drying section and particles are only fluidised and cooled down in this section.

The product particles exit the granulator at point F with a conveyor belt to be transported to the screens (G & H). The first screen (G) removes oversize product, which is then transported to the jaw crusher (K) to be crushed and recycled to the granulator. The second screen (H) removes undersize particles from the product, which is also recycled to the granulator. The on-size product particles are transported for further processing using a conveyor belt (L), where the samples of the final product are taken for analysis.

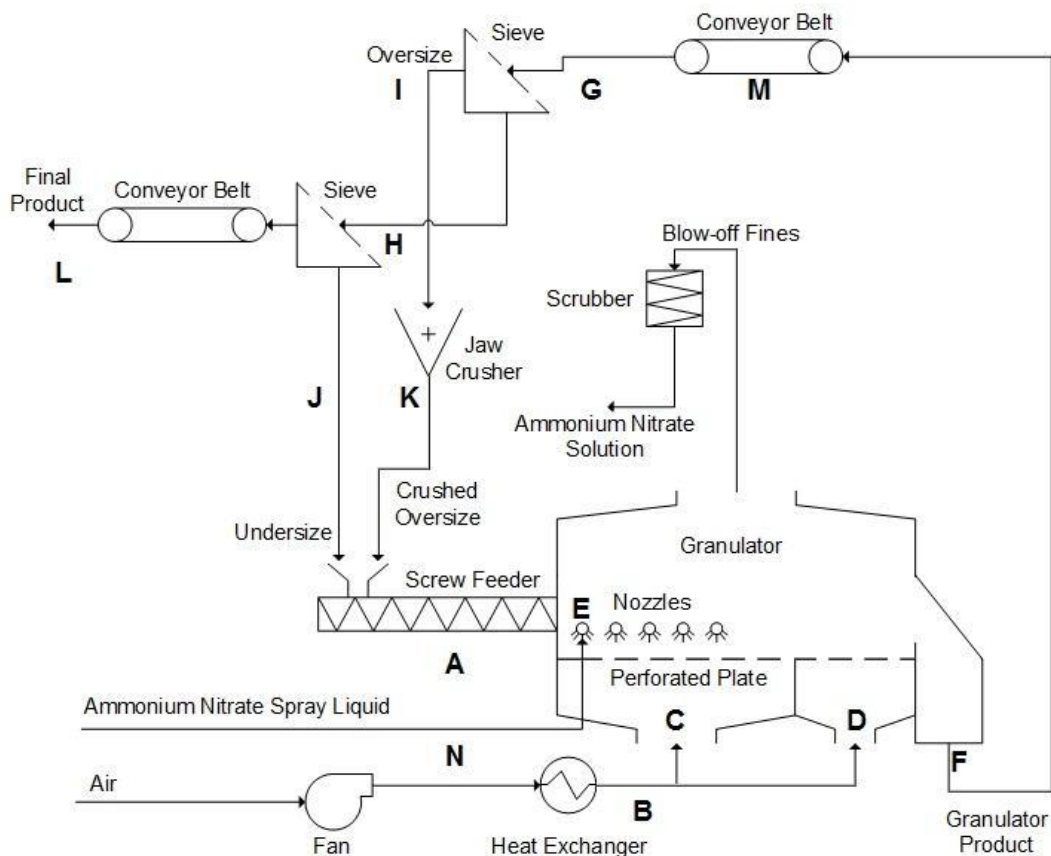


Figure 3.2-3: Ammonium nitrate granulator process flow diagram

### 3.2.2.3 Sample analysis

Porosity and sphericity values were determined for each sample. As will be seen in Section 5.1.1.3, the optical porosity measurement method correlates well with the oil absorption method and was thus used for porosity determination. The reason for using the optical porosity measurement method is that it is faster to execute than the oil absorption. A sub-set of 20 samples were additionally subjected to oil absorption to verify the optical porosity measurement values as well as to obtain a regression line equation for changing optical porosity measurement values into oil absorption values.

#### Optical porosity measurement

The optical porosity measurement analysis was done as explained in Section 3.1.1.2. The samples were sieved into a smaller size fraction so that the  $d_{50}$  value is still representative of the whole sample. For these analyses, the samples were screened to a size range of 2.5 mm to 3.0 mm. The samples were then analysed using the Haver CPA 2-1 particle size analyser. The calculation method remains as described in Section 3.1.1.2.

#### Oil absorption

Oil absorption was done on 20 samples to verify the optical porosity measurement method. Approximately 2.5 g of the sample was used to determine oil absorption. The method as well as calculations are explained in Section 3.1.1.1.

### 3.2.2.4 Time-advance determination

The samples were taken relatively far from the granulator and a time-delay on the inputs compared to the outputs were expected. The time-advances were calculated on a part of the data that is considered continuous (without any shut-downs) to get the best representative value for a continuous running plant.

The time-advance was obtained by creating a correlation graph where the output value is shifted in time for each input value. The first correlation value was calculated with the input and output data perfectly synchronised, while the second correlation was calculated by shifting the output data back in time with one minute, relative to the inputs. The data was shifted 30 times in total, with one-minute shift intervals. The correlation (R) was calculated using Equation 3.2-3.

$$R = \frac{n \sum xy - \sum x \sum y}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \quad 3.2-3$$

In the equation above,  $x$  represents the input value at a given time,  $y$  represents the output value corresponding to the input value and  $n$  is the total number of values present.

The correlation values were plotted against time (shifted) values and a minimum (negative correlation) or maximum (positive correlation) was observed at the advance time the input parameter has on the output. The time value at the minimum or maximum shows the time-advance that the output (quality value) needs to be shifted back in time to correspond with the set of input values in real time. The reason for this analysis is to eventually guide the predictive model to estimate the values of the selectivity variables at the point of granulation for in-time control, and not at the time of a down-stream point in the plant.

### 3.2.2.5 Linear statistical sensor filtering

Three of the six input values had sensor noise that needed to be filtered. The input values that were filtered include fluidising air pressure, liquid AN pressure and atomising air pressure. The procedure for filtering was explained in Section 3.2.1.2.

## 3.3 Chapter summary

The porosity measurement technique was investigated for both precision and to possibly suggest a better techno-economic technique if available. The technique used by the industry partner is based on liquid absorption called oil absorption. A new method was also developed and named the optical porosity measurement method. The method is based on bulk density as well as a density based on the sphericity of the particles, particle count as well as the mass of the sample. The sphericity test was tested for precision by repeating the measurement technique and calculating a repeatability on the samples.

The procedure for the data collections for the predictive model analysis was explained in detail. This includes data filtering, the compilation of different datasets (phase 1) as well as process sampling and the analyses of the samples (phase 2).

## Chapter 4: Model development

The chapter below explains the model development of phases 1 and 2.

Section 4.1 consists of the development of an artificial neural network for phase 1 data. The equipment used as well as the procedure is explained.

Section 4.2 elaborates on the development procedure for a model for phase 2 data, which separately consists of the development of artificial neural networks for porosity prediction and sphericity prediction. Both dataset's procedures are explained separately.

### 4.1 Predictive model based on phase 1 data

The artificial neural network testing in phase 1 consisted of assessing each of the generated data sets (as explained and detailed in Section 3.2.1.3) on different neural network configurations and topologies.

#### 4.1.1 Model equipment

The MatLab™ software's artificial neural network platform was used to obtain and evaluate the different artificial neural networks. The built-in graphic interface was used.

#### 4.1.2 Model procedure

The artificial neural networks all consisted of an input layer, one hidden layer and an output layer. The activation function was also kept constant as the hyperbolic tangent function. As stated in Section 2.2.2, the hyperbolic tangent function is chosen for its ability to predict output values with complicated relations to the input values. The hyperbolic tangent function was, however, tested against other activation functions in phase 2 for diligence.

Figure 4.1-1 shows the different training algorithms and hidden layer neuron configurations assessed in phase 1.

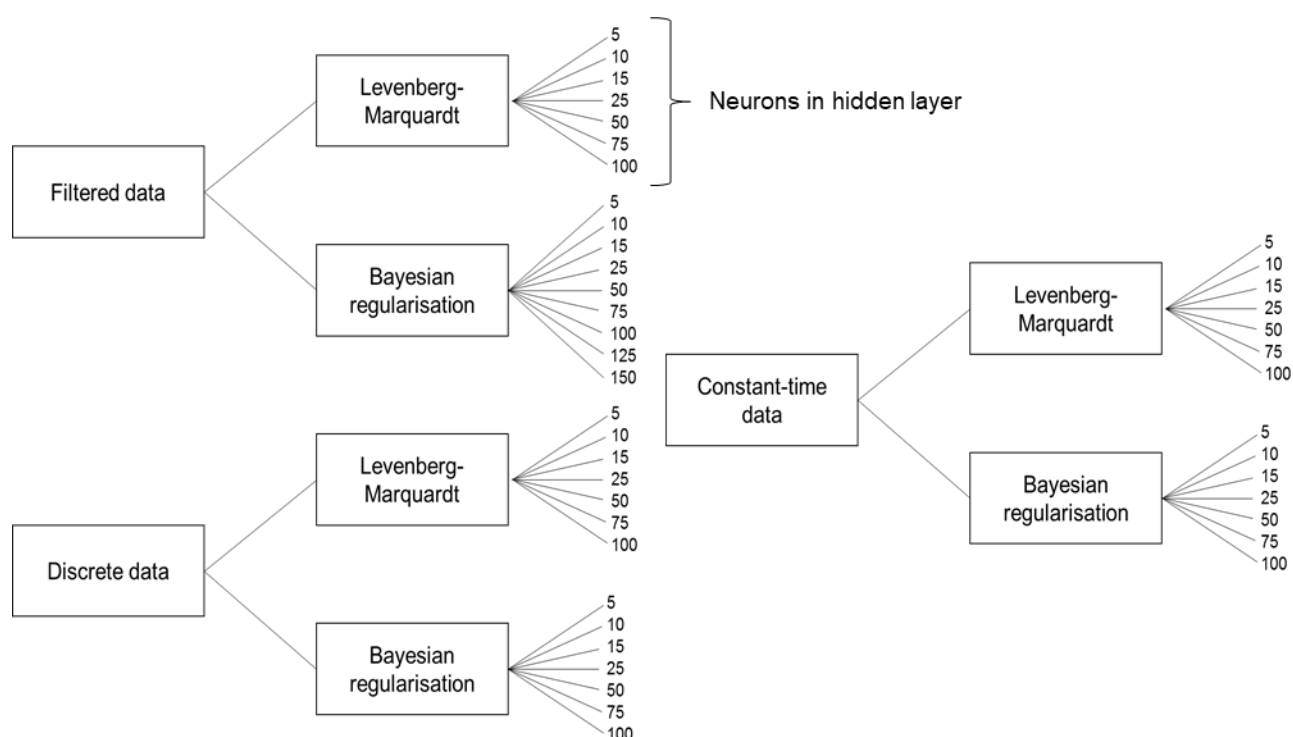


**Figure 4.1-1: Neural network configurations and topologies tested - Phase 1**

The training variables that were chosen before each run, were kept constant to ensure comparability. Table 4.1-1 lists the variables chosen for the Levenberg-Marquardt training algorithm and Table 4.1-2 the variables chosen for the Bayesian regularisation algorithm. The initial weights for the Levenberg-Marquardt training were set to 0.5 to keep it constant; however, when using the Bayesian regularisation method, it resulted in unusable (constant) output values. The weights for the Bayesian regularisation method were therefore randomised.

**Table 4.1-1: Training variables for Levenberg-Marquardt training algorithm**

| Training parameter           | Set value |
|------------------------------|-----------|
| Epoch                        | 1000-4000 |
| Performance goal             | 0         |
| Maximum validation failures  | 6         |
| Minimum performance gradient | $10^{-7}$ |
| Initial mu                   | 0.001     |
| Mu decrease factor           | 0.1       |
| Mu increase factor           | 10        |
| Maximum mu                   | $10^{10}$ |
| Initial weights              | 0.5       |

**Table 4.1-2: Training variables for Bayesian regularisation training algorithm**

| Training parameter           | Set value |
|------------------------------|-----------|
| Epoch                        | 1000-4000 |
| Performance goal             | 0         |
| Minimum performance gradient | $10^{-7}$ |
| Initial mu                   | 0.005     |
| Mu decrease factor           | 0.1       |
| Mu increase factor           | 10        |
| Maximum mu                   | $10^{10}$ |

Each data set was divided into a training set and a simulation set. The training set consisted of 80% of the original dataset. The training dataset was then randomly divided into training, validation and test data, for the Levenberg-Marquardt training algorithm, or training and test data, for the Bayesian regularisation training algorithm. The percentage of data used for each subset is given in Figure 4.1-2. The simulation set consisted of 20% of the data, which was kept completely separate from the training data, and was chosen so that the minimum and maximum values of the complete data set are still in the training set. This is important to always ensure that the ANN performs interpolation during estimation of the output, as extrapolation has proven to be inaccurate, as explained in Section 2.2 (Abraham, 2004; Burggraeve, 2013; Burns, 2001:358).

The performance evaluation of each artificial neural network was based on the mean-squared error (MSE) as calculated as well as the correlation coefficient between the artificial neural network outputs and target values. A mean-squared error closer to zero indicates a good performance while the correlation coefficient should be as close as possible to one.

For each dataset, the two best performing neuron configurations were obtained. To establish whether a better performing neuron configuration can be obtained, the performance evaluation values, namely the MSE and correlation coefficient of each tested neuron configuration were added to a graph. As far as possible quadratic trend lines were fitted to the graphs, so that the number of neurons at the minimum and maximum for each of the performance evaluations could be estimated. The number of neurons were obtained using the built-in MS-Excel™ solver tool. An average between the two values was subsequently assessed to establish whether the trend line prediction was correct.

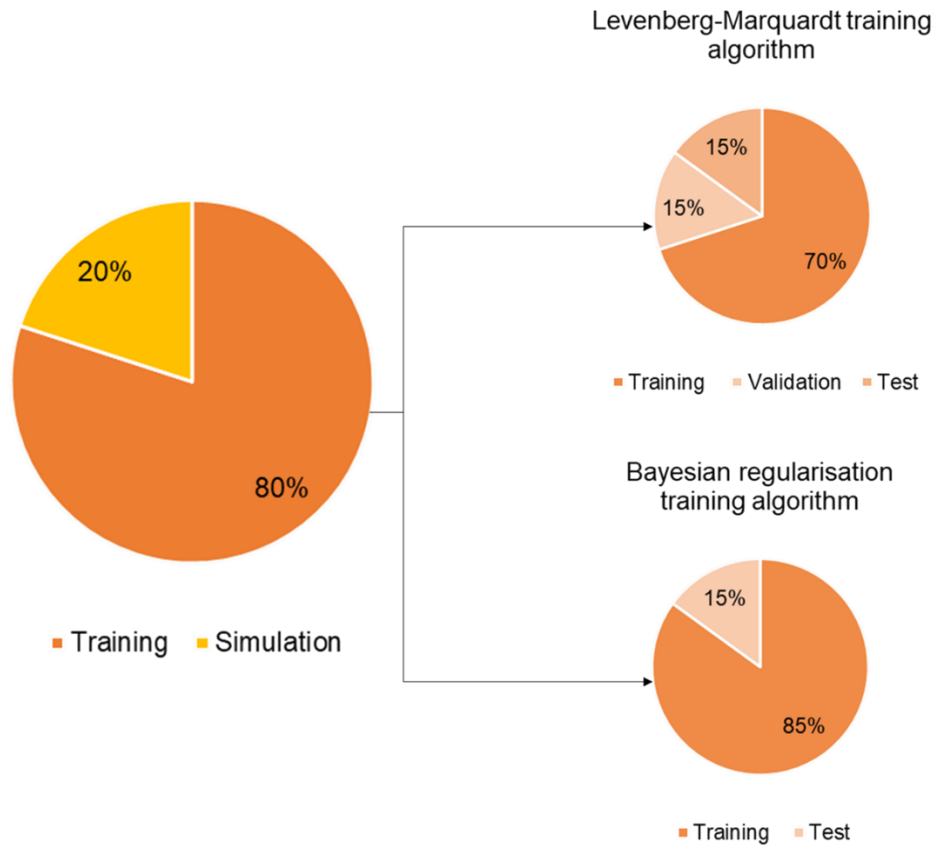


Figure 4.1-2: Division of data for neural network testing

## 4.2 Predictive model based on phase 2 data

For phase 2, artificial neural network configurations were tested to predict both porosity and sphericity. Porosity artificial neural network assessment originated from knowledge obtained in phase 1, while the sphericity estimation artificial neural network was based only on lessons learned during both phases.

### 4.2.1 Model equipment

As with phase 1, MatLab™ R2016b along with the neural network toolbox was used to train and simulate the different configurations.

### 4.2.2 Model procedure

Each artificial neural network consisted of an input layer with 6 inputs, one hidden layer and an output layer, with 1 output. Different procedures were followed for each of the different outputs, which are explained below.

#### 4.2.2.1 Porosity

The artificial neural networks assessed for the porosity in phase 2 were based on observations during ANN generation in phase 1. The hyperbolic tangent activation function was used for all configurations, however, the sigmoid function as well as the linear activation function were also tested on the best performing neural network configuration.

From phase 1, the Bayesian regularisation training algorithm showed the best performance between the two training algorithms. Phase 2 artificial neural network testing consisted of using sensor filtered data, with the Bayesian regularisation training algorithm and a fairly large number of neurons in the hidden layer. Figure 4.2-1 shows the artificial neural network assessment programme for phase 2 porosity data.

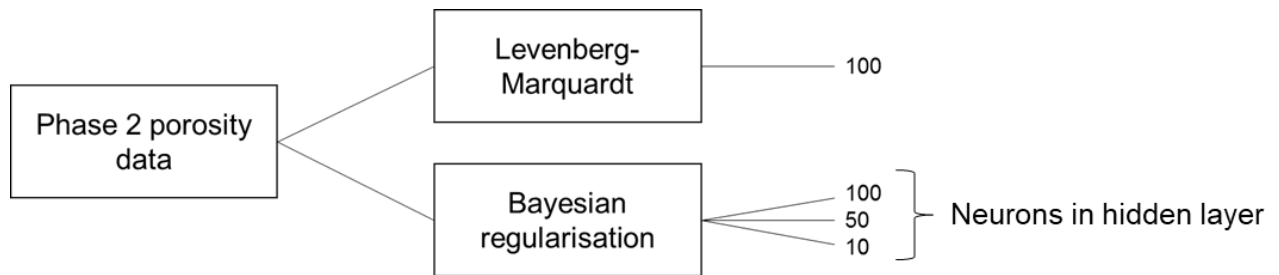


**Figure 4.2-1: Neural network configurations tested – Phase 2 Porosity data**

The Levenberg-Marquardt training algorithm was tested on one set of neuron topologies to be sure that it does not outperform the Bayesian regularisation training algorithm. The 10-neuron neural network was conducted to obtain a representative quadratic trend line, which is used to determining the neuron configuration which should theoretically perform the best.

The procedure for the rest of the artificial neural network assessment was kept similar to those in phase 1, including the division of the dataset into training and simulation, the training variables for each training algorithm as well as the method used for performance evaluation. The procedure is fully explained in Section 4.1.2.

#### 4.2.2.2 Sphericity

The test procedure for the ANN used to estimate ammonium nitrate granular sphericity is similar to the procedure followed in phase 1 as sphericity data was not available in phase 1 to test all possible network configurations. The hyperbolic tangent activation function was used for all the

tests, while the sigmoid and linear activation functions were also tested on the best performing neural network. Figure 4.2-2 shows the artificial neural network configurations that were assessed for the estimation of sphericity data.

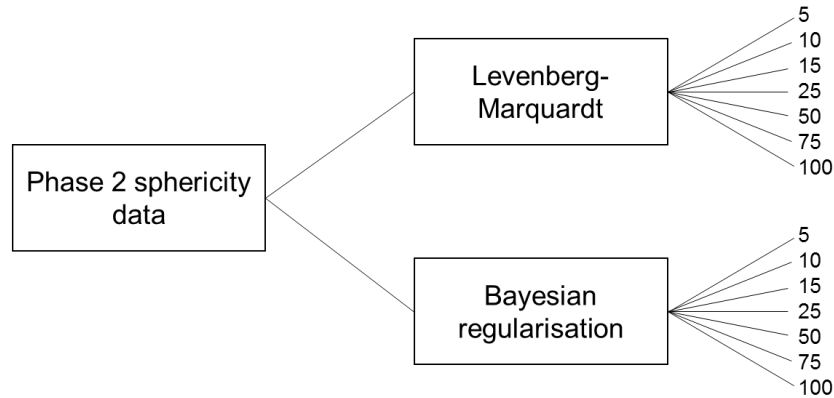


**Figure 4.2-2: Neural network configurations tested – Phase 2 Sphericity data**

### 4.3 Chapter summary

The model development for this study was divided into two phases. Phase 1 used historical plant data, even though the data needed major processing before being able to be used. The first phase tested different types of datasets on different topologies and two different training algorithms, for the prediction of porosity. Phase 2 of the model development used self-acquired data. The data consisted of both porosity prediction data as well as sphericity prediction data. The neural network testing for porosity prediction was based on results obtained in phase 1 testing, while the neural network testing for sphericity prediction was done similar to phase 1 testing.

## Chapter 5: Results and discussion

In this chapter, the results of each experimental procedure discussed in Chapter 3 are given and discussed.

Section 5.1 consists of the results and discussion of the selectivity variables' measurement techniques together with the validation thereof. The porosity measurement techniques are discussed and compared first, after which the sphericity measurement technique is tested for repeatability.

Section 5.2 contains the results of the predictive models acquired using phase 1 and phase 2 data. The results of the data processing in each phase is provided first. After this, the results of all tested neural network configurations (as detailed in Chapter 4) for both phases are shown and discussed.

For phase 1 data (Section 5.2.1), the neuron configurations for each data set are compared to find the best configuration after which the best training algorithm for each data set is determined. The best neural network for each data set is compared to one another to obtain the best data set, which then demarcated the data scope for phase 2 data.

For phase 2 data (Section 5.2.2), the porosity prediction results are discussed first, as the neural networks tested in this section is based on findings from phase 1 data. The sphericity prediction results are summarised in a similar way as phase 1 results, where neuron configurations are compared, followed by training algorithms comparison, in order to obtain the best possible artificial neural network for the prediction of sphericity. In each of these data sets, the best neural network was tested with different activation functions and the results are also shown and discussed in this section.

### 5.1 Selectivity variables measurement and validation

#### 5.1.1 Porosity measurement

Porosity measurements were conducted on 8 samples with unknown porosities. The results of each method used to measure the porosity is given in the paragraphs below, after which a comparison is made between the oil absorption method and each of the other methods.

### 5.1.1.1 Oil absorption

The oil absorption method was done on all 8 samples. Samples 1, 5 and 8 were repeated for the sake of repeatability calculations. Samples 1 and 8 showed significant variety in the tests and were thus subjected to five repetitions each. An outlier test was used for these two samples to determine if any of the results should be discarded. The outliers test and results are shown in Appendix A.1.1. Each of the two samples had one outlier. The repeatability calculations are detailed in Appendix A.1.2.

As explained in Section 3.1.1.1, two types of porosity percentage values were calculated, namely one based on the mass of absorbed liquid and another based on the volume of liquid absorbed. Although porosity of particles is normally a volume-based attribute, the industry partner uses a porosity percentage based on mass for quality control purposes. Therefore, both values were calculated and used for comparison.

Table 5.1-1 as well as Figure 5.1-1 show the average results obtained for both the mass-based porosity percentage and the volume-based porosity percentage.

**Table 5.1-1: Oil absorption granule porosity results**

| <b>Sample</b> | <b>Volume based porosity [%]</b> | <b>Mass based porosity [%]</b> |
|---------------|----------------------------------|--------------------------------|
| <b>1</b>      | 17.2 %                           | 8.7 %                          |
| <b>2</b>      | 22.6 %                           | 11.8 %                         |
| <b>3</b>      | 29.6 %                           | 16.1 %                         |
| <b>4</b>      | 30.4 %                           | 16.7 %                         |
| <b>5</b>      | 26.6 %                           | 14.3 %                         |
| <b>6</b>      | 31.0 %                           | 17.0 %                         |
| <b>7</b>      | 32.4 %                           | 18.0 %                         |
| <b>8</b>      | 21.6 %                           | 11.2 %                         |

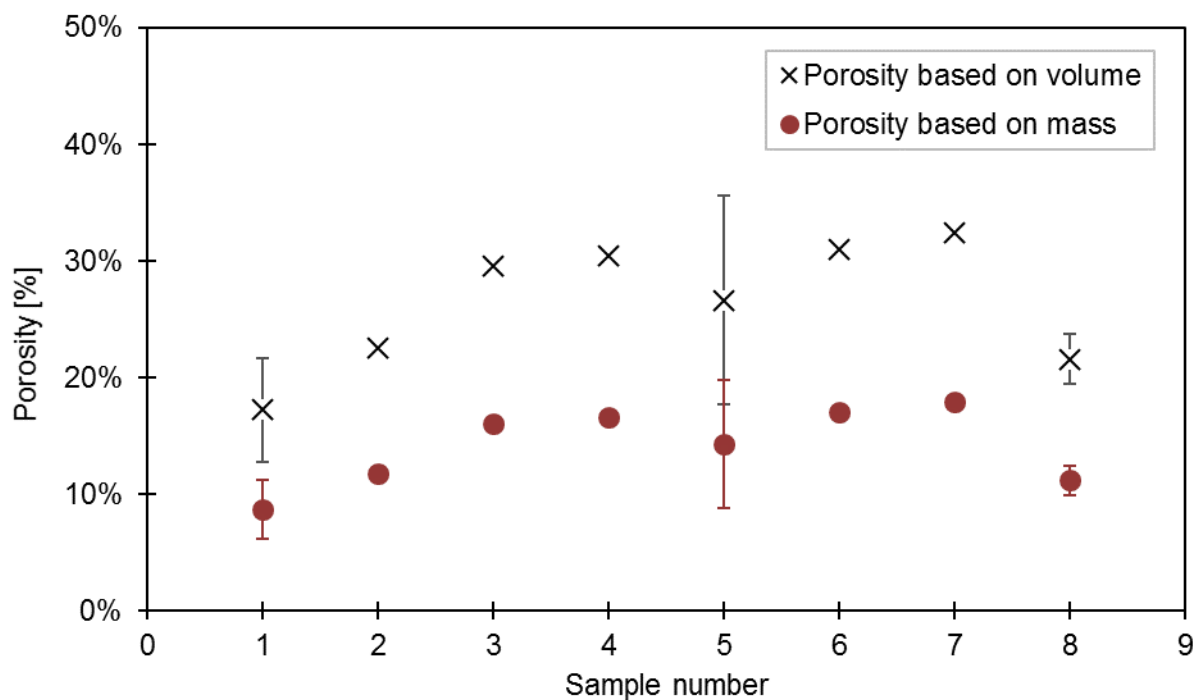


**Figure 5.1-1: Oil absorption results**

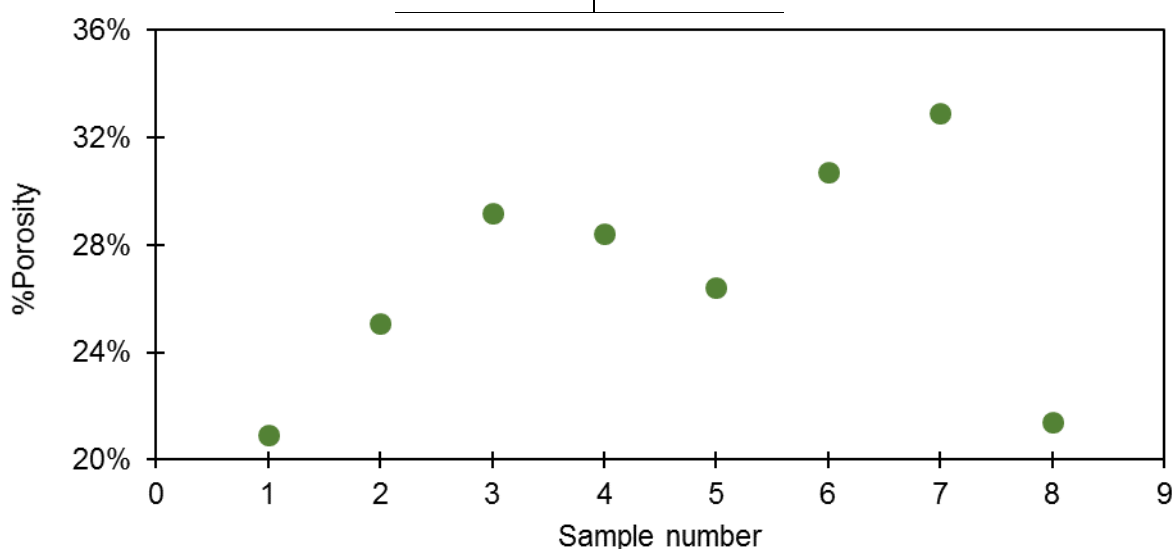
In the graph above, volume-based porosity percentage is perceived to be higher than mass-based porosity, however, a similar trend is observed. The most porous sample is sample 7, with a volume-based porosity of 32.4 % which corresponds to 18.0 % by mass of liquid absorbed. The least porous sample is sample 1, with a porosity of 17.2 % and a corresponding 8.7 % by mass of liquid absorbed. Sample 5 has the highest error value; however, the higher error value could be attributed to a smaller number of repeated experiments. The value used for comparison in the following experiments is the mass-based porosity, since this is the applied method used for quality control.

#### **5.1.1.2 Optical porosity measurement (OPM) method**

The optical porosity measurement method, as detailed in Section 3.1.1.2, was done on all 8 samples of granules. The full results are shown in Appendix B.2. Table 5.1-2 and Figure 5.1-2 shows the results for granular porosity analysis using the optical porosity measurement method.

**Table 5.1-2: Optical porosity measurement results**

| Sample | Porosity [%] |
|--------|--------------|
| 1      | 20.9 %       |
| 2      | 25.1 %       |
| 3      | 29.2 %       |
| 4      | 28.4 %       |
| 5      | 26.4 %       |
| 6      | 30.7 %       |
| 7      | 32.9 %       |
| 8      | 21.4 %       |

**Figure 5.1-2: Optical porosity measurement results**

Sample 1 has the lowest porosity of 20.9 %, resulting in the densest sample according the optical porosity measurement method. The most porous sample is sample 7 with a porosity percentage of 32.9 %.

### 5.1.1.3 Comparison of porosity techniques

The oil absorption method was compared to the optical porosity measurement method. Figure 5.1-3 shows the results of the two methods on the same graph. The results for both mass-based and volume-based porosity values originating from the oil absorption method is provided on the graph.

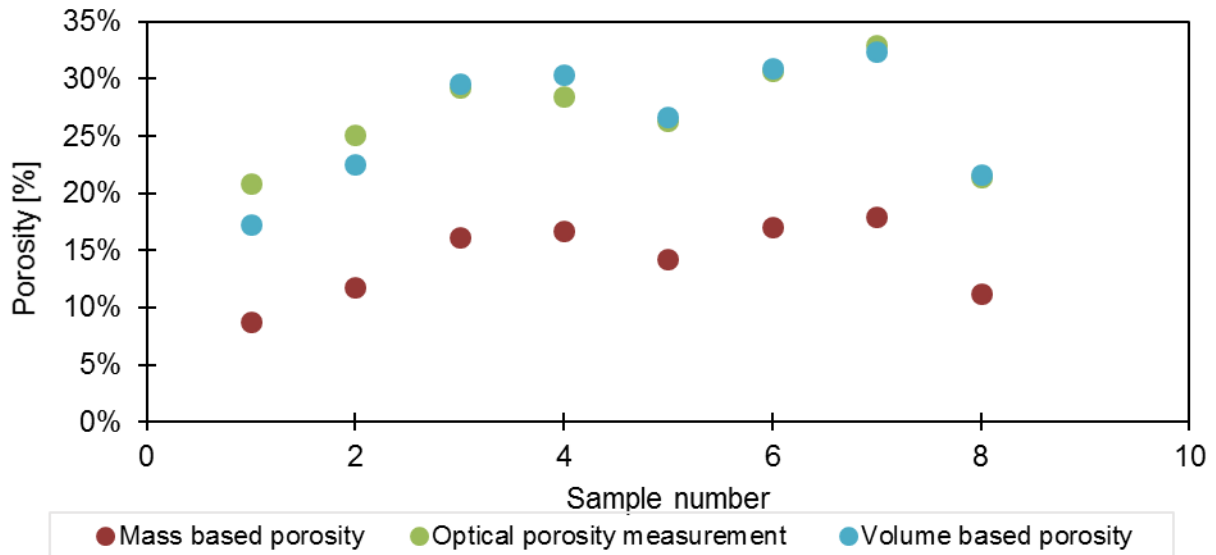


**Figure 5.1-3: Comparison of different porosity measurement techniques**

A linear regression plot was obtained to evaluate the agreement between the percentage porosity by mass using oil absorption and percentage porosity calculated using the optical porosity measurement method. Figure 5.1-4 shows the linear regression plot.

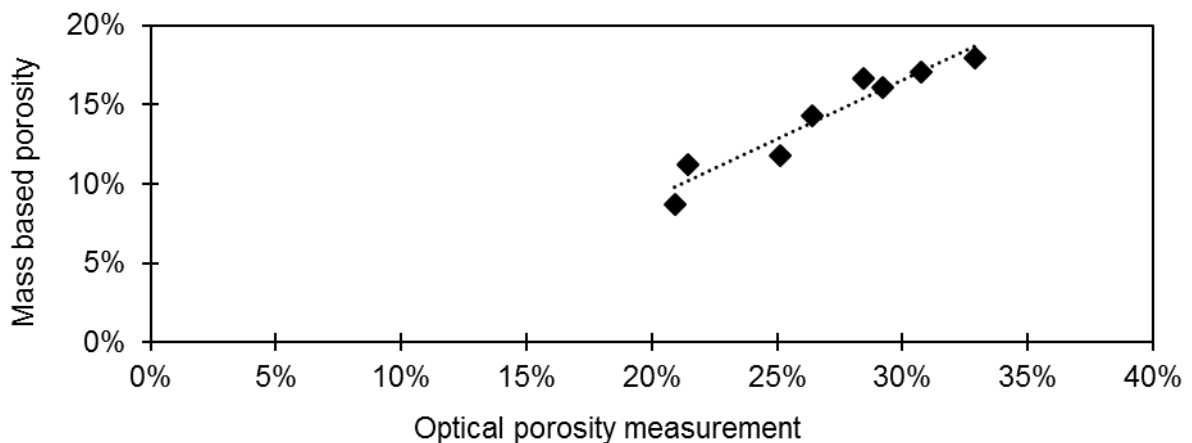


**Figure 5.1-4: Linear regression between mass-based porosity and OPM results**

The coefficient of determination between the two methods is 0.92, which indicates a very good correlation between the two methods. The mean absolute error between the two methods is 12.7 %. The regression results in a linear fit with an offset. This function is given by (Equation 5.1-1) and can be used to calculate a porosity percentage based on mass oil absorbed from optical porosity measurement values.

$$\text{Porosity \% (mass)} = 0.75(\text{OPM value}) - 0.06 \quad 5.1-1$$

### 5.1.2 Sphericity repeatability

Since the orientation of particles, when passing the lens in the particle size analyser, will be different with each pass, several passes will shed light on whether a two-dimensional particle circularity measurement can be used to estimate a three-dimensional sphericity value. Samples 1, 2, 6, 7 and 8 were subjected to analyses on the particle size analyser in order to assess repeatability thereof for this purpose. The results are tabulated in Table 5.1-3.

**Table 5.1-3: Sphericity repeatability results**

| Sample number | Mean ( $\bar{x}$ ) | Error              | %Error |
|---------------|--------------------|--------------------|--------|
| 1             | 0.82               | $4 \times 10^{-3}$ | 0.5 %  |
| 2             | 0.89               | $1 \times 10^{-3}$ | 0.2 %  |
| 6             | 0.87               | $1 \times 10^{-3}$ | 0.2 %  |
| 7             | 0.85               | 0                  | 0.0 %  |
| 8             | 0.91               | 0                  | 0.0 %  |
|               |                    |                    | 0.7 %  |

Sample 1 showed the highest deviation while sample 7 and 8 had no deviation. The highest deviation (sample 1) was calculated as 0.0043, which is considered a low deviation from a mean value of 0.820. Thus, the particle size analyser can be considered a precise analyser for determining mean circularity and thus particle sphericity.

## 5.2 Artificial neural network

### 5.2.1 Phase 1 predictive model

#### 5.2.1.1 Data processing

The original data after adding the mass-based porosity obtained using the oil absorption technique thereto, contained about 192 700 time-based data points for each input. Graphs of each input parameter together with the oil absorption porosity data, are given in Appendix C.2.

#### Conditional filtering

Raw filtering was applied on the data in preparation for statistical filtering. The procedure is explained in detail in Section 3.2.1.1, but consisted mainly of the removal of downtime data and

logical outliers. The filtering was applied for each individual input parameter and the graphs of the data sets are given in Appendix F.1.

### Statistical filtering

Statistical filtering analysis was done on each of the six input parameters. Each parameter's mean and standard deviation was calculated separately after which a box-and-whisker diagram was drawn up for each parameter. The five-number summaries along with their respective box-and-whisker diagrams are provided in Appendix C.1. Table 5.2-1 lists the minimum and maximum filtering values of each parameter.

**Table 5.2-1: Statistical filtering values**

| <b>Parameter</b>           | <b>Minimum</b> | <b>Maximum</b> |
|----------------------------|----------------|----------------|
| Fluidising air flow rate   | 8845           | 14544          |
| Fluidising air temperature | 97.8           | 143.0          |
| Liquid AN density          | 1.35           | 1.43           |
| Liquid AN temperature      | 118.5          | 150.6          |
| Liquid AN flow rate        | 2.35           | 13.78          |
| Atomising air flow rate    | 322            | 529            |
| Oil absorption             | 10.1           | 15.3           |

### Data sets produced

Three data sets were produced from the plant data obtained in phase 1, as detailed in Section 3.2.1.3. These include (i) a filtered data set, (ii) a discrete data set and (iii) a constant-time data set.

#### *Filtered data*

After filtering, the dataset contained 42 399 data points. Part of the graphs is shown in Figure 5.2-1. The full graphs are in Appendix C.2.2. Although these data sets do not contain any long stretches of down-time anymore, the time between consecutive oil absorption data is still not constant. The figure below shows each input value as the coloured line on each graph, while the mass-based granular porosity as output value in each graph is shown as a black line. The graphs show that the input data does not show large variations. The fluidising air flow rate and atomising

air flow rate shows some variation with changes in oil absorption values, however, both these input values contain noise that might affect the neural network's performance.

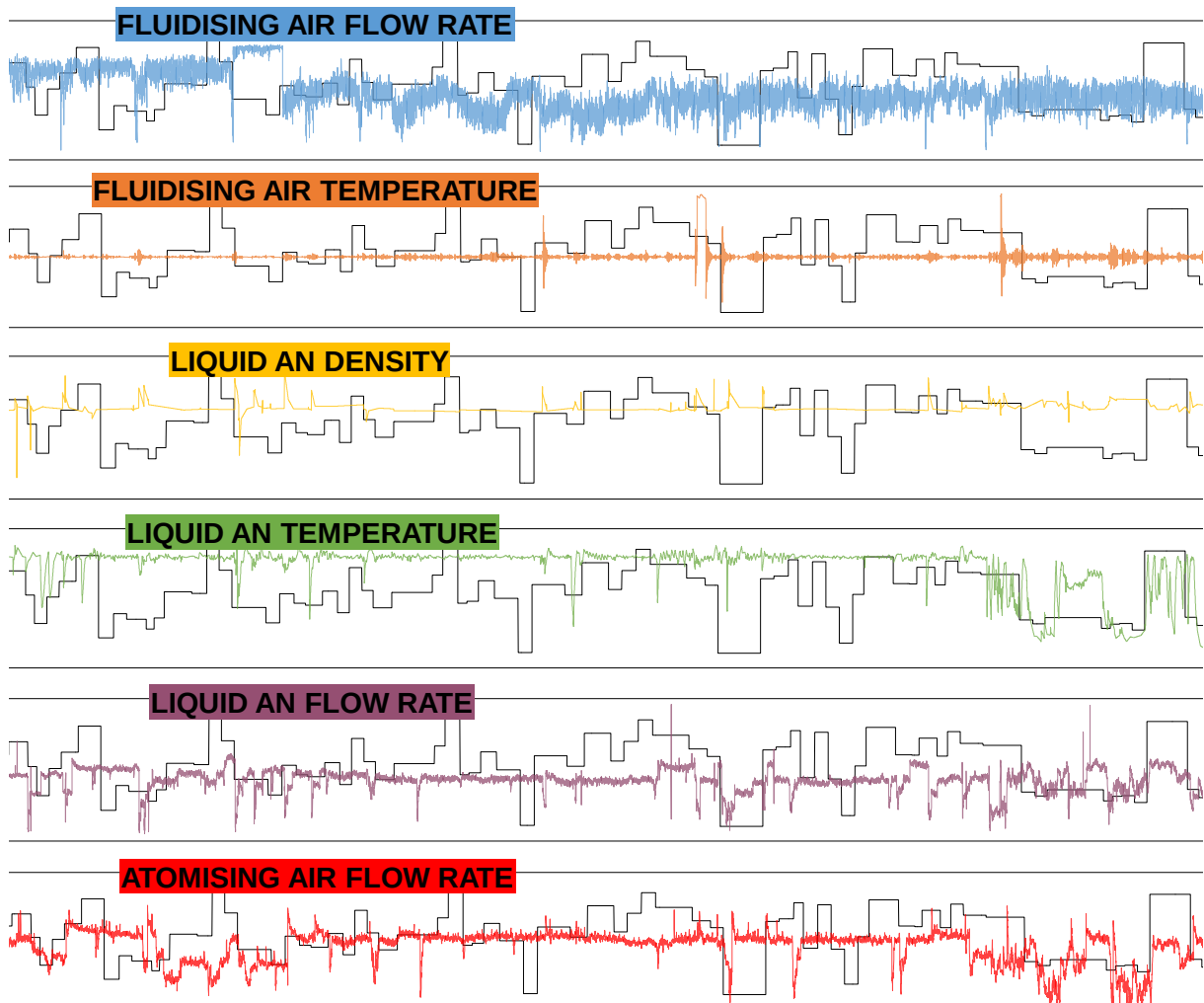


Figure 5.2-1: Partial sets of filtered data

#### *Discrete data*

The discrete data set consists of 156 data points. Figure 5.2-2 shows a part of the data, while the full graph is given in Appendix C.2.3. For the discrete data, the input values (coloured lines) do not significantly vary, however, the output value (again the black line in each case) changes continuously. It can qualitatively be seen that there is not a good correlation between input and output values.

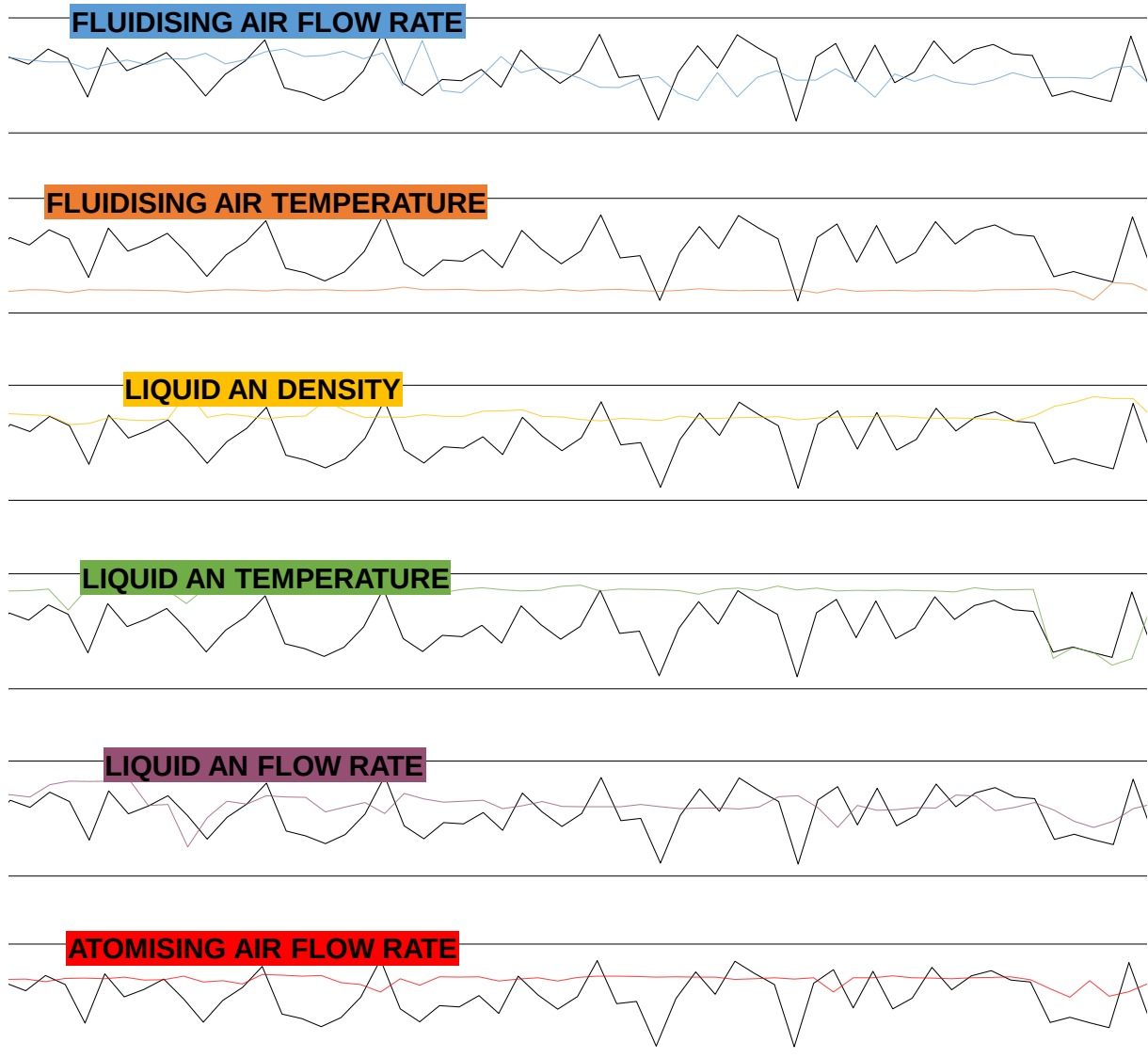


Figure 5.2-2: Partial discrete data

### *Constant-time data*

The constant-time data sprouted from the simulation of an ideal quality control situation where samples are taken at constant intervals. As shown in Figure 5.2-3 (with the full data set in Appendix C.2.4), the time between consecutive oil absorption values is kept constant at 10 minutes. The dataset consisted of 1 560 data points. The constant-time data shows a more apparent change in several input values, which qualitatively seems to correlate better with the output value. The input value with the most noise is still fluidising air flow rate, showing that noise filtering might be beneficial for this data set, necessitating further investigation.

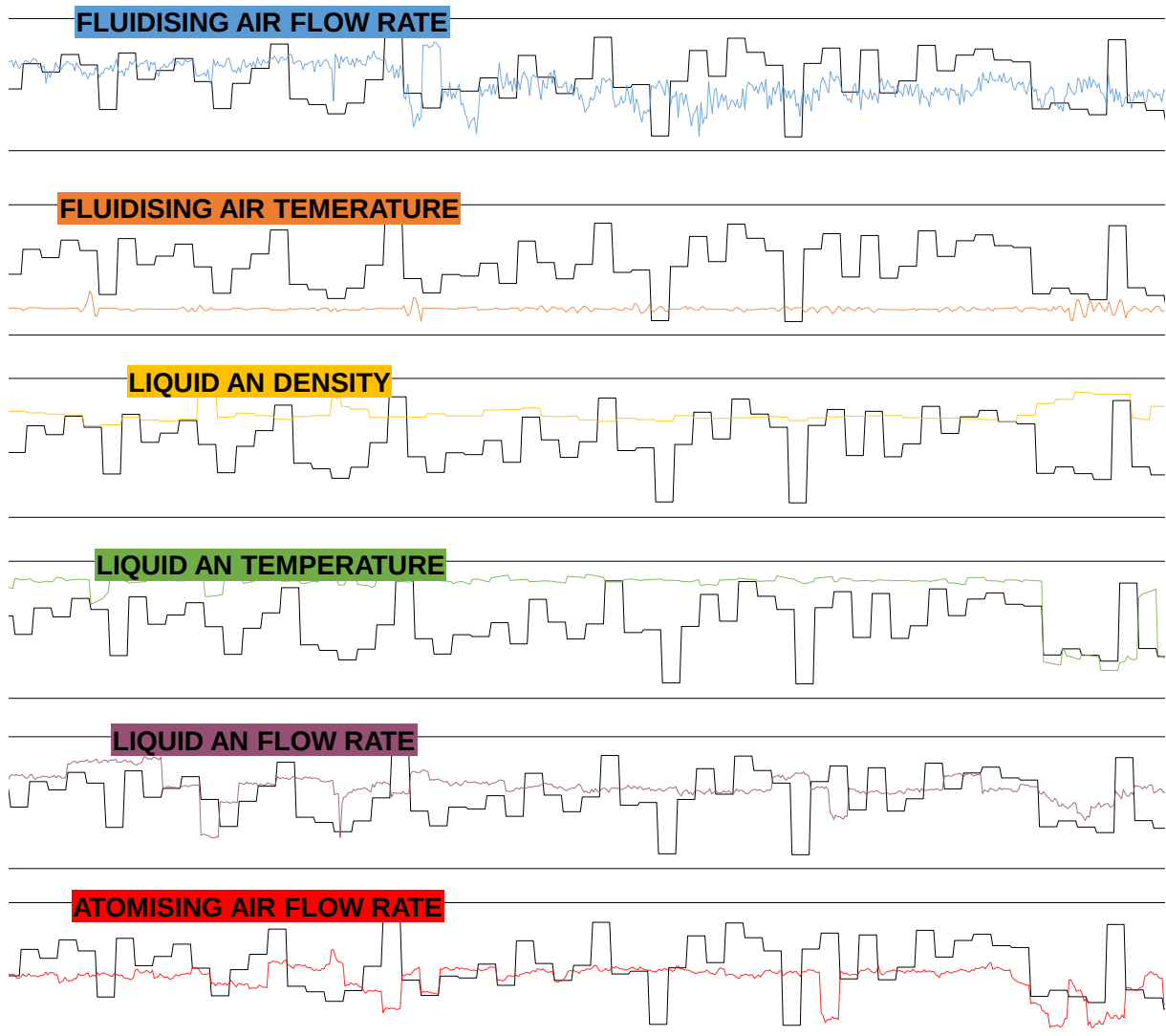


Figure 5.2-3: Partial constant-time data

### 5.2.1.2 Artificial neural network training and testing on the filtered data set

The filtered data set consisted of 33 919 time-stamped data points for each input for training and 8 480 data points for simulation of the neural network. The data division is shown in Appendix F.1.

#### Levenberg-Marquardt training algorithm

The filtered data set was trained with the Levenberg-Marquardt algorithm using different neuron configurations. Table 5.2-2 shows the results of each of the different numbers of neurons in the hidden layer.

**Table 5.2-2: Performance of Levenberg-Marquardt algorithm on filtered data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------------|----------------|---------------|
| <i>5 neurons</i>   | 1.17       | 0.35               | 0.36                 | 0.36           | 0.36          |
| <i>10 neurons</i>  | 1.01       | 0.51               | 0.50                 | 0.51           | 0.51          |
| <i>15 neurons</i>  | 0.97       | 0.56               | 0.54                 | 0.56           | 0.56          |
| <i>25 neurons</i>  | 0.87       | 0.61               | 0.60                 | 0.61           | 0.61          |
| <i>50 neurons</i>  | 0.72       | 0.71               | 0.69                 | 0.68           | 0.70          |
| <i>75 neurons</i>  | 0.71       | 0.73               | 0.71                 | 0.71           | 0.72          |
| <i>100 neurons</i> | 0.74       | 0.73               | 0.69                 | 0.71           | 0.72          |

The mean squared error reduced with an increase in neurons in the hidden layer while the correlation coefficient increased with the increase in neurons. This shows that many neurons perform better than a small number of neurons, which can be attributed to the large amount of data being used for training.

A hidden layer consisting of 75 neurons performed the best when trained with the Levenberg-Marquardt algorithm, resulting in a mean-squared error of 0.706 and a correlation coefficient (R) of 0.722 for all the data. All the relevant regression plots are provided in Appendix C.3 as Figure C.3- and Figure C.3-2. The coefficient of determination ( $R^2$ ) for the training of the 75-neuron network is given as 0.521. This coefficient of determination could be influenced by the large amount of data present, since the data still shows numerous data points that are not close to the trend line.

The simulation of each neural network was done to test whether the neural network can estimate the output variable using unseen data accurately. Like the assessment of the training efficiency of the neural network above, the neural network's simulation ability was analysed using statistical regression analysis. The coefficient of determination for the simulation of the neural network is determined at 0.502, resulting in a correlation coefficient of 0.708. The mean-squared error for the simulation of the network is 0.684. This is close to the correlation coefficient and better than the mean-squared error obtained during training, thus the neural network seems to provide adequate estimations with unseen data for the training it received.

The mean-squared errors and correlation coefficients of each neuron configuration were plotted against the number of neurons in the hidden layers to determine if an optimum number of neurons for the hidden layer can be predicted using both these goodness-of-fit variables. The data is presented in Figure 5.2-4.

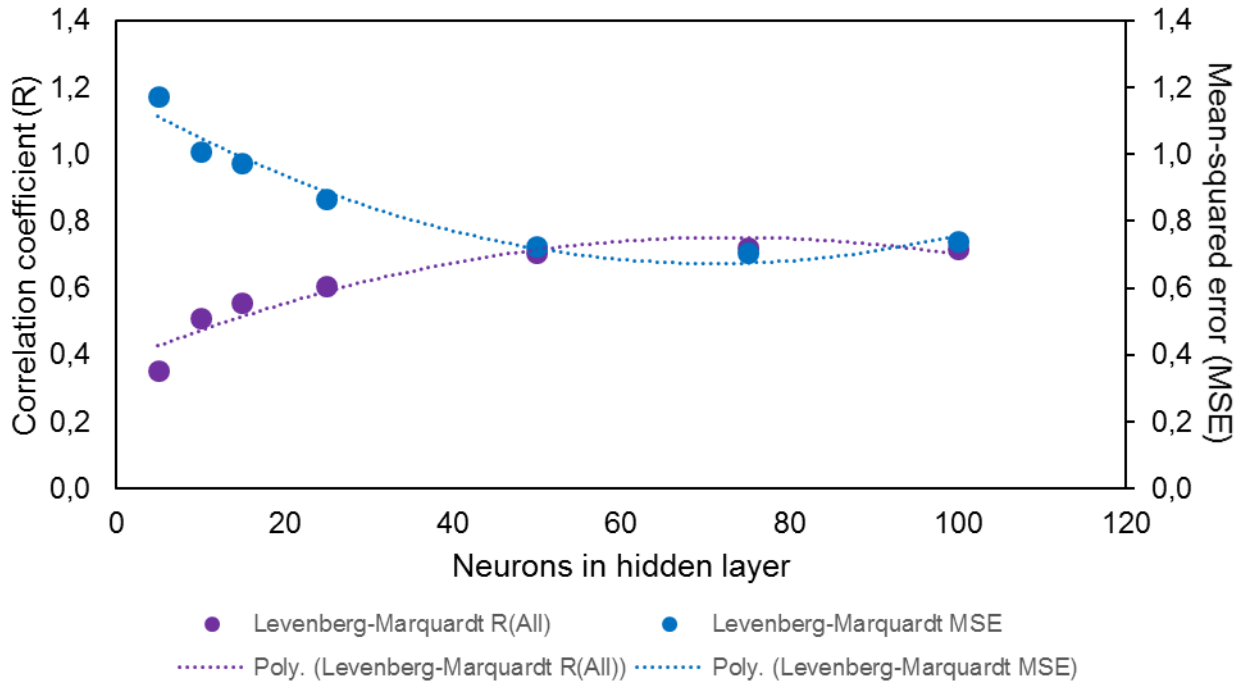


**Figure 5.2-4: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training algorithm on filtered data**

As seen with the polynomial fitting trend lines in the graph, the trends are quadratic and have a minimum (for MSE) and maximum (for R) between 50 and 75. The equation of the trend line of the correlation coefficient is given in Equation 5.2-1:

$$y = (-7.03 \times 10^{-5})x^2 + (1.02 \times 10^{-2})x + 3.77 \times 10^{-1} \quad 5.2-1$$

The maximum for the abovementioned quadratic equation is calculated to be at 73 neurons. The equation of the trend line for the mean-squared error is given in Equation 5.2-2:

$$y = (1.01 \times 10^{-4})x^2 + (-1.44 \times 10^{-2})x + 1.18 \quad 5.2-2$$

The minimum for the abovementioned quadratic equation is calculated to be at 71 neurons. This results in an average of 72 neurons. A 72-neuron neural network was tested under the same conditions as the first neural networks. The training regression plot is given in Figure 5.2-5.

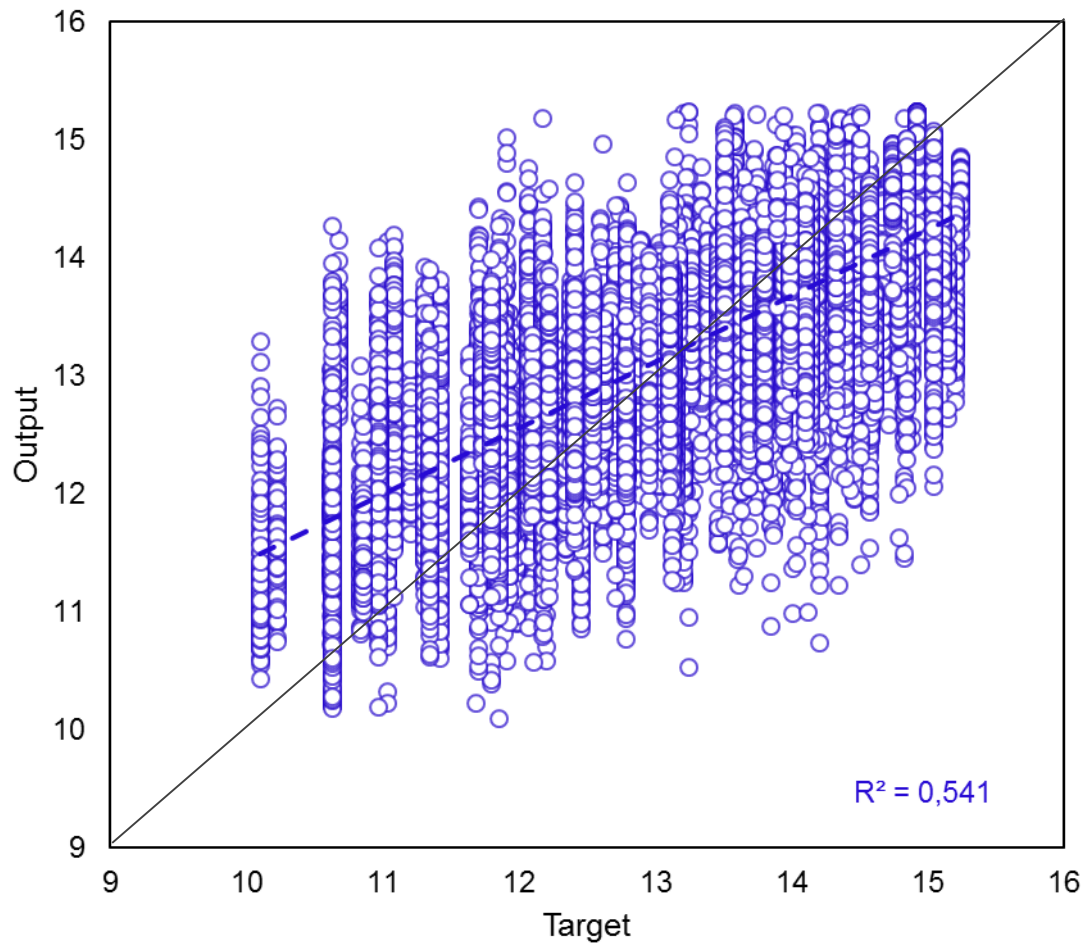


**Figure 5.2-5: Regression plot of the training of a 72-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data**

The coefficient of determination is 0.541. Figure C.3-3 in Appendix C.3 shows all the other regression plots. The mean-squared error was calculated as 0.662 and the correlation coefficient of all the data was determined at 0.736, which shows that the 72-neuron neural network performed better than the 75-neuron neural network.

The regression plot of the simulation of the 72-neuron neural network is given in Figure 5.2-6:

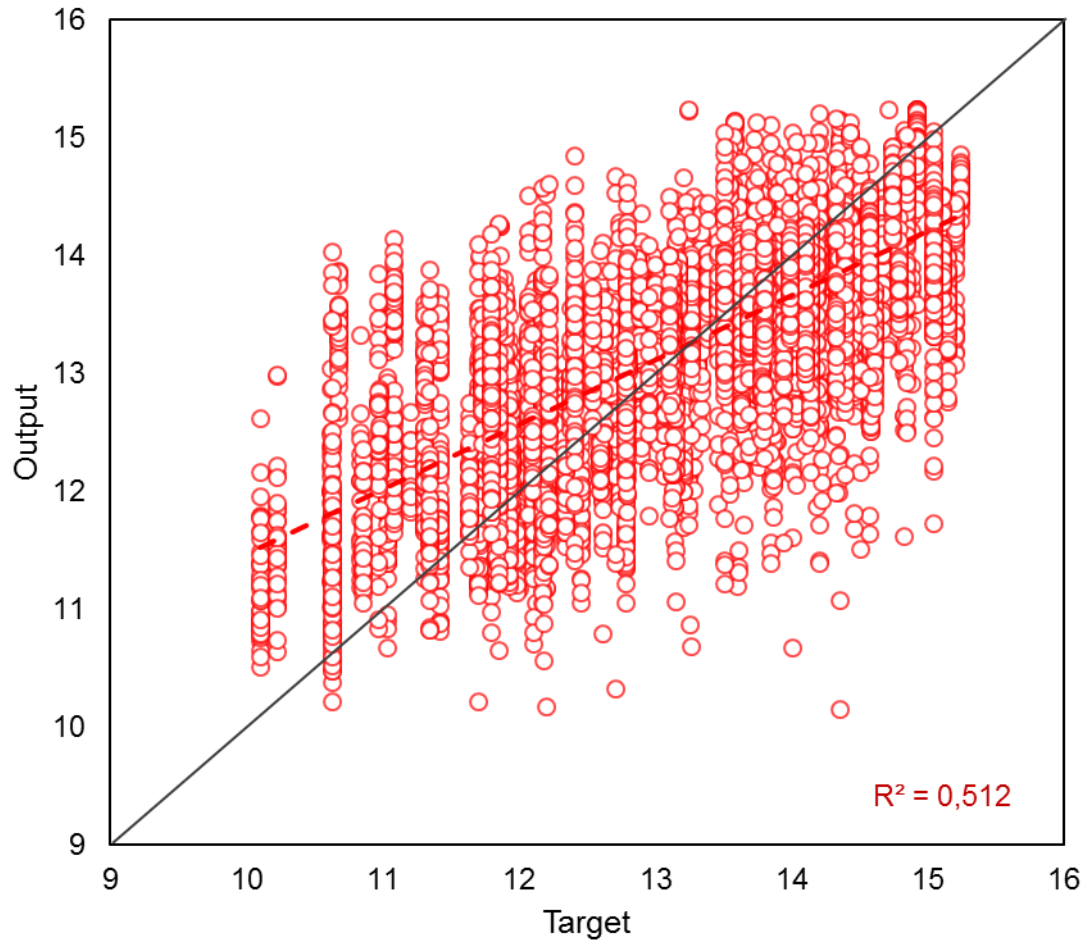


**Figure 5.2-6: Regression of the simulation of a 72-neuron network trained with the Levenberg-Marquardt algorithm using filtered data**

The coefficient of determination of 0.512 shows that the 72-neuron neural network simulated better than all the other neuron configurations. The mean-squared error and the correlation coefficient of the simulation of the neural network were calculated as 0.716 and 0.670 respectively. The performance of the simulation is close to the performance of the training data, showing that the neural network performs well when fed with unseen data.

The mean-squared error and correlation coefficient of the 72-neuron neural network was added to the data presented in Figure 5.2-4 to establish if the trend line predictions can be considered accurate. This data is shown in Figure 5.2-7.

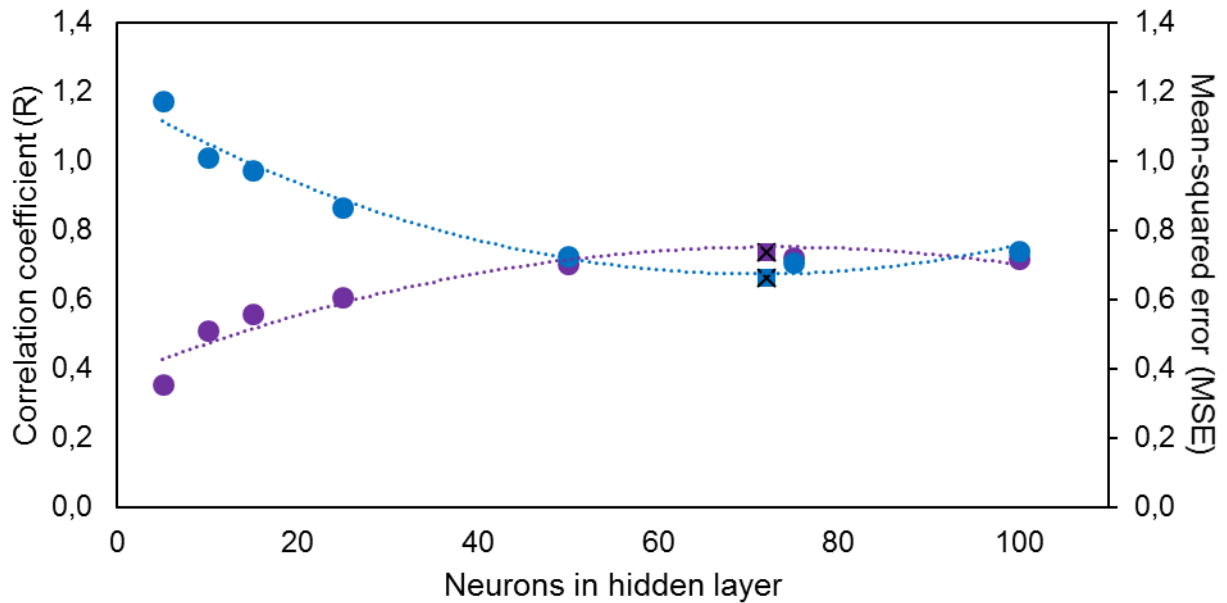


**Figure 5.2-7: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training and filtered data; including the best-case neuron configuration**

The figure shows that the 72-neuron neural network performed as predicted in terms of the mean-squared error and close to prediction in terms of the correlation coefficient.

It was thus found that the best performing neural network using the Levenberg-Marquardt training algorithm is the neural network with 72 neurons in the hidden layer. An indication of the model's performance is depicted in Figure 5.2-8 showing the plant-generated porosity data points together with the corresponding artificial neural network estimated porosity data points. The figure shows only a part of the data for training as well as simulation, with the full graph available as raw data in Appendix F.1.

It is observed in the model performance graph below that neural network output data varies a lot compared to the plant generated data. This shows that the neural network does not train sufficiently, which is also the reason for the large fluctuation in the simulation data. The amount that the neural network outputs vary, stays constant, which shows that the neural network works within boundaries, however, does not train adequately within these input boundaries to be very accurate in its estimation. The reason for this could be that the filtered data set contains a vast amount of data and thus could contain 'false' as well as duplicate examples of input combinations with different output values, which means the neural network is fed with highly variable combinations during the training phase, resulting in a variable and sometimes erratic simulation behaviour of the neural network, as is observed in the yellow data points in the figure below.

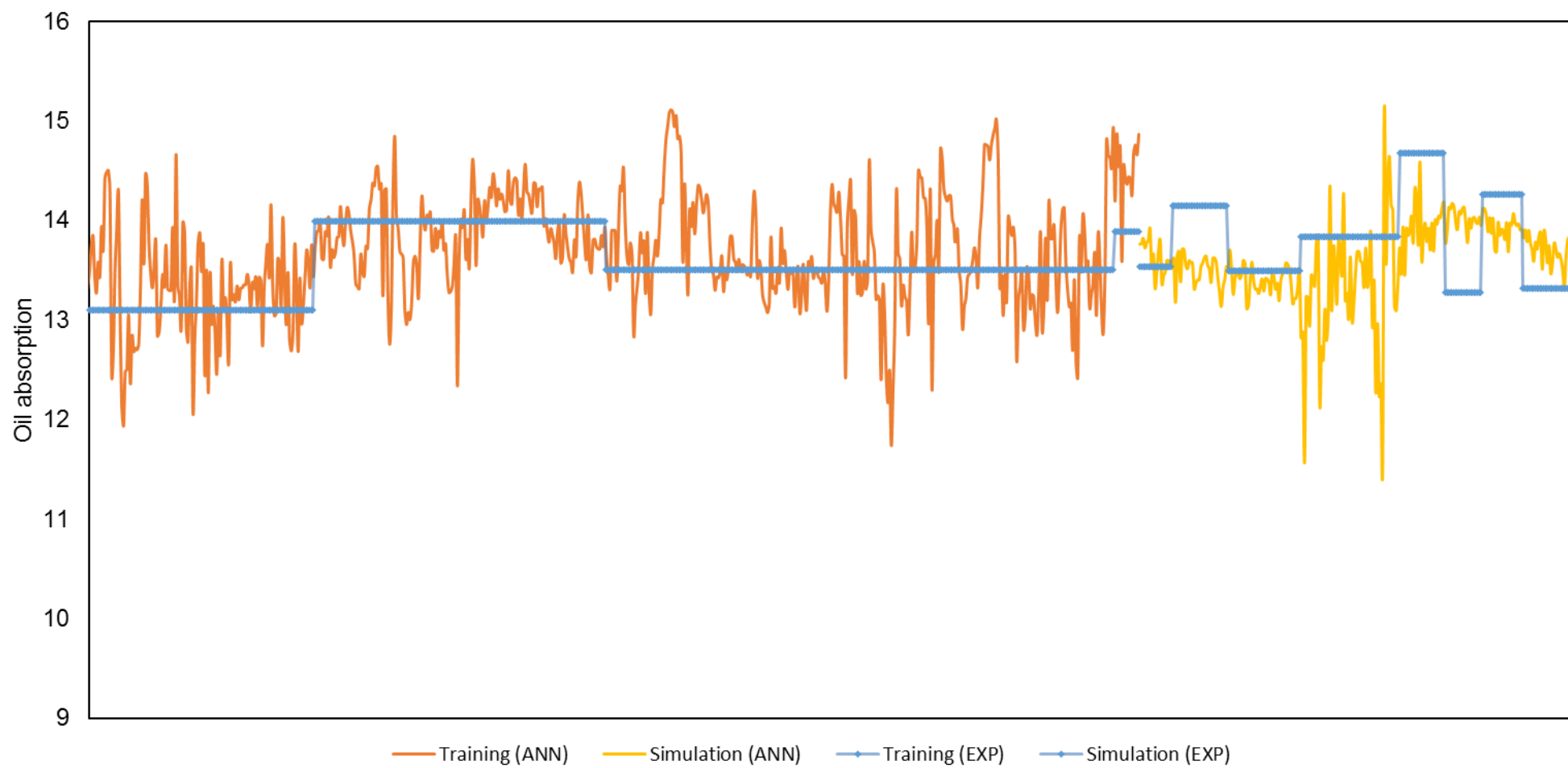


Figure 5.2-8: Model performance of a 72-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data

Bayesian regularisation training algorithm

Similar to the study described above for the Levenberg-Marquardt training algorithm, different neuron configurations were tested on the Bayesian regularisation training algorithm, with the results shown in Table 5.2-3.

**Table 5.2-3: Performance of Bayesian regularisation algorithm on filtered data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------|---------------|
| <i>5 neurons</i>   | 1.17       | 0.38               | 0.347          | 0.37          |
| <i>10 neurons</i>  | 0.98       | 0.53               | 0.52           | 0.53          |
| <i>15 neurons</i>  | 0.94       | 0.56               | 0.54           | 0.56          |
| <i>25 neurons</i>  | 0.81       | 0.63               | 0.63           | 0.63          |
| <i>50 neurons</i>  | 0.67       | 0.71               | 0.71           | 0.71          |
| <i>75 neurons</i>  | 0.59       | 0.75               | 0.73           | 0.75          |
| <i>100 neurons</i> | 0.53       | 0.78               | 0.76           | 0.78          |
| <i>125 neurons</i> | 0.47       | 0.81               | 0.78           | 0.81          |
| <i>150 neurons</i> | 0.53       | 0.78               | 0.76           | 0.78          |

An increase in neurons in the hidden layer brought about an increase in the performance of the neural network, since the mean-squared error decreased, and the correlation coefficient increased up to a certain point.

The best performance was obtained from the 125-neuron neural network. It resulted in a mean-squared error of 0.469 and a correlation coefficient of 0.806. The correlation found for the training data is good, with a coefficient of determination ( $R^2$ ) of 0.649. Figure C.3-4 in Appendix C.3 contains all other relevant regression plots for this training algorithm. This shows that the neural network trained with the Bayesian regularisation algorithm resulted in a good training performance. The Bayesian regularisation algorithm does not make use of a validation step as in the case of the Levenberg-Marquardt algorithm, hence the absence of this data.

The coefficient of determination for the simulation of the abovementioned neural network is 0.609, yielding a correlation coefficient of 0.780. This is close to what was obtained in training (0.806), resulting in a good validation of the neural network. The mean-squared error for the simulation of the neural network was calculated to be 0.538. This shows that the neural network performed as well with unseen data as with training data.

Figure 5.2-9 shows the correlation coefficient and mean-squared error of each of the assessed neuron configurations.

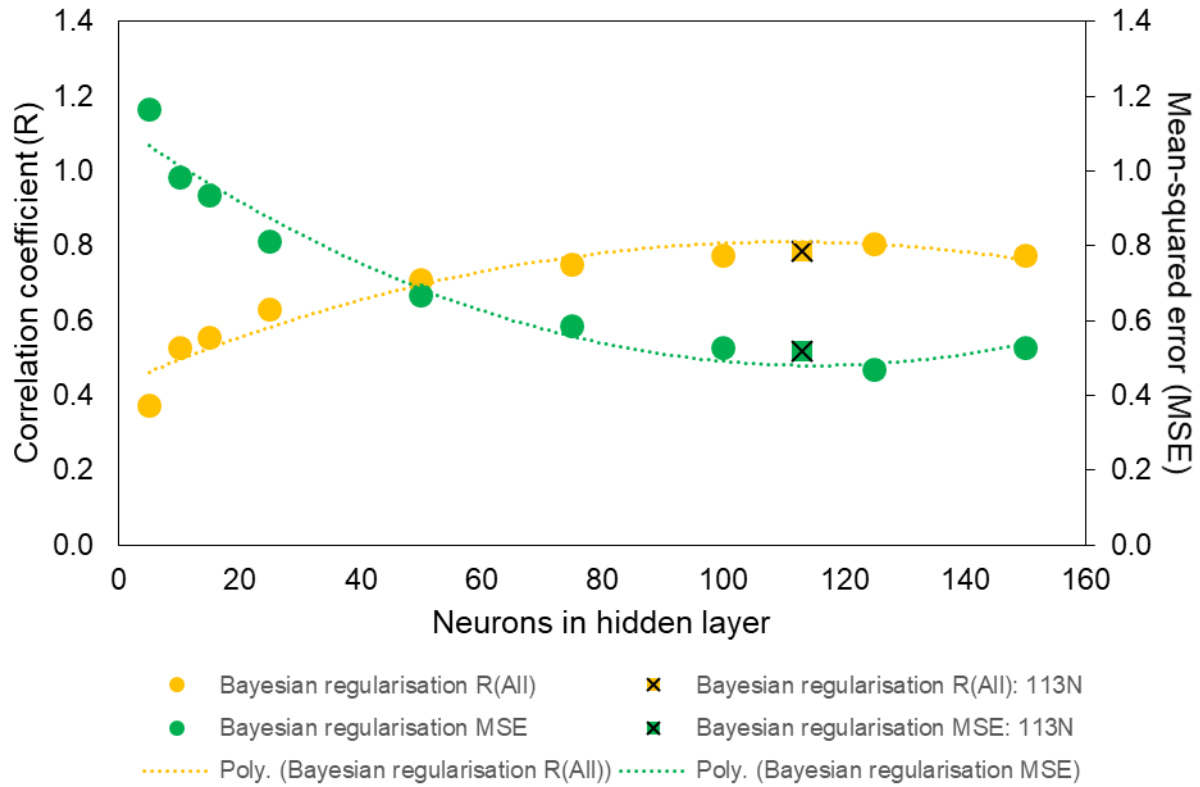


**Figure 5.2-9: MSE-values and R-values for neuron-configurations using the Bayesian regularisation training and filtered data**

The trend lines for the 5-neuron to 150-neuron network were used to obtain the best neuron configuration. The trend line equation for the mean-squared error is given in Equation 5.2-3:

$$y = (4.84 \times 10^{-5})x^2 + (-1.11 \times 10^{-2})x + 1.12 \quad 5.2-3$$

The minimum for this equation is calculated to be at 111 neurons. Equation 5.2-4 gives the trend line equation for the correlation coefficient:

$$y = (-3.11 \times 10^{-5})x^2 + (6.88 \times 10^{-3})x + 4.30 \times 10^{-1} \quad 5.2-4$$

The maximum of this equation yielded in 115 neurons. This results in an average neuron configuration of 113. The configuration was tested, resulting in the regression plot given in Figure 5.2-10.

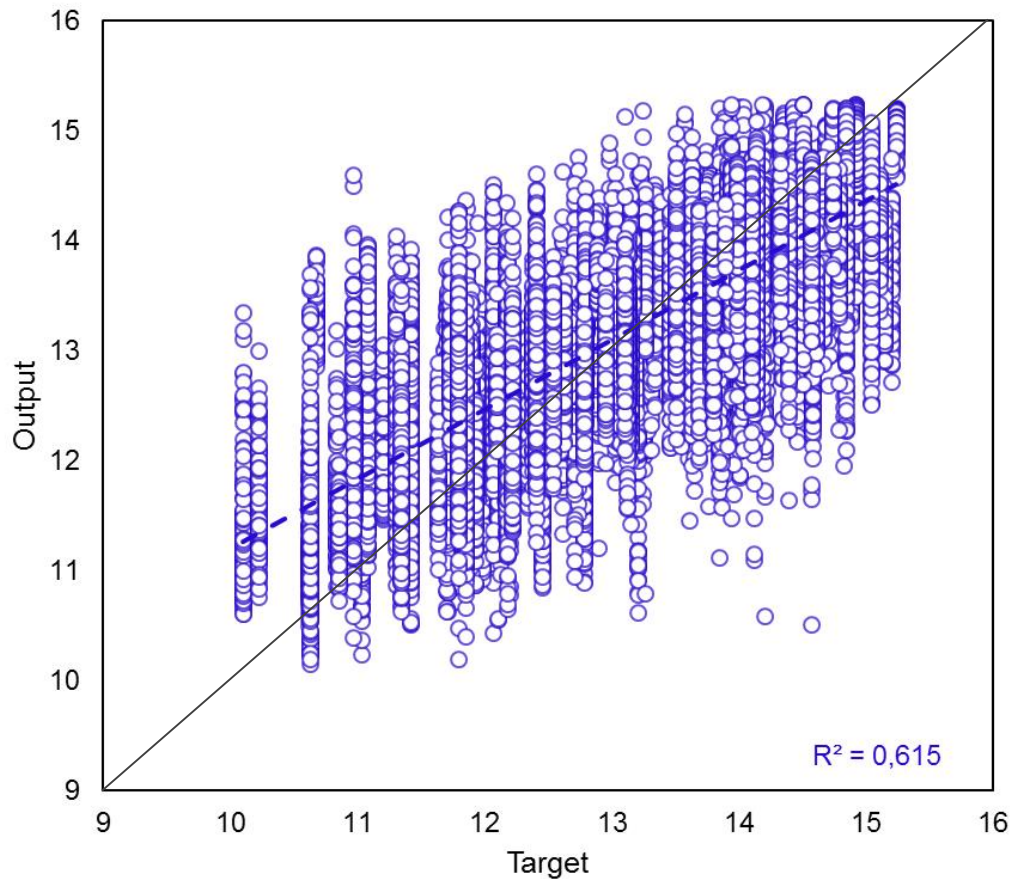


**Figure 5.2-10: Regression plot of the training of a 113-neuron neural network trained with the Bayesian regularisation algorithm using filtered data**

The coefficient of determination of the training of the 113-neuron network is 0.615. Figure C.3-6 in Appendix C.3 shows the regression obtained. The mean-squared error was calculated as 0.519, while the correlation coefficient of all the training data was calculated as 0.736. This shows that the 113-neuron performed worse than expected; not even exceeding the performance of the 125-neuron network. The simulation of the network is shown in Figure 5.2-11.

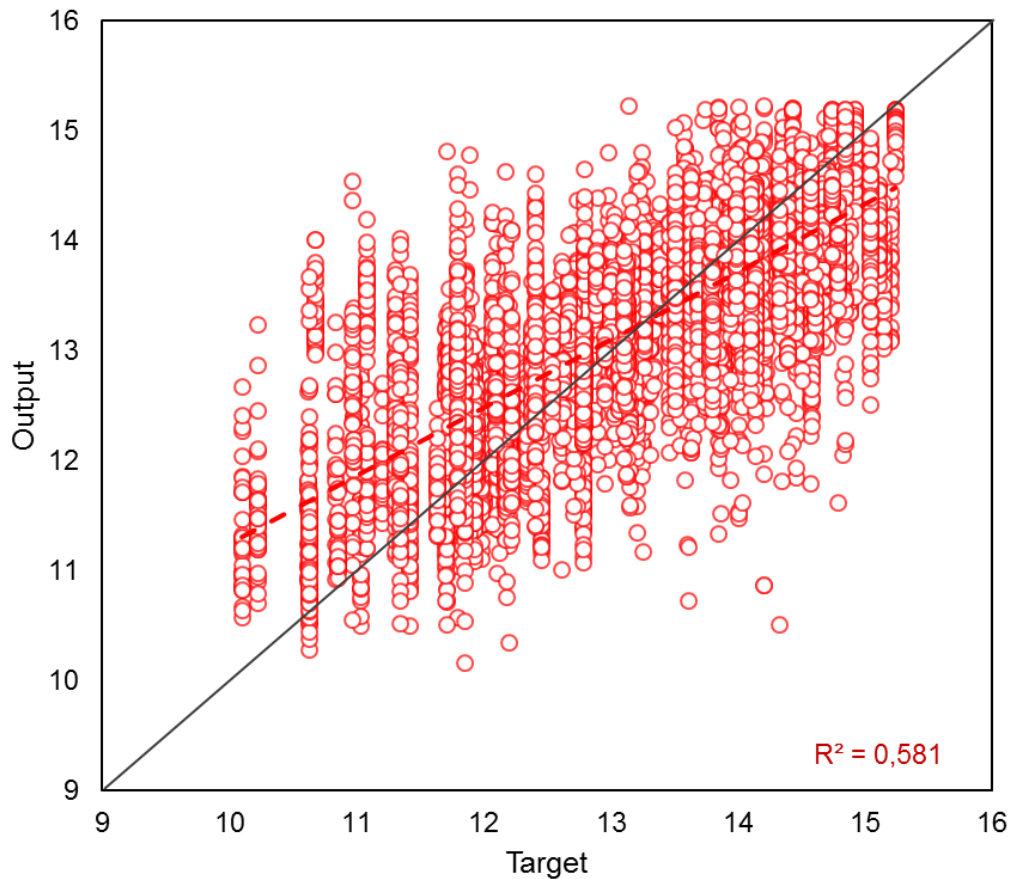


**Figure 5.2-11: Regression of the simulation of a 113-neuron network trained with the Bayesian regularisation algorithm using filtered data**

The simulation resulted in a coefficient of determination of 0.581. This results in a correlation coefficient of 0.762, which is not as close to the training correlation coefficient as it should be to be considered a good performing neural network. The lower correlation coefficient could be due to the ANN training that was not very efficient, as observed in Figure C.3-6 in Appendix C.3, which shows that the training data does not result in a good correlation coefficient. Therefore, the neural network cannot (and does not) perform well with unseen data.

Figure 5.2-9 confirms that the 113-neuron network did not perform as predicted by the trend lines obtained from the original neural networks. This results in the 125-neuron network performing the best when using the Bayesian regularisation training algorithm and filtered data. The model performance of the 125-neuron neural network is given in Figure 5.2-12. The figure shows a partial set of the data, with the full data set provided as raw data in Appendix F.1. As with the best performing neural network when using the Levenberg-Marquardt algorithm, the data shows a lot of variation due to a high number of variable combinations in the training data.

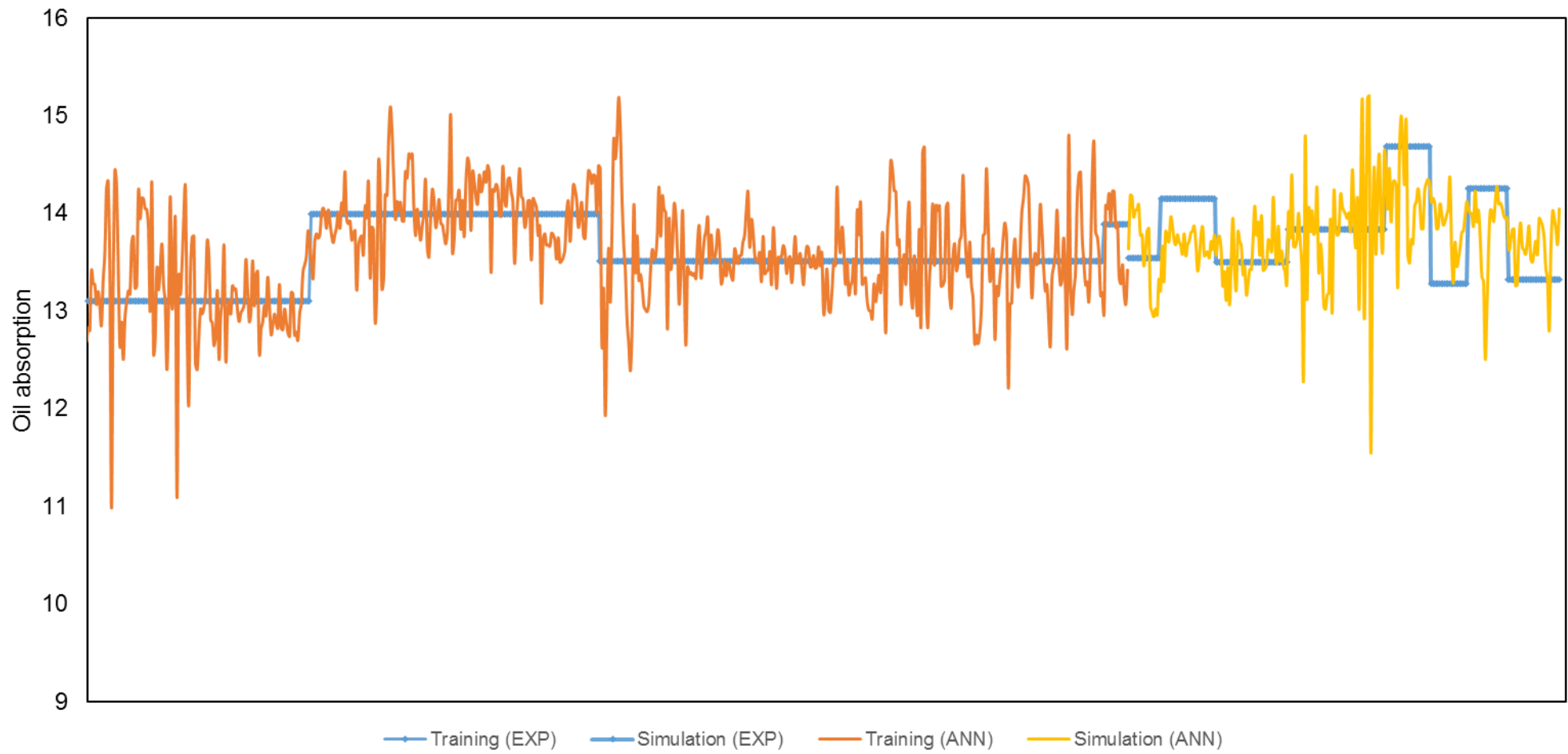


Figure 5.2-12: Model performance of a 125-neuron neural network trained with the Bayesian regularisation algorithm using filtered data

### Training algorithm comparison for the filtered data set

The comparison of two training algorithms, i.e. the Levenberg-Marquardt and Bayesian regularisation algorithms, is based on the same performance evaluations as for the number of neurons in the hidden layer. Table 5.2-4 lists the highest performing neural networks using the filtered data set and both Levenberg-Marquardt and Bayesian regularisation algorithms.

**Table 5.2-4: Comparison of training algorithms using filtered data**

|                                                   | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|---------------------------------------------------|------------|--------------------|----------------------|----------------|---------------|
| <b>Levenberg-Marquardt training algorithm</b>     |            |                    |                      |                |               |
| <i>50 neurons</i>                                 | 0.72       | 0.71               | 0.69                 | 0.68           | 0.70          |
| <i>72 neurons</i>                                 | 0.66       | 0.74               | 0.71                 | 0.72           | 0.74          |
| <i>75 neurons</i>                                 | 0.71       | 0.73               | 0.71                 | 0.71           | 0.72          |
| <b>Bayesian regularisation training algorithm</b> |            |                    |                      |                |               |
| <i>100 neurons</i>                                | 0.53       | 0.78               |                      | 0.76           | 0.78          |
| <i>113 neurons</i>                                | 0.52       | 0.79               |                      | 0.77           | 0.78          |
| <i>125 neurons</i>                                | 0.47       | 0.81               |                      | 0.78           | 0.81          |

As can be seen in the table above, the Bayesian regularisation training algorithm performed better than the Levenberg-Marquardt training algorithm, irrespective of the neuron configurations selected. The performance of the abovementioned neural networks is, however, not satisfactory for implementation since the variations in the ANN output values exceed the variations in the actual plant measurements, as is clearly observed in both Figure 5-2.12 and Figure 5.2-21, as well as due to a relatively low correlation coefficient and a relatively high mean-squared error as compared to performance values obtained later in this study.

The artificial neural network that performed the best with the filtered data is the 125-neuron neural network trained with the Bayesian regularisation training algorithm.

#### **5.2.1.3 Artificial neural network training and testing on the discrete data set**

The discrete data set consisted of 125 data points for training and 31 data points for simulating the neural network. The data is available in Appendix F.1.

Levenberg-Marquardt training algorithm

The discrete dataset was tested on different neural network configurations. Table 5.2-5 lists the performance evaluation results of each neuron configuration.

**Table 5.2-5: Performance of Levenberg-Marquardt algorithm on discrete data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------------|----------------|---------------|
| <i>5 neurons</i>   | 0.97       | 0.39               | 0.56                 | 0.35           | 0.41          |
| <i>10 neurons</i>  | 0.72       | 0.55               | 0.34                 | 0.23           | 0.48          |
| <i>15 neurons</i>  | 0.71       | 0.50               | 0.59                 | 0.55           | 0.51          |
| <i>25 neurons</i>  | 0.93       | 0.37               | 0.44                 | 0.52           | 0.40          |
| <i>50 neurons</i>  | 0.62       | 0.27               | 0.49                 | 0.44           | 0.32          |
| <i>75 neurons</i>  | 0.88       | 0.25               | 0.17                 | 0.43           | 0.29          |
| <i>100 neurons</i> | 0.56       | 0.53               | 0.80                 | 0.51           | 0.56          |

There is no apparent trend in the mean-squared error or the correlation coefficient for the discrete data set which rules out a possible analysis to find the best performing neuron configuration through statistical regression techniques (i.e. a quadratic fit) as in the previous data set.

Although a hidden layer with 100 neurons fared the best with a mean-squared error of 0.564 and a correlation coefficient of 0.559, it did not perform good at all. Figure 5.2-13 shows the regression plot for the training and simulation of the 100-neuron network, while Figure C.3-7 in Appendix C.3 shows the regression plots obtained for only the training phase. Figure C.3-7 shows that the correlation of validation data is the best in this neural network. This can be problematic, since the training validation is not considered good. As the neural network is set to stop training after the validation criteria is met, this neural network appears to suffer from premature training abandonment. Further training would either make the neural network better or over train the neural network, in which case it would perform worse.

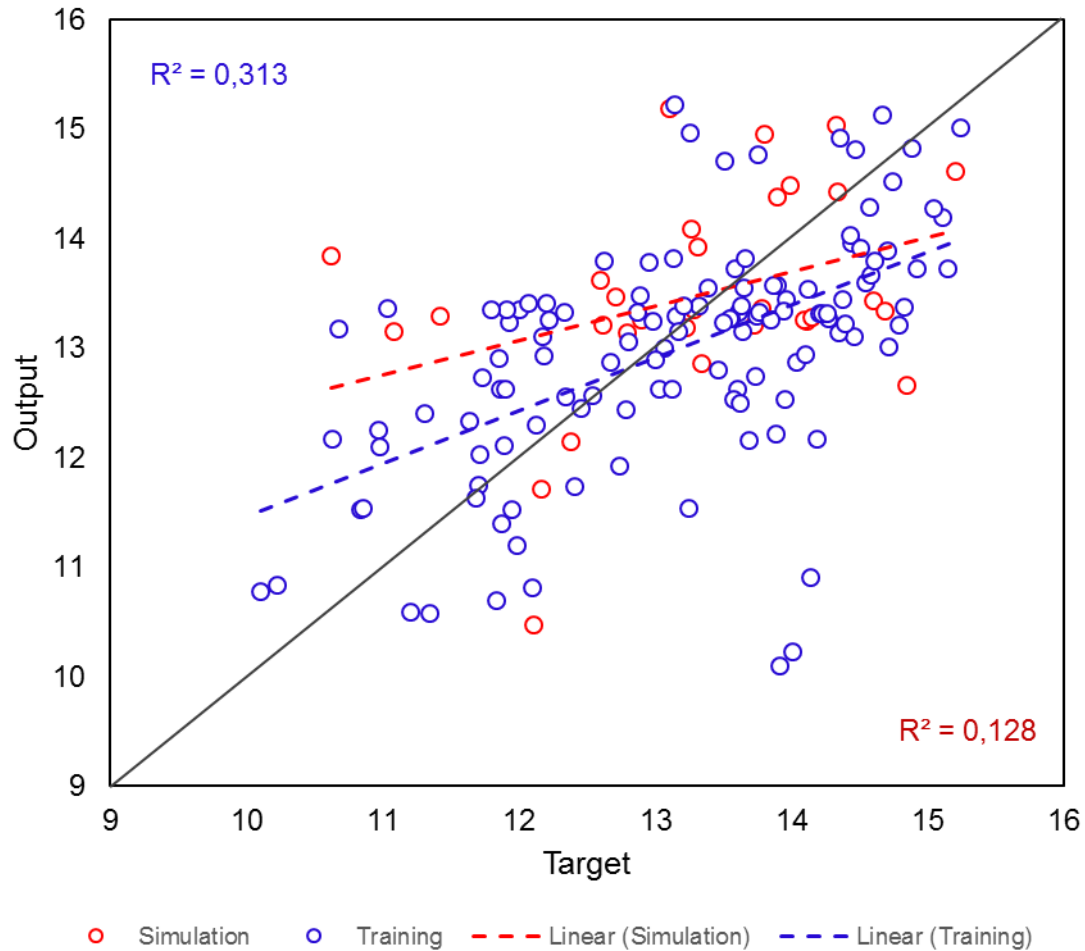


**Figure 5.2-13: Regression plot of the training and simulation of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using discrete data**

The coefficient of determination for the training of the network is given as 0.313, which shows that the training fared poorly. The coefficient of determination for the simulation of the 100-neuron neural network is 0.128, resulting in a correlation coefficient of 0.358. This is far from the training correlation coefficient of all the data, namely 0.559. The mean-squared error for the simulation of the neural network was calculated to be 1,366. This also shows that the data training was not sufficient, as the neural network very poorly predicts unseen data.

The model performance of the 100-neuron network using the Levenberg-Marquardt training algorithm and discrete data is given in Figure 5.2-14. The graph shows that the training of the neural network follows the trend of the plant-generated quality (porosity) data in most cases, however, frequently not reaching the specific value. This is the reason for the large mean-squared error value for the training of the neural network. The neural network did not train or simulate well at all, showing that the data set is not adequate for training of a neural network.

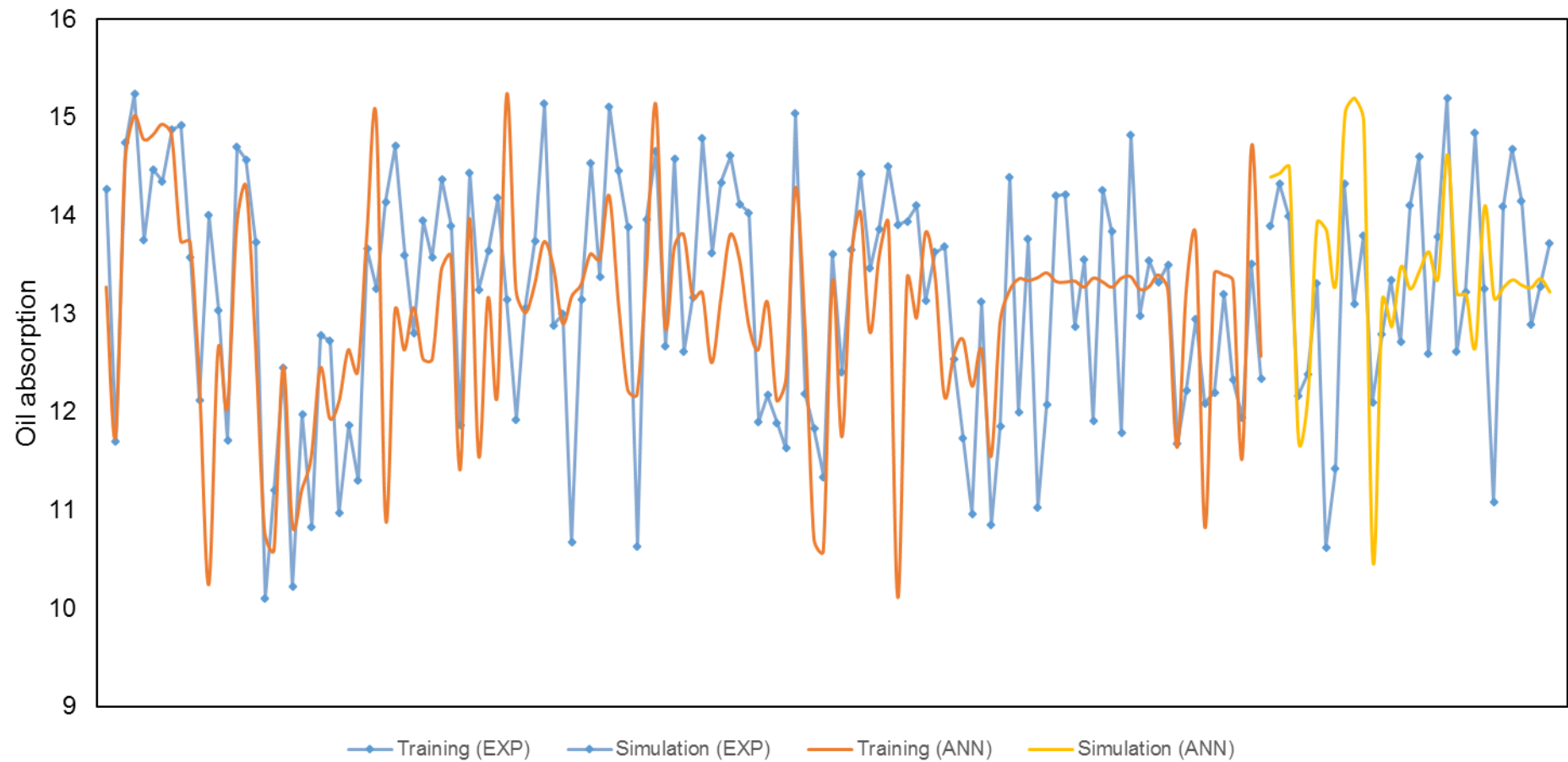


Figure 5.2-14: Model performance of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using discrete data

Bayesian regularisation training algorithm

The performance results obtained for the Bayesian regularisation training on discrete data are tabulated in Table 5.2-6 below.

**Table 5.2-6: Performance of Bayesian regularisation algorithm on discrete data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------|---------------|
| <i>5 neurons</i>   | 1.15       | 0.34               | 0.01           | 0.25          |
| <i>10 neurons</i>  | 1.57       | 0.17               | 0.34           | 0.18          |
| <i>15 neurons</i>  | 1.34       | 0.16               | 0.45           | 0.22          |
| <i>25 neurons</i>  | 1.39       | 0.10               | 0.48           | 0.17          |
| <i>50 neurons</i>  | 1.47       | 0.24               | 0.17           | 0.23          |
| <i>75 neurons</i>  | 1.49       | 0.23               | 0.10           | 0.21          |
| <i>100 neurons</i> | 1.06       | 0.30               | 0.24           | 0.25          |

The mean-squared errors and correlation coefficients of the neural networks show that none of the above configurations performed well. The two best neural networks will be discussed for comparison purposes even though they are not considered usable neural networks. As with the Levenberg-Marquardt training algorithm, no apparent trend is observed between number of neurons in the hidden layer and performance of the neural network.

The best performing neural network among the results shown in the table above, is the 100-neuron neural network. The mean-squared error was calculated as 1.064, which is considered high. The correlation coefficient of all the data is 0.254, which is too low to consider the neural network as a well performing neural network. Figure C.3-8 shows the regression plots of the 100-neuron neural network using Bayesian regularisation as training method.

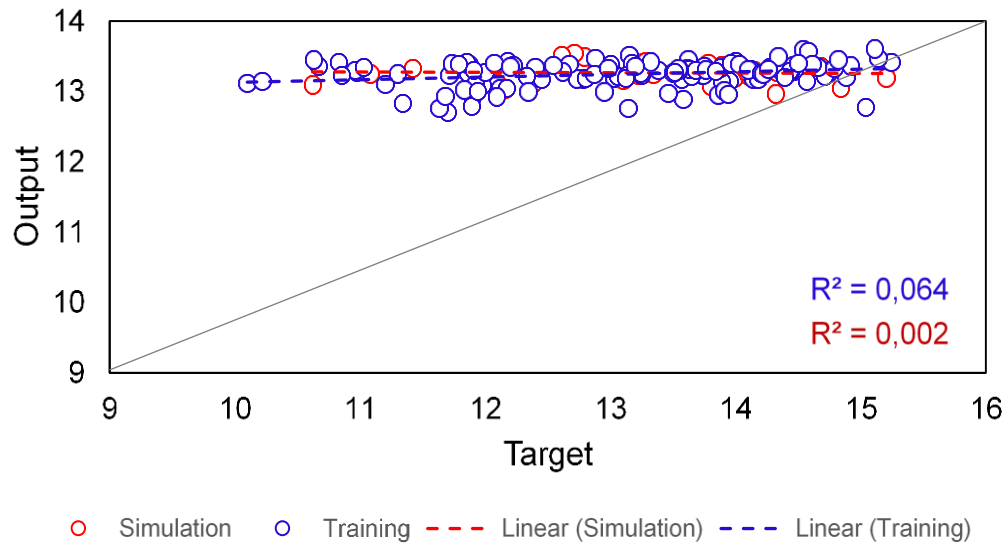


**Figure 5.2-15: Regression plot of the training and simulation of a 100-neuron neural network trained with the Bayesian regularisation algorithm using discrete data**

The reason for the poor performance from the Bayesian regularisation training algorithm was the weight adjustment. The weights from a single input converged to a single number resulting in a single output for all input values, as shown above. The neural network could thus not train sufficiently due to an insufficient number of samples provided for training.

The simulation of the neural networks performed similar to the training, i.e. converging the output values to a single number, which is to be expected. The regression plot for the simulation of the abovementioned neural network is given in Figure 5.2-15. The coefficient of determination for the simulation is 0.0018, resulting in a correlation coefficient of -0.042. The simulation mean-squared error is 1.202. The correlation coefficient and mean-square error is far from that of the training dataset, thus the neural network did not predict unseen data well.

The model performance of the 100-neuron network using the Bayesian regularisation training algorithm and discrete data is shown in Figure 5.2-16. The model performance graph shows that the discrete data set did not train well at all with the Bayesian regularisation training algorithm and can be attributed to the small number of data points. The neural network output value converged to 13 in each instance, which is the average value of all output values. This shows that the neural network could not train and thus also not simulate unseen data.



Figure 5.2-16: Model performance of a 100-neuron neural network trained with the Bayesian regularisation algorithm using discrete data

Training algorithm comparison for the discrete data set

The performance figures for the best performing neural networks of both the Levenberg-Marquardt and Bayesian regularisation methods on the discrete data set are given in Table 5.2-7.

**Table 5.2-7: Comparison of training algorithms using discrete data**

|                                                   | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|---------------------------------------------------|------------|--------------------|----------------------|----------------|---------------|
| <b>Levenberg-Marquardt training algorithm</b>     |            |                    |                      |                |               |
| <i>15 neurons</i>                                 | 0.71       | 0.50               | 0.59                 | 0.55           | 0.51          |
| <i>100 neurons</i>                                | 0.56       | 0.53               | 0.80                 | 0.51           | 0.56          |
| <b>Bayesian regularisation training algorithm</b> |            |                    |                      |                |               |
| <i>5 neurons</i>                                  | 1.15       | 0.34               |                      | 0.01           | 0.25          |
| <i>100 neurons</i>                                | 1.06       | 0.30               |                      | 0.24           | 0.25          |

The table shows that the Levenberg-Marquardt training algorithm outperformed the Bayesian regularisation training algorithm by far. The small dataset used for the discrete data is accommodated more by the Levenberg-Marquardt training algorithm, hence the better performance with this dataset.

The best performing artificial neural network using the discrete data was the 100-neuron neural network using the Levenberg-Marquardt algorithm for training. The performance of this neural network is still, however, very far from an acceptable performance when compared to the performance of other configurations studied.

#### **5.2.1.4 Artificial neural network training and testing on the constant-time data set**

The constant-time data forms a sub set of the filtered data set. The time before each new oil absorption value was first taken as 10 minutes. The dataset was divided into 1 364 data points per input parameter for training and 341 data points for simulation. The dataset is given in Appendix F.1.

Levenberg-Marquardt training algorithm

The results obtained by training the constant-time data with the Levenberg-Marquardt training algorithm is tabulated in Table 5.2-8.

**Table 5.2-8: Performance of Levenberg-Marquardt algorithm on constant-time data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------------|----------------|---------------|
| <i>5 neurons</i>   | 1.14       | 0.51               | 0.53                 | 0.38           | 0.50          |
| <i>10 neurons</i>  | 1.04       | 0.35               | 0.43                 | 0.39           | 0.37          |
| <i>15 neurons</i>  | 0.89       | 0.56               | 0.56                 | 0.46           | 0.55          |
| <i>25 neurons</i>  | 0.74       | 0.82               | 0.70                 | 0.69           | 0.78          |
| <i>50 neurons</i>  | 0.71       | 0.82               | 0.71                 | 0.77           | 0.80          |
| <i>75 neurons</i>  | 0.66       | 0.89               | 0.74                 | 0.76           | 0.85          |
| <i>100 neurons</i> | 0.74       | 0.85               | 0.73                 | 0.76           | 0.82          |

As was also observed for the filtered data, the performance of the neural network increases with an increase in neurons in the hidden layers. The performance decreases again at 100 neurons making the 75-neuron network as the best performer amongst the configurations tested with a mean-squared error of 0.656 and correlation coefficient of 0.848. Figure C.3-9 in Appendix C.3 shows the regression plots obtained for the training of this network. The coefficient of determination for the training is obtained at 0.720 and for the simulation of the network the coefficient of determination is 0.637.

The correlation coefficient for the simulation is calculated as 0.798. The mean-squared error for the simulation of the neural network was calculated to be 0.503. This is close to the correlation coefficient and the mean-squared error obtained during the training, showing that the neural network performed as well as possible when presented with unseen data.

The correlation coefficient and mean-squared error for each neuron configuration is given in Figure 5.2-17.

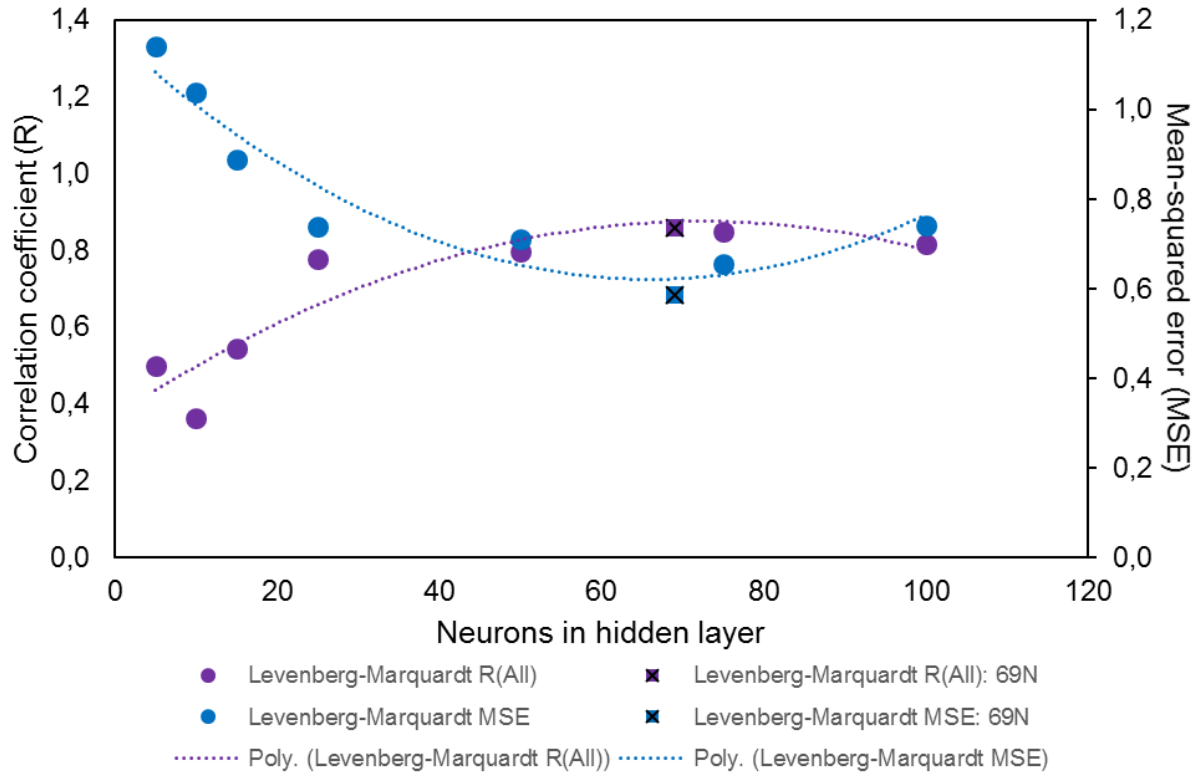


**Figure 5.2-17: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training and constant-time data**

The trend of both the mean-squared error and the correlation coefficient are quadratic with minimum and maximum values between 50 and 75. The trend line equation for the correlation coefficient is given in Equation 5.2-5.

$$y = (-9.7 \times 10^{-5})x^2 + (1.40 \times 10^{-2})x + 3.67 \times 10^{-1} \quad 5.2-5$$

The maximum for the equation is calculated at 72 neurons. The mean-squared error trend line equation is given in Equation

$$y = (1.24 \times 10^{-4})x^2 + (-1.64 \times 10^{-2})x + 1.16 \quad 5.2-6$$

The minimum of the equation is obtained at 66 neurons. This gives an average number of 69 neurons that were assessed under the same conditions. Figure 5.2-18 shows the regression plot for the training and simulation of the 69-neuron network.

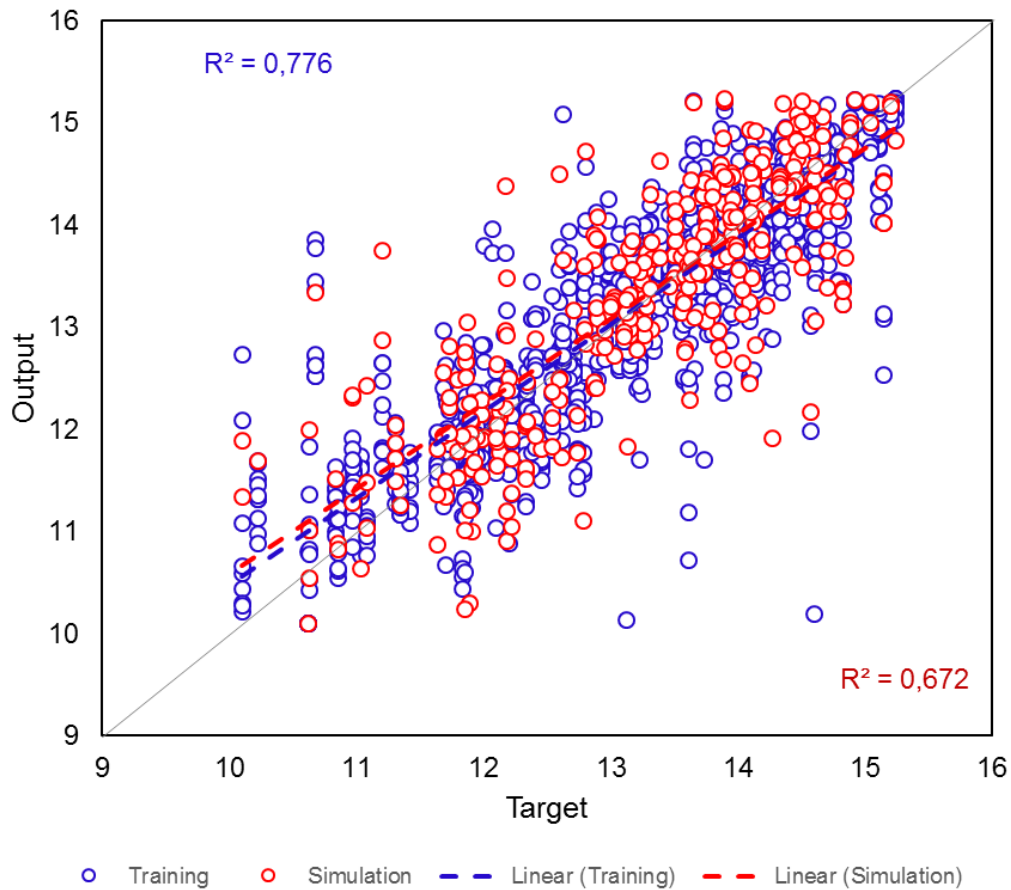


**Figure 5.2-18: Regression plots of the training and simulation of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using constant-time data**

The mean-squared error obtained for the training of the neural network is 0.585. The correlation coefficient for the training is given in Figure C.3-11 in Appendix C.3 as 0.859. The performance is considered good. The 69-neuron network also performed better than the 75-neuron network, as predicted by the trend line.

The regression plot shows the coefficients of determination for both training and simulation as 0.766 and 0.672 respectively. The correlation coefficient of the simulation of the abovementioned network is 0.820. The mean-squared error was calculated to be 0.487. The simulation fared like the training, showing the network could predict unseen data to the best of its ability.

Figure 5.2-17 includes the correlation coefficient and mean square error value for the 69-neuron network. The graph shows that the 69-neuron network performed the best. In the case of the correlation coefficient the network performed almost as predicted by the trend line while the mean-squared error turned out to be even lower than predicted by the trend line.

Figure 5.2-19 shows a partial data set of the model performance of the 69-neuron network, with the full graph presented in Appendix F.1. The graph shows that the neural network follows the trend of the experimental data quite well, however, there is still an amount of noise around the constant parts of the plant-generated output porosity values. This shows that the neural network trains well in terms of a general trend, however, the accuracy of the network leaves room for improvement. The simulation data confirms this, with several data points simulating to values well above or below the production values. Although the neural network performance is deemed good, it should be better for accurate use as an inference measurement technique.

#### Bayesian regularisation training algorithm

The constant-time data was also subjected to a neural network trained with the Bayesian regularisation training algorithm. The tested neuron configurations yielded performances as listed in Table 5.2-9.

**Table 5.2-9: Performance of Bayesian regularisation algorithm on constant-time data**

|                    | <b>MSE</b> | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|------------|--------------------|----------------|---------------|
| <i>5 neurons</i>   | 0.89       | 0.63               | 0.49           | 0.61          |
| <i>10 neurons</i>  | 0.64       | 0.75               | 0.73           | 0.75          |
| <i>15 neurons</i>  | 0.49       | 0.82               | 0.78           | 0.81          |
| <i>25 neurons</i>  | 0.35       | 0.87               | 0.81           | 0.86          |
| <i>50 neurons</i>  | 0.20       | 0.93               | 0.83           | 0.91          |
| <i>75 neurons</i>  | 0.14       | 0.95               | 0.75           | 0.92          |
| <i>100 neurons</i> | 0.19       | 0.93               | 0.72           | 0.90          |

Similar to earlier observations related to the filtered and constant-time data sets, the performance of the artificial neural networks trained with the Bayesian regularisation training algorithm improves as the number of neurons in the hidden layer increases. This trend continues for each neuron configuration up to 75 neurons, where after the performance deteriorates. The best configuration is found between 50 and 75 neurons in the hidden layer.

The best configuration is the 75-neuron neural network. The configuration resulted in a mean-squared error of 0.141 and a correlation coefficient (R) of 0.921. The regression plots obtained for the training of the neural network is given in Appendix C.3, Figure C.3-12.

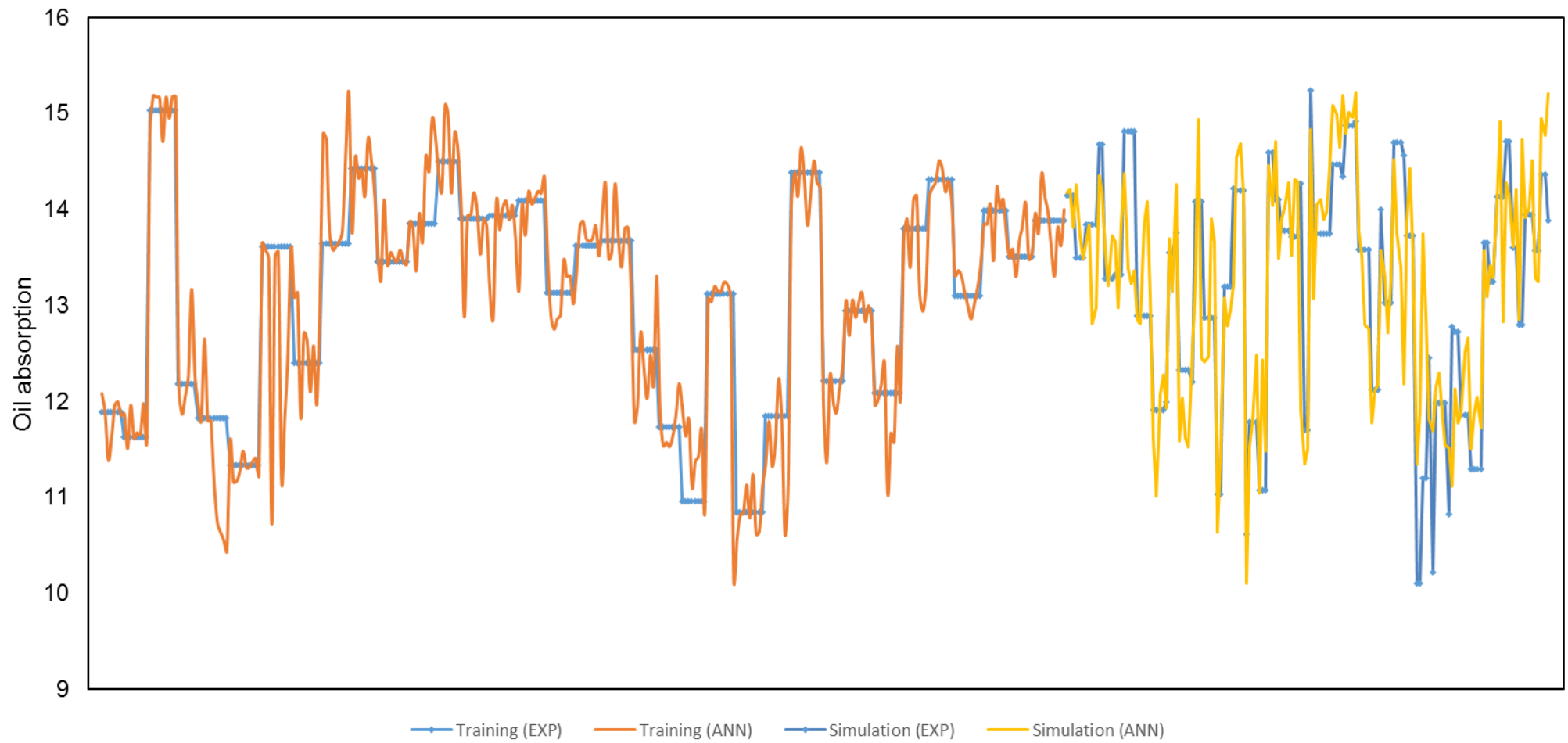


Figure 5.2-19: Model performance of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using constant-time data

The coefficients of determination ( $R^2$ ) for training and simulation is given as 0.848 and 0.650 respectively. This gives a correlation coefficient of 0.806 for the simulation of the abovementioned neural network. The mean-squared error for the simulation of the neural network is 0.512. The simulation fared good but did not correlate as good as the training of the neural network. This shows that the neural network training is sufficient however the testing during training does not fare good enough to simulate well. This is confirmed by Figure C.3-12 in Appendix C.3.

Figure 5.2-20 shows the mean-squared error and correlation coefficients of each of the neuron configurations that was tested, as well as the trend each variable forms.

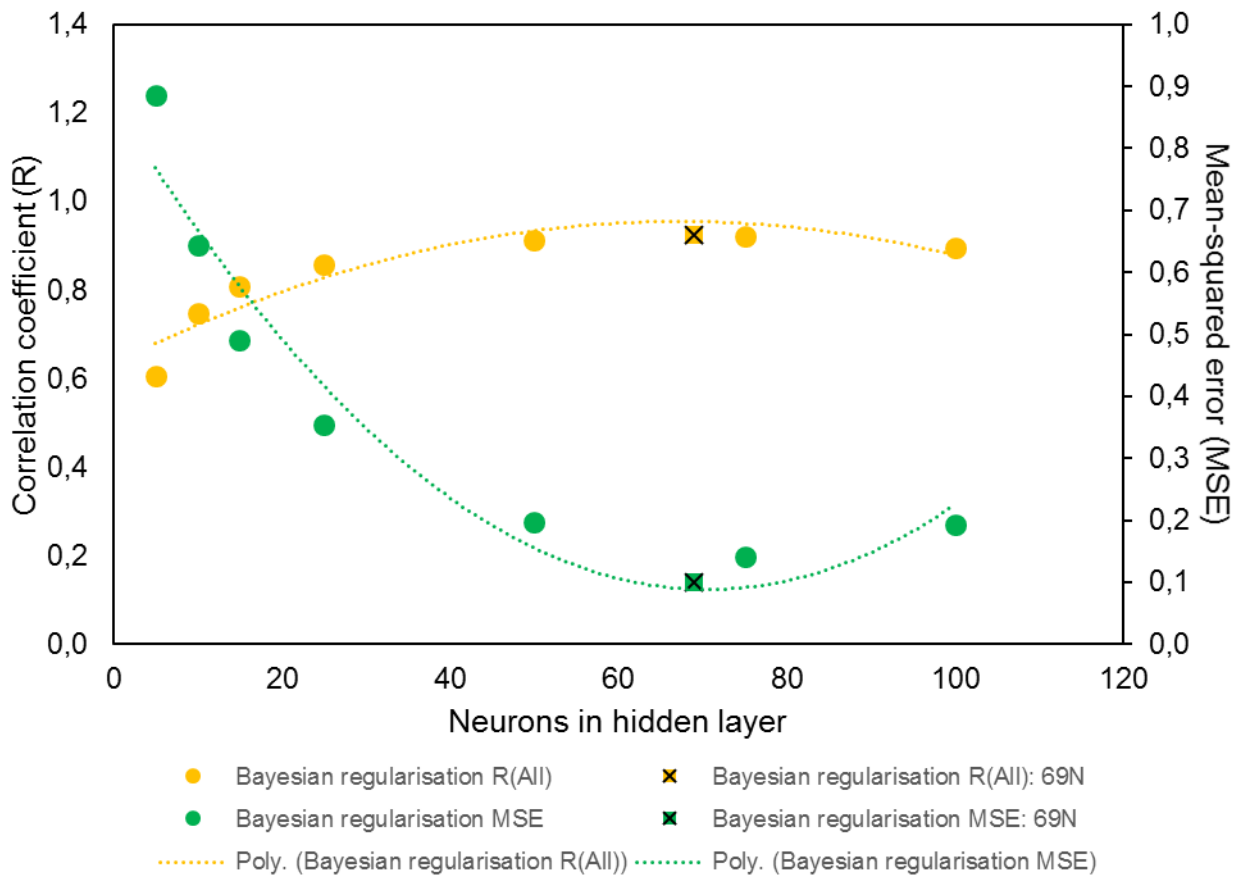


**Figure 5.2-20: MSE-values and R-values for neuron-configurations using the Bayesian regularisation training and constant-time data**

The trend lines for both the correlation coefficient and the mean-squared error is quadratic with minimum and maximum values between 50 and 75 neurons. The equation for the trend line of the correlation coefficient is given as Equation 5.2-7:

$$y = (-7.14 \times 10^{-5})x^2 + (9.60 \times 10^{-3})x + 6.32 \times 10^{-1} \quad 5.2-7$$

The maximum for the trend line is found to be 67 neurons. Equation 5.2-8 gives the trend line equation for the mean-squared error:

$$y = (1.58 \times 10^{-4})x^2 + (-2.23 \times 10^{-2})x + 8.76 \times 10^{-1} \quad 5.2-8$$

The minimum value for the trend line is at 71 neurons. This results in an average of 69 neurons which was tested under the same conditions as all the original neuron configurations. The regression for both training and simulation for the 69-neuron network is given in Figure 5.2-21.

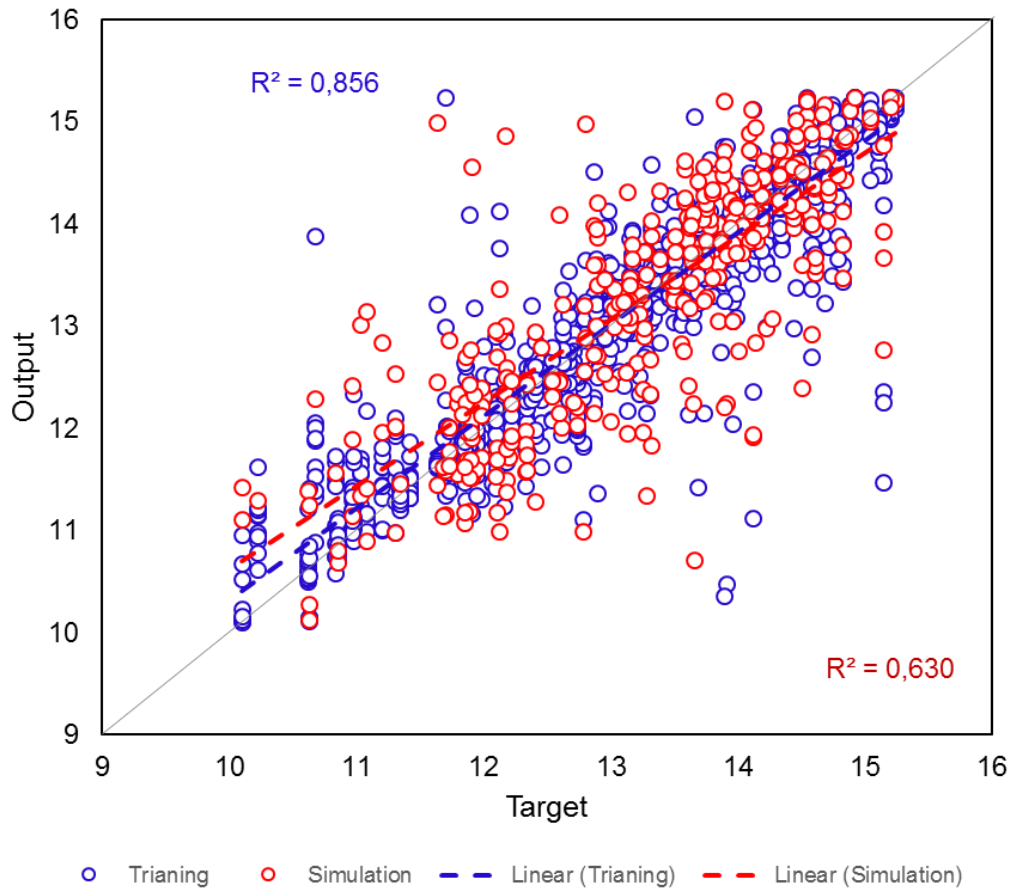


**Figure 5.2-21: Regression plots of the training and simulation of a 69-neuron neural network trained with the Bayesian regularisation algorithm using constant-time data**

The mean-squared error for the training of the abovementioned neural network is given as 0.099 while the correlation coefficient for training is 0.925. The regression plots obtained for the neural network is given in Appendix C.3 as Figure C.3-14.

From the figure above, the coefficient of determination for training and simulation of the neural network is found to be 0.856 and 0.630 respectively. The simulation correlation coefficient is determined at 0.793, which is a lot lower than that of the training. The mean-squared error for the

simulation is 0.561. The performance shows that the neural network trains well but does not predict accurately using unseen data.

Figure 5.2-20 contains the performance values for the 69-neuron network. The figure shows the correlation coefficient did not fare as predicted but resulted in a slightly lower value than what the trend line predicted. The correlation coefficient still performed better than that of the 75-neuron network, showing that the network did train better. The mean-squared error fared as predicted by the trend line. Thus, the 69-neuron network still performed the best with the Bayesian regularisation training algorithm in the case of constant-time data.

Figure 5.2-22 shows the model performance of the abovementioned neural network. The figure only shows a part of the data, while the graph of the full data set is given in Appendix F.1. As with the best performance using the Levenberg-Marquardt training algorithm, the neural network follows the trend set by the production data, however, the neural network output still does not remain close to the constant plant porosity values during the periods wherein these are kept constant. It is noticeable, however, that the number of outliers in the figure below is less than that for the Levenberg-Marquardt neural network, however, the few outliers shown in this graph for both training and simulation of the neural network are very large outliers.

#### Training algorithm comparison for the constant-time data

The best performing neural networks using both Levenberg-Marquardt training algorithm and Bayesian regularisation training algorithm, are given in Table 5.2-10.

**Table 5.2-10: Comparison of training algorithms using constant-time data**

|                                                   | <b>MSE</b> | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|---------------------------------------------------|------------|--------------------|----------------------|----------------|---------------|
| <b>Levenberg-Marquardt training algorithm</b>     |            |                    |                      |                |               |
| <i>50 neurons</i>                                 | 0.71       | 0.82               | 0.71                 | 0.77           | 0.77          |
| <i>69 neurons</i>                                 | 0.59       | 0.91               | 0.81                 | 0.69           | 0.86          |
| <i>75 neurons</i>                                 | 0.66       | 0.89               | 0.74                 | 0.76           | 0.85          |
| <b>Bayesian regularisation training algorithm</b> |            |                    |                      |                |               |
| <i>50 neurons</i>                                 | 0.20       | 0.93               |                      | 0.83           | 0.91          |
| <i>69 neurons</i>                                 | 0.10       | 0.97               |                      | 0.73           | 0.93          |
| <i>75 neurons</i>                                 | 0.14       | 0.95               |                      | 0.75           | 0.92          |

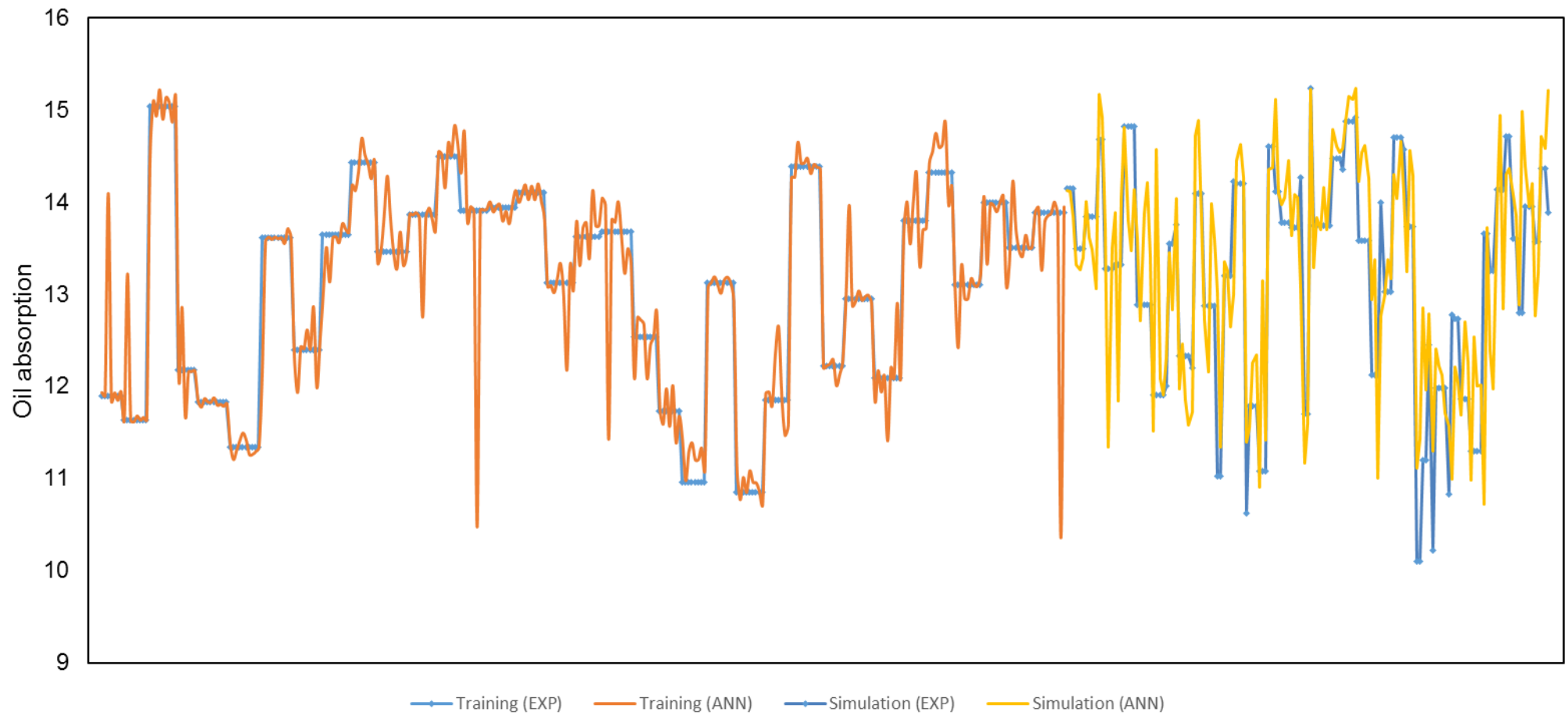


Figure 5.2-22: Model performance of a 69-neuron neural network trained with the Bayesian regularisation algorithm using constant-time data

The Bayesian regularisation training algorithm outperformed the Levenberg-Marquardt training algorithm by far in every instance investigated. The dataset is quite large, thus favouring the Bayesian regularisation training algorithm.

The best trained neural network when using constant-time data is obtained with a 69-neuron hidden layer and the Bayesian regularisation training algorithm. The performance of the neural network is acceptable, however, the lack of performance in the simulation must also be considered in choosing an adequate network. The performance of the network could possibly be improved with more consistent data.

#### 5.2.1.5 Assessing the influence of the time-interval on the constant-time data set

The initial constant-time interval of 10 minutes was chosen arbitrarily. Different time intervals were tested to find an optimum sampling interval. The additional time intervals that were assessed include 5, 15, 20, 30 and 40 minutes.

The intervals were tested with a 75-neuron hidden layer ANN trained with the Bayesian regularisation training algorithm. Table 5.2-11 shows the performance of the ANN trained with data using each time interval specified above.

**Table 5.2-11: Performance of time-interval testing for constant-time data**

|                   | MSE  | R(training) | R(test) | R(all) |
|-------------------|------|-------------|---------|--------|
| <i>5 minutes</i>  | 0.22 | 0.92        | 0.74    | 0.89   |
| <i>10 minutes</i> | 0.14 | 0.95        | 0.75    | 0.92   |
| <i>15 minutes</i> | 0.15 | 0.94        | 0.78    | 0.92   |
| <i>20 minutes</i> | 0.22 | 0.92        | 0.84    | 0.91   |
| <i>30 minutes</i> | 0.24 | 0.91        | 0.83    | 0.90   |
| <i>40 minutes</i> | 0.28 | 0.89        | 0.84    | 0.88   |

The mean-squared error as well as the correlation coefficients of each configuration was plotted on a graph, which is depicted in Figure 5.2-23.

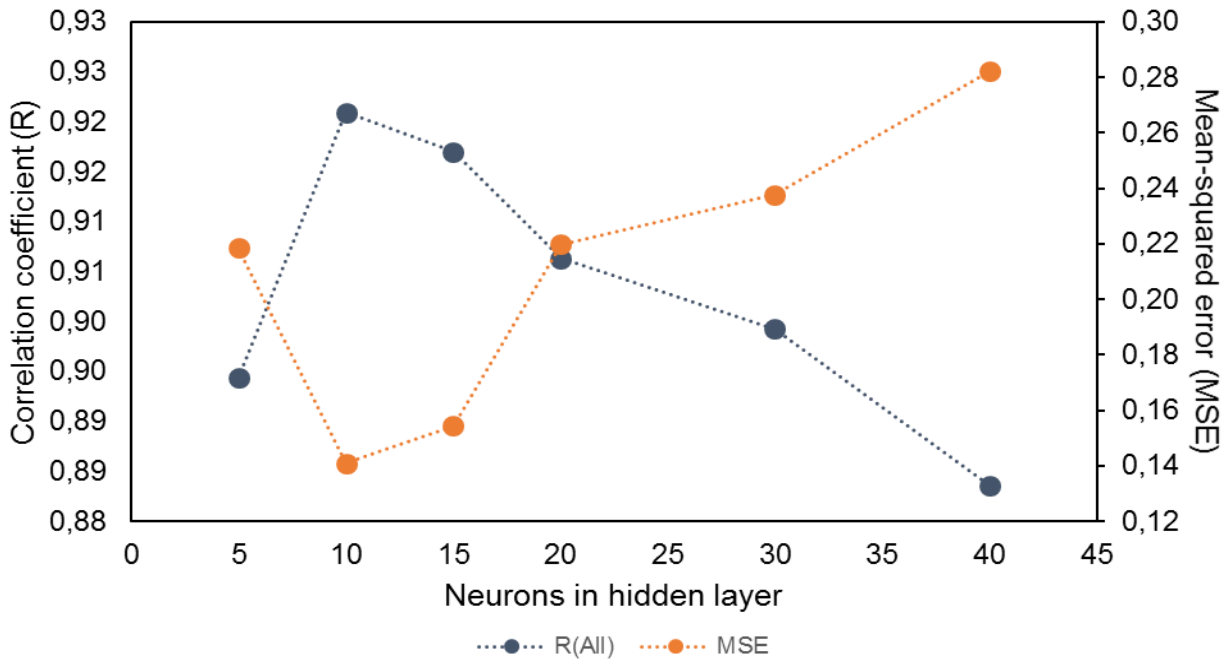


Figure 5.2-23: MSE and R-values for time-interval testing for constant-time data

The data shows that an ANN trained with data with a simulated 10-minute sampling interval performs better than a similar ANN trained with data with any other chosen simulated sampling interval. The performance decreases significantly when the time interval increases beyond 10 minutes, which can be due to the data becoming less significant the further it is removed in time from the actual plant measured oil absorption value.

The results show that data for training a neural network for this application should consist of 10 input steps for each new output, meaning data sampling for any new data should be done in 10-minute intervals.

#### 5.2.1.6 Assessing the influence of sensor filtering on constant-time data set

##### Levenberg-Marquardt training algorithm

The 10-minute constant-time data was also subjected to sensor filtering to assess the effect of this treatment on the performance of the artificial neural network. Both Levenberg-Marquardt and Bayesian regularisation training algorithms were tested with data filtered to remove noise. In each case the 50-neuron and 75-neuron networks were tested to see whether performance of the networks improve, after which the neurons in the hidden layer was increased until the performance worsens. The data was filtered using linear sensor filtering. The time constant for each variable is given in Table 5.2-12:

**Table 5.2-12: Time constants for linear sensor filtering for phase 1**

| Input parameter            | Time constant ( $\tau_f$ ) [min] |
|----------------------------|----------------------------------|
| Fluidising air flow rate   | 4.5                              |
| Fluidising air temperature | 3.5                              |
| Liquid AN density          | 4                                |
| Liquid AN temperature      | 2.5                              |
| Liquid AN flow rate        | 2                                |
| Atomising air flow rate    | 1.5                              |

The performance of the artificial neural networks using sensor filtered data and the Levenberg-Marquardt training algorithm is tabulated in Table 5.2-13. The performance of the corresponding neural networks using normal (unfiltered sensor input) constant-time data is also included in the table.

**Table 5.2-13: Comparison between normal and sensor filtered 10-minute constant-time data using the Levenberg-Marquardt training algorithm**

|                             | MSE  | R(training) | R(validation) | R(test) | R(all) |
|-----------------------------|------|-------------|---------------|---------|--------|
| <b>Normal data</b>          |      |             |               |         |        |
| 50 neurons                  | 0.71 | 0.82        | 0.71          | 0.77    | 0.80   |
| 75 neurons                  | 0.66 | 0.89        | 0.74          | 0.76    | 0.85   |
| 100 neurons                 | 0.74 | 0.85        | 0.73          | 0.76    | 0.82   |
| <b>Sensor filtered data</b> |      |             |               |         |        |
| 50 neurons                  | 0.28 | 0.88        | 0.88          | 0.91    | 0.89   |
| 75 neurons                  | 0.20 | 0.92        | 0.93          | 0.91    | 0.92   |
| 100 neurons                 | 0.16 | 0.99        | 0.95          | 0.93    | 0.97   |
| 125 neurons                 | 0.28 | 0.97        | 0.89          | 0.86    | 0.94   |

The table shows that the training with sensor filtered data performed significantly better than normal constant-time data. The performance increases with an increase in neurons up to 100 neurons in the hidden layer, however, the performance decreased when testing a 125-neuron network.

The regression plot of the training as well as the simulation of the abovementioned network is given in Figure 5.2-24. The training coefficient of determination is given as 0.949, while for simulation it is 0.876. This results in a correlation coefficient of 0.936. The mean-squared error

for the simulation of the 100-neuron network is 0.193. The neural network trained and simulated very well, resulting in a very good predictive model.

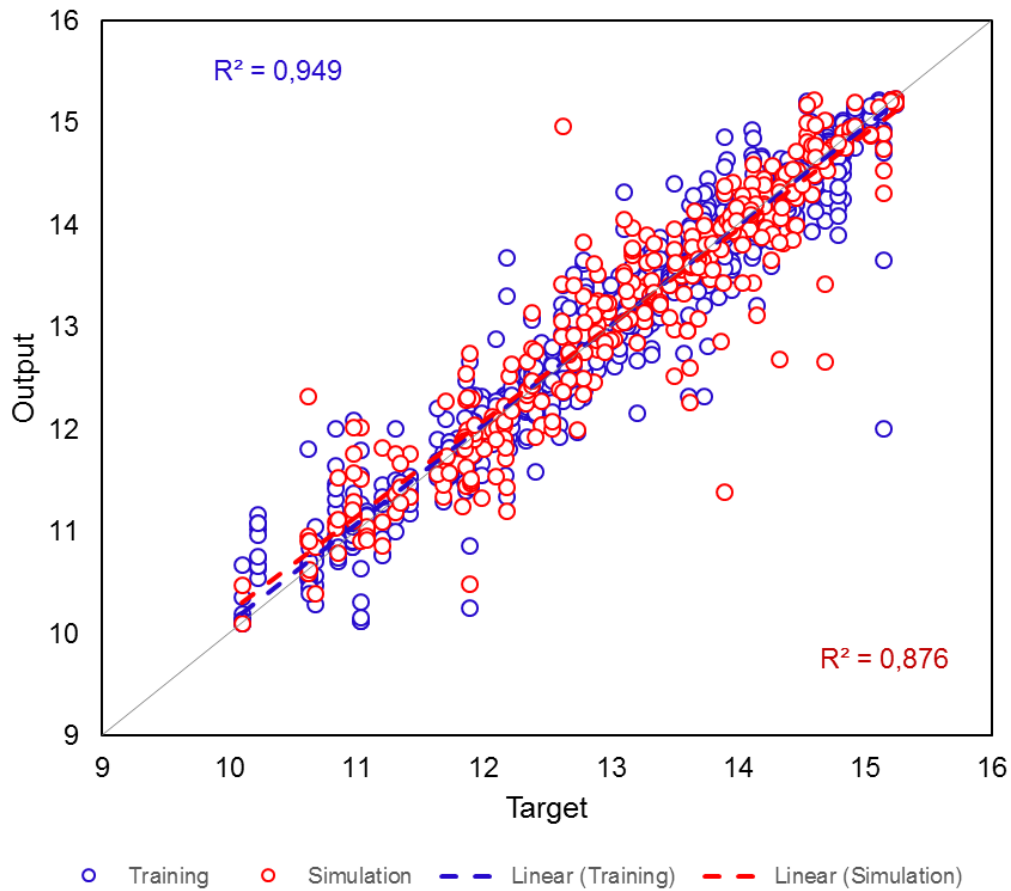


**Figure 5.2-24: Regression plots of the training and simulation of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using sensor filtered constant-time data**

The model performance of the 100-neuron network trained using sensor filtered constant-time data is given in Appendix F.1. Figure 5.2-25 shows a graph containing a portion of the data. The figure shows that the neural network follows the trend of the plant data very well. The oscillations around the constant operational values are reduced significantly. Less significant outliers are still present in the simulation data, which shows that, although the neural network does not predict unseen data as well as it should, a visible improvement is achieved relative to the data with unfiltered sensor inputs.

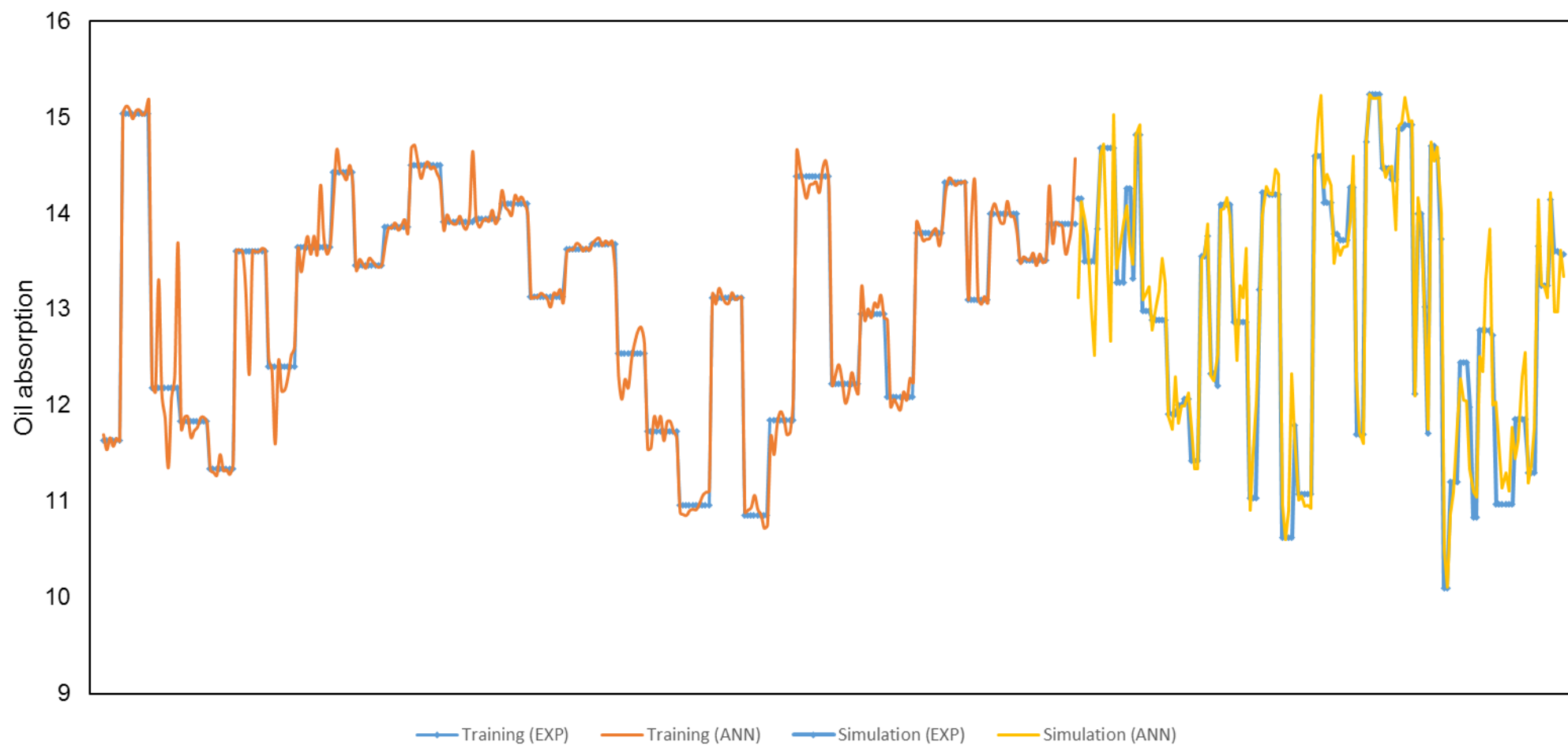


Figure 5.2-25: Model performance of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using sensor filtered constant-time data

### Bayesian regularisation training algorithm

The performance obtained with the Bayesian regularisation training using sensor filtered constant-time data, together with unfiltered sensor data neural networks' performances for comparison, is given in Table 5.2-14.

**Table 5.2-14: Comparison between normal and sensor filtered 10-minute constant-time data using Bayesian regularisation training algorithm**

|                             | <b>MSE</b>           | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|-----------------------------|----------------------|--------------------|----------------|---------------|
| <b>Normal data</b>          |                      |                    |                |               |
| <i>50 neurons</i>           | 0.20                 | 0.93               | 0.83           | 0.91          |
| <i>75 neurons</i>           | 0.14                 | 0.95               | 0.75           | 0.92          |
| <i>100 neurons</i>          | 0.19                 | 0.93               | 0.72           | 0.90          |
| <b>Sensor filtered data</b> |                      |                    |                |               |
| <i>50 neurons</i>           | $2.5 \times 10^{-2}$ | 0.99               | 0.92           | 0.98          |
| <i>75 neurons</i>           | $8.0 \times 10^{-3}$ | 0.99               | 0.92           | 0.98          |
| <i>100 neurons</i>          | $2.5 \times 10^{-3}$ | 0.99               | 0.93           | 0.99          |
| <i>125 neurons</i>          | $4.2 \times 10^{-2}$ | 0.99               | 0.92           | 0.98          |

As shown in the table above, the sensor filtered data trained significantly better than the normal (unfiltered sensor input) data. An increase in neurons in the hidden layers brought about an improvement in the performance of the neural networks, with the performance of the 125-neuron network starting to deteriorate from that of the 100-neuron network. As with the Levenberg-Marquardt algorithm, the neural network with 100 neurons in the hidden layer showed the better performance in this assessment.

The regression plot for the 100-neuron network training and simulation is depicted in Figure 5.2-26 below. The coefficients of determination for training and simulation is obtained at 0.976 and 0.875 respectively. This results in a simulation correlation coefficient of 0.935 which is close to 0.976. The mean-squared error of the simulation of the above-mentioned neural network is 0.206. The simulation shows that the abovementioned network, even with excellent training, was not very good at predicting the particle porosity using unseen data.

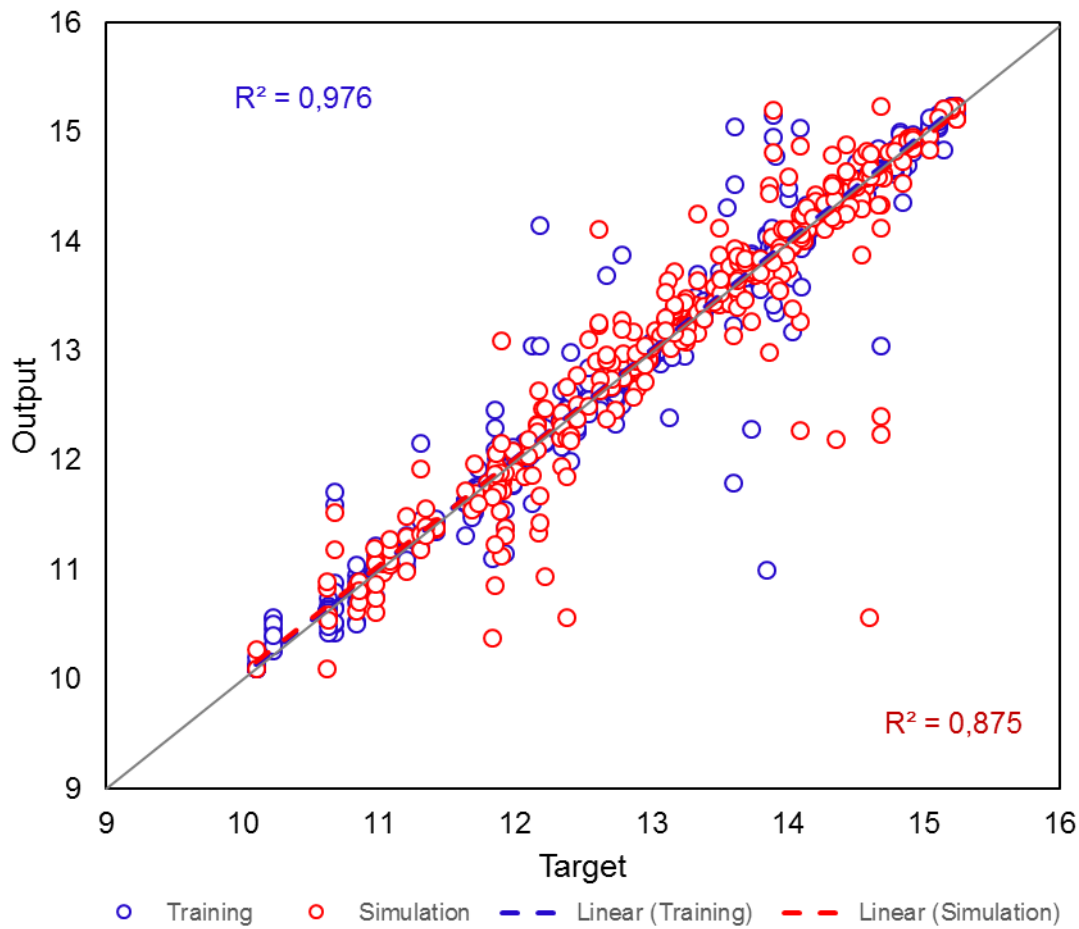


**Figure 5.2-26: Regression plots of the training and simulation of a 100-neuron neural network trained with the Bayesian regularisation algorithm using sensor filtered constant-time data**

The full graph showing the model's performance is given in Appendix F.1 while a part of the graph is presented below in Figure 5.2-27. The model performance graph shows that the neural network trained very well with almost no oscillation around the constant experimental values. The noise filtering of the input values greatly benefits the training of the neural network as well as the simulation of unseen data using the trained neural network. The simulation shows a few visually rather insignificant outliers compared to the plant's data, however, the general trend shows a visual good performance using unseen data, as is visible in the yellow data set at the right end of the graph.

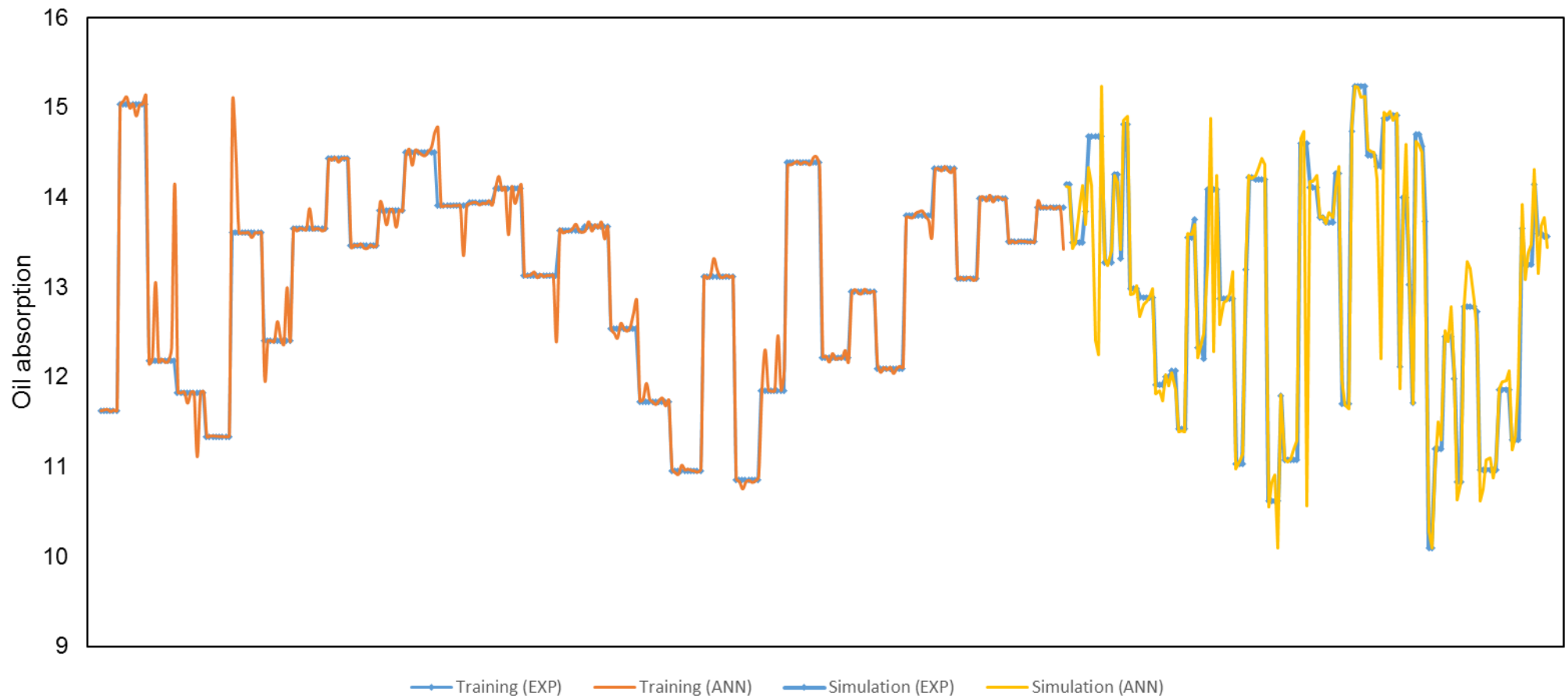


Figure 5.2-27: Model performance of a 100-neuron neural network trained with the Bayesian regularisation algorithm using sensor filtered constant-time data

### 5.2.1.7 Dataset comparison

The datasets above, including the dataset that was filtered for sensor noise, was compared by taking the best performing neural network in each case and comparing mean-squared error and correlation coefficient values. Table 5.2-15 shows the performance of each data set:

**Table 5.2-15: Comparison of neural network performance with different datasets**

|                                 | Info            | MSE  | R(train) | R(val) | R(test) | R(all) | Rating |
|---------------------------------|-----------------|------|----------|--------|---------|--------|--------|
| <i>Filtered data</i>            | <i>BR; 125N</i> | 0.47 | 0.81     |        | 0.78    | 0.81   | 3      |
| <i>Discrete data</i>            | <i>LM; 100N</i> | 0.56 | 0.53     | 0.80   | 0.51    | 0.56   | 4      |
| <i>Constant-time (CT) data</i>  | <i>BR; 69N</i>  | 0.10 | 0.97     |        | 0.73    | 0.93   | 2      |
| <i>CT data, sensor filtered</i> | <i>BR; 100N</i> | 0.00 | 0.99     |        | 0.93    | 0.99   | 1      |

The table shows that the constant-time data set clearly outperformed the other datasets with a further increase in the performance using sensor filtering. This shows the need for continuous sampling to obtain output values as well as sensor filtering when generating an artificial neural network for inference control. The artificial neural network training performs best with data with constant time between output sampling.

The purpose of phase 1 analyses was to obtain variables in which phase 2 data must be obtained and analysed. The lessons learned from phase 1 data analyses, are as follows:

- The constant-time data set performed the best, showing that samples should be taken in constant intervals of 10 minutes.
- The input data should be filtered to reduce excessive noise in the data.
- The neural network will perform better with a high number of neurons in the hidden layer.
- The constant-time data set performed best when trained with the Bayesian regularisation algorithm, thus the neural network should be trained using that training algorithm.

Phase 2 will consist of new data acquired with a 1-minute sensor sampling rate and a 10-minute interval product sampling rate with no interruptions. In phase 1 the neural network training algorithm with the best performance is the Bayesian regularisation using 100 neurons in the hidden layer. This will be the starting point for neural network testing in phase 2.

## 5.2.2 Phase 2 predictive model

### 5.2.2.1 Sample analysis

This phase of data accumulation entailed plant trials during which time samples were taken and analysed to obtain particle porosity and sphericity data using the optical porosity measurement method. The method correlates well with the oil absorption method, as shown in Section 5.1.1.3. During the trial work a different, denser granular ammonium nitrate product as for phase 1 was produced in the plant, and therefore twenty of the samples were subjected to both the optical porosity measurement method and the oil absorption method for comparison reasons. Figure 5.2-28 shows the volume-based porosity values of these twenty samples as measured by both the optical porosity measurement and the oil absorption techniques. The results are also given in Table D.1-1 in Appendix D.1.1.

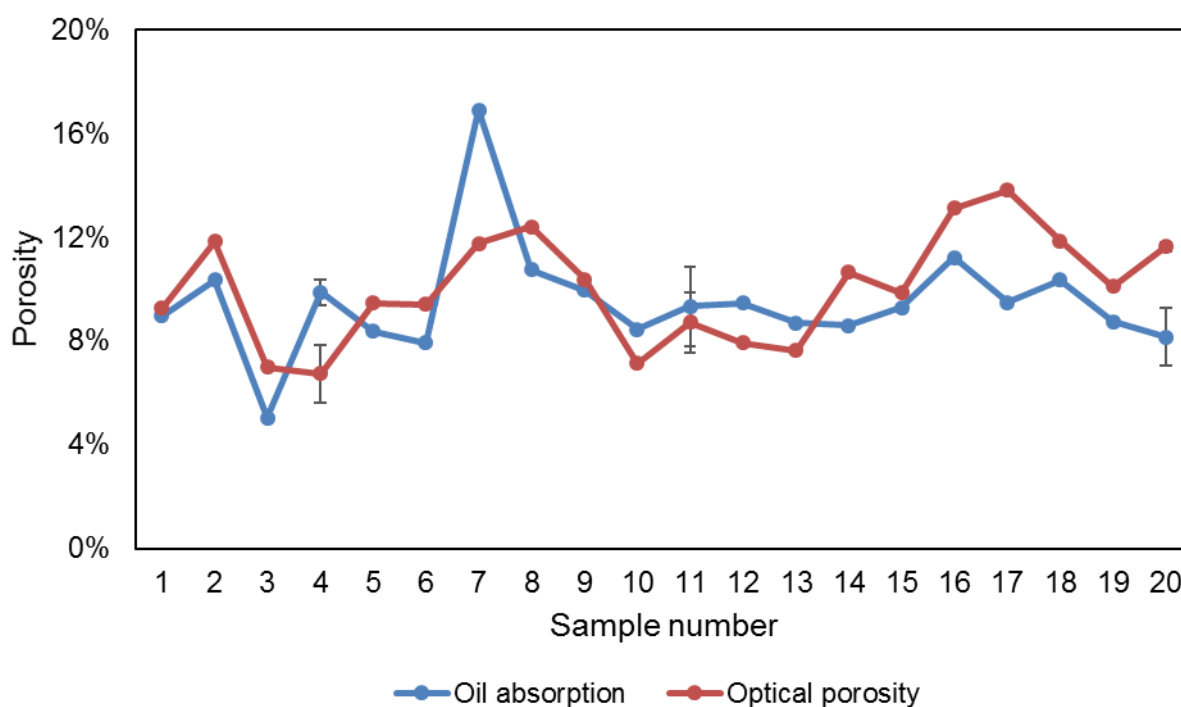


**Figure 5.2-28: Optical porosity measurement and oil absorption values for phase 2**

The errors for three of the samples, namely samples 4, 11 and 20, that were repeated are shown in the figure above. The repeatability was calculated on samples of which the analyses techniques were repeated three times each on both the abovementioned methods. The values are given in Table D.1-2 to Table D.1-7 in Appendix D.1. A regression graph of the two methods were generated and is shown in Figure 5.2-29.

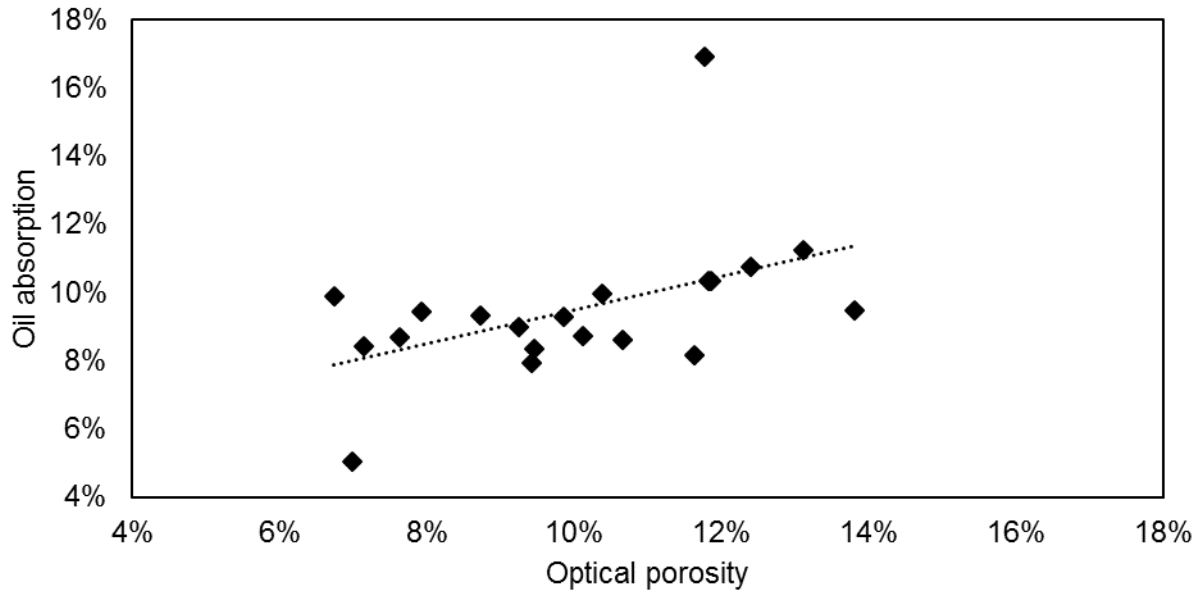


**Figure 5.2-29: Linear regression between oil absorption and optical porosity measurement for phase 2**

The coefficient of determination between the optical porosity measurement method and the oil absorption method when making high density ammonium nitrate, is 0.226. The equation of the trend-line is given in Equation 5.2-9.

$$y = 0.49x + 0.05 \quad 5.2-9$$

This equation was used to “convert” each optical porosity measurement value into an equivalent oil absorption technique value for use as output variable in the neural networks.

### 5.2.2.2 Data processing

#### Time-advance determination

The samples were taken at a point quite far downstream from the granulator and thus, a time advance analysis of the output data relative to the input data was conducted. The variable showing the most reliable results for a time-advance is the liquid ammonium nitrate temperature. Figure 5.2-30 shows the correlation coefficient for each time shift. The correlation coefficient given in the figure show the correlation calculated between liquid ammonium nitrate temperature (input parameter) and the oil absorption value (output variable).

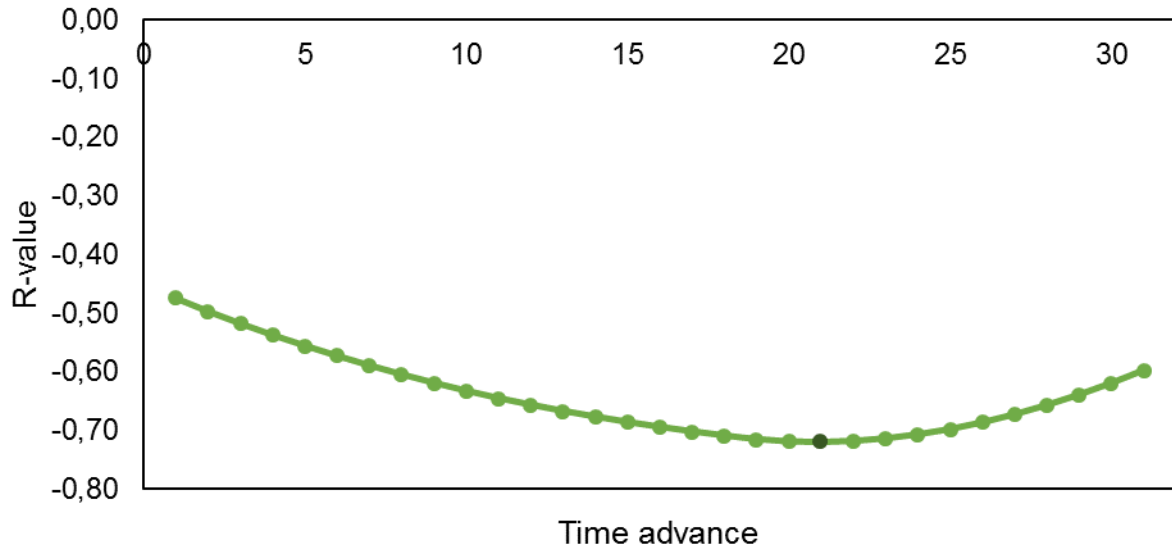


**Figure 5.2-30: Time-advance correlation coefficients for liquid AN temperature**

The time-advance is shown in the graph where a minimum correlation coefficient is found at 21 minutes, meaning that the output variable must be shifted “back in time” for 21 minutes to correspond with the conditions in the granulator at the moment of manufacturing. The idea behind this inference measurement is to estimate the selectivity variables at the moment the product is manufactured to allow for in-time control actions to immediately correct any deviations.

#### Linear statistical sensor filtering

The phase 2 data did not need linear statistical filtering for all the input parameters. The three variables that needed filtering as well as their time constants are listed in Table 5.2-16.

**Table 5.2-16: Time constants for linear sensor filtering for phase 2**

| Input parameter        | Time constant ( $\tau_f$ ) [min] |
|------------------------|----------------------------------|
| Bed height             | 1,5                              |
| Liquid AN pressure     | 2,5                              |
| Atomising air pressure | 2,5                              |

#### **5.2.2.3 Artificial neural network training and testing on porosity estimation data**

The scope for the neural network testing for porosity estimation was based on phase 1 results. Phase 1 predicted the best performing network to have a high number of neurons in a single hidden layer. The configuration testing started with 100 neurons in the hidden layer then reducing

the neurons in the hidden layer to 10 to be able to obtain a quadratic regression fit for the neural network performance evaluations.

The dataset consisted of 2 067 data points in total. The data was divided into a training dataset of 1 654 data points and a simulation dataset of 413 data points. The full dataset is given in Appendix F.2.

Figure 5.2-31 shows a partial set of data for each input parameter together with the porosity values as output variable. The full figure is given in Appendix E.1. The bed height, liquid ammonium nitrate pressure and the atomising air pressure had to be filtered to reduce noise in the data, as can be seen by the darker coloured lines in each of the abovementioned graphs.

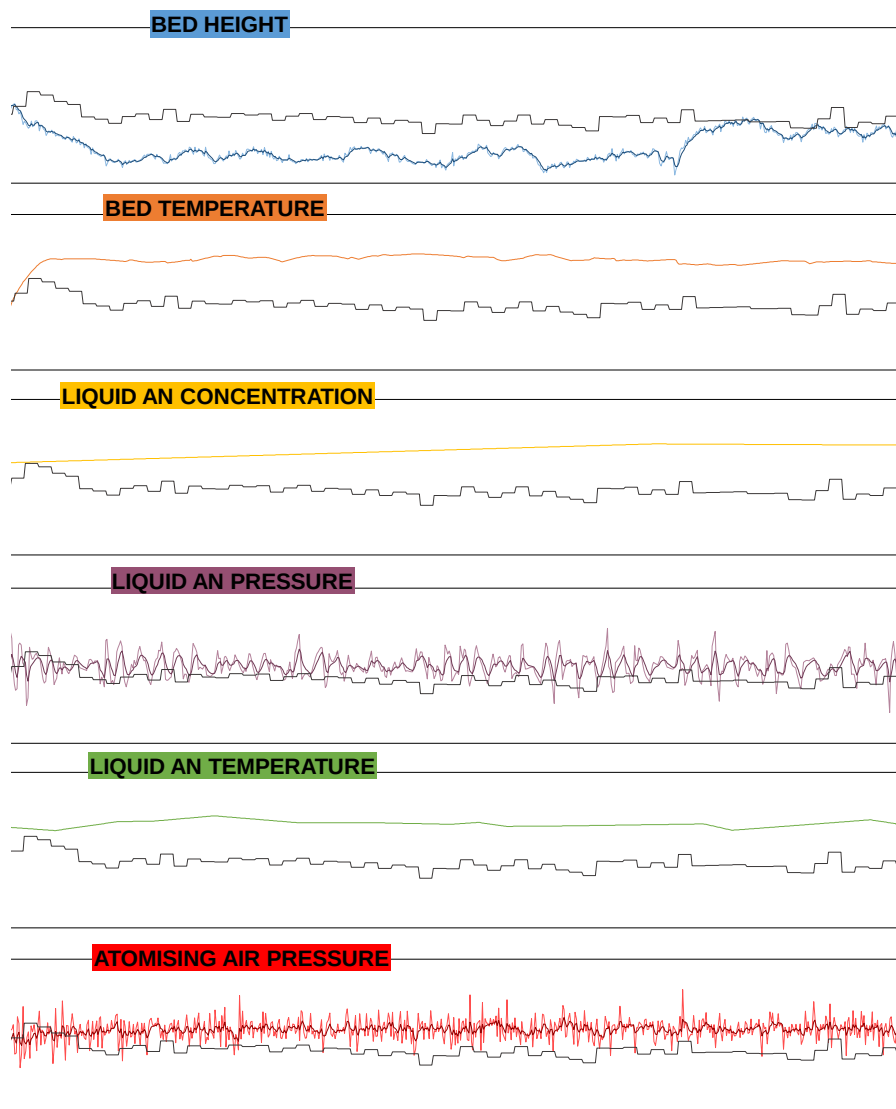


Figure 5.2-31: Partial phase 2 sensor input and porosity data

Table 5.2-17 shows the results for each neural network's neuron configuration tested for porosity estimation.

**Table 5.2-17: Performance of phase 2 porosity estimation using the Bayesian regularisation training algorithm**

|                    | <b>MSE</b>           | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|----------------------|--------------------|----------------|---------------|
| <i>10 neurons</i>  | $1.6 \times 10^{-5}$ | 0.91               | 0.88           | 0.90          |
| <i>50 neurons</i>  | $5.5 \times 10^{-6}$ | 0.97               | 0.92           | 0.96          |
| <i>100 neurons</i> | $1.1 \times 10^{-5}$ | 0.93               | 0.93           | 0.93          |

The table shows that the best performing neural network configuration was obtained when using 50 neurons in the hidden layer. The mean-squared error for the training of the 50-neuron network is calculated as  $5.5 \times 10^{-6}$ , while the correlation coefficient (R) for training is 0.959. The generated regression graph is given in Appendix E.2.1 as Figure E.1-.

The linear regression plot shows that the coefficient of determination ( $R^2$ ) for training is 0.920 and for simulation is 0.853. This shows very good correlation between artificial neural network output values and the target values. The correlation coefficient for simulation was calculated as 0.923, which is close to the training correlation coefficient of 0.959. The mean-squared error for the simulation is  $1.6 \times 10^{-5}$ . This shows that the network trained well, but also had the ability to predict unseen data with a good correlation but with a larger error as with training. The lower mean-squared error for simulation is due to the simulation data set having more outliers than the training data set.

The mean-squared error as well as the correlation coefficients of each neural network configuration was plotted on a graph to obtain quadratic equations for each of the performance evaluations. Figure 5.2-32 shows the graph.

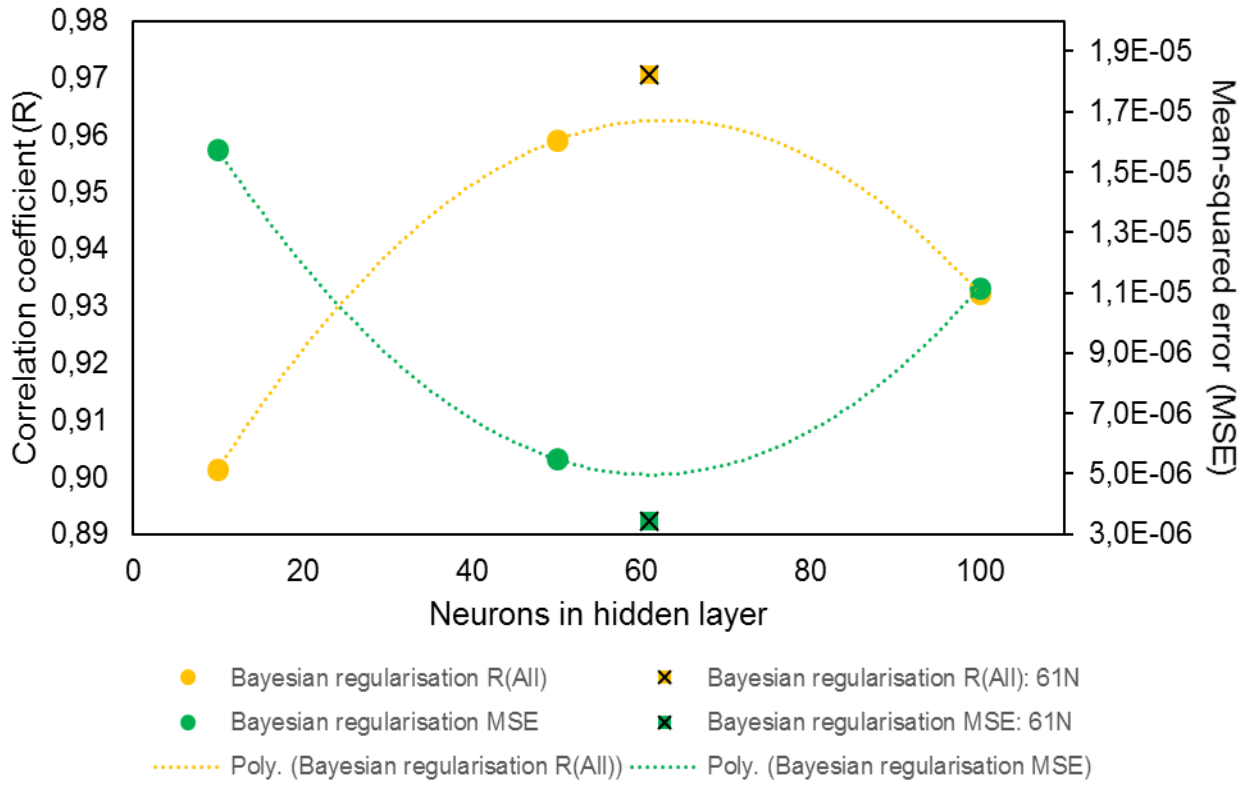


**Figure 5.2-32 MSE-values and R-values for neuron-configurations using the Bayesian regularisation training algorithm and phase 2 porosity data**

The quadratic regression line equation for the correlation coefficient should yield a maximum between 50 and 100 neurons in the hidden layer. The regression equation is given in Equation 5.2-10.

$$y = (-2.20 \times 10^{-5})x^2 + (2.76 \times 10^{-3})x + 8.76 \times 10^{-1} \quad 5.2-10$$

The maximum correlation coefficient was predicted to be obtained with 63 neurons in the hidden layer.

The mean-squared error quadratic equation is given as Equation 5.2-11 and should have a minimum between 50 and 100 neurons in the hidden layer.

$$y = (4.11 \times 10^{-9})x^2 + (-5.03 \times 10^{-7})x + 2.04 \times 10^{-5} \quad 5.2-11$$

The minimum of this equation was obtained at 60 neurons in the hidden layer.

The average between the two predicted configurations resulted in a neural network with 61 neurons in the hidden layer. Figure 5.2-33 shows the linear regression plot for the training and simulation of the 61-neuron network. The generated regression plot is Figure E.2-3 in Appendix E.2.1.

As seen in the figure below, the coefficients of determination for the training and simulation of the 61-neuron network is 0.942 and 0.841 respectively. This results in correlation coefficients of 0.971 and 0.902 for training and simulation respectively. The mean-squared error for the training of the neural network is  $3.4 \times 10^{-6}$ , while the mean-squared error for the simulation of the neural network is  $1.9 \times 10^{-5}$ . The performance of the training of the neural network is very good, however, the simulation is not performing as good as the training showing that the neural network does not estimate the porosity from unseen data as well as it does with the data it was trained with. The performance of the simulation, however not as good as the training, is still exceptional and the neural network is still good enough to be used as a predictive model.



**Figure 5.2-33: Regression plots of the training and simulation of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data**

Figure 5.2-32 shows that the 61-neuron network performed better than expected. The best performing neural network with phase 2 data is therefore a 61-neuron network using the Bayesian regularisation training algorithm.

Figure 5.2-34 shows the model performance of the 61-neuron network. The figure shows that the neural network follows the tendency of the experimental data very well, even though some spikes are present and visible in the neural network's output values. The model performance shows that the simulation of the neural network performed very well and thus, the network will be able to predict unseen data very well.

#### Activation function comparison

The activation function used in all previous neural network configurations was the hyperbolic tangent function. The function was used due to its ability to handle data with unpredictable behaviour. The sigmoidal as well as pure linear activation functions were tested on the 61-neuron network using the Bayesian regularisation training algorithm. The results were compared to the hyperbolic tangent function. Table 5.2-18 shows the performance evaluation of each of the abovementioned tests.

**Table 5.2-18: Activation function comparison using phase 2 porosity data**

|                           | <b>MSE</b>           | <b>R(training)</b> | <b>R(test)</b> | <b>R(all)</b> |
|---------------------------|----------------------|--------------------|----------------|---------------|
| <i>Hyperbolic tangent</i> | $3.4 \times 10^{-6}$ | 0.98               | 0.92           | 0.97          |
| <i>Sigmoidal</i>          | $6.6 \times 10^{-6}$ | 0.96               | 0.91           | 0.95          |
| <i>Pure linear</i>        | $5.9 \times 10^{-5}$ | 0.56               | 0.53           | 0.56          |

The table shows that, as predicted, the hyperbolic tangent functioned had the best performance on the porosity estimation data. The regression plots for the sigmoidal and pure linear activation functions are given in Appendix E.2.2.



Figure 5.2-34: Model performance of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data

#### 5.2.2.4 Artificial neural network training and testing on sphericity estimation data

The sphericity estimation is based on the mean circularity value obtained from the Haver CPA 2-1 particle size analyser as output. The inputs as well as data processing was kept the same as with the porosity estimation data.

The sphericity estimation data consisted of 2 068 data points in total for each input parameter, which was divided into 1 654 training data points and 414 simulation data points. The full dataset is given in Appendix F.2. Figure 5.2-35 shows a portion of each input's data together with the corresponding sphericity values. The full figure is given in Appendix E.3. The input values for the sphericity estimation data are the same as for the porosity estimation data. It is noticed that the sphericity values, however, did vary more than the porosity values.

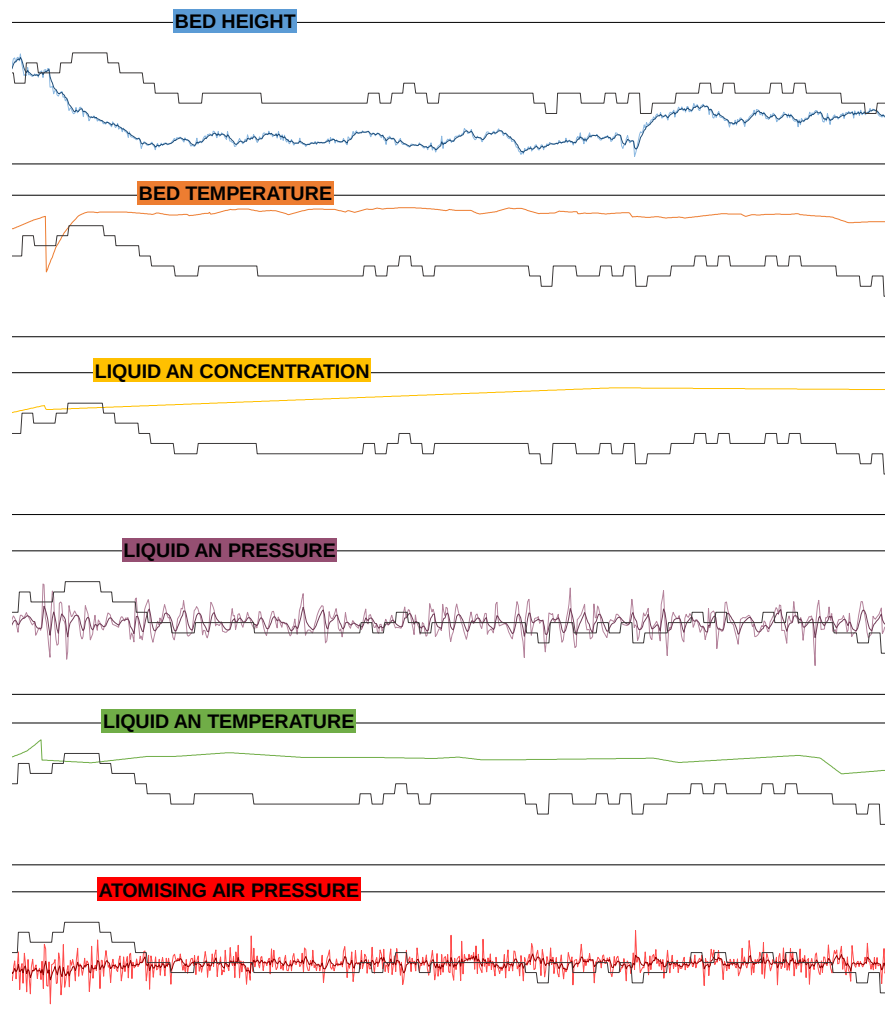


Figure 5.2-35: Partial phase 2 sphericity data

Levenberg-Marquardt training algorithm

The results for the artificial neural networks generated for sphericity estimation using the Levenberg-Marquardt training algorithm is tabulated in Table 5.2-19 below.

**Table 5.2-19: Performance of Levenberg-Marquardt algorithm on phase 2 sphericity data**

|                    | <b>MSE</b>           | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|--------------------|----------------------|--------------------|----------------------|----------------|---------------|
| <i>5 neurons</i>   | $2.9 \times 10^{-6}$ | 0.83               | 0.74                 | 0.75           | 0.80          |
| <i>10 neurons</i>  | $2.9 \times 10^{-6}$ | 0.79               | 0.74                 | 0.80           | 0.78          |
| <i>15 neurons</i>  | $1.8 \times 10^{-6}$ | 0.91               | 0.86                 | 0.88           | 0.90          |
| <i>25 neurons</i>  | $7.3 \times 10^{-7}$ | 0.97               | 0.94                 | 0.92           | 0.95          |
| <i>50 neurons</i>  | $7.7 \times 10^{-7}$ | 0.98               | 0.93                 | 0.93           | 0.97          |
| <i>75 neurons</i>  | $9.4 \times 10^{-7}$ | 0.98               | 0.92                 | 0.94           | 0.97          |
| <i>100 neurons</i> | $1.2 \times 10^{-6}$ | 0.98               | 0.90                 | 0.93           | 0.96          |

The mean-squared error improves up until 25 neurons in the hidden layer, after which it worsens for any additional neurons in the hidden layer. The correlation coefficient in turn increases with a maximum being reached at 50- and 75-neuron networks, which resulted in the same correlation coefficient. The correlation coefficient decreases for the 100-neuron network, showing a decrease in performance for this network.

The best performing neural network configuration when using the Levenberg-Marquardt training algorithm to predict sphericity, is the 50-neuron network. The mean-squared error and correlation coefficient for the abovementioned network was  $7.7 \times 10^{-7}$  and 0.967 respectively. The generated regression plots are given in Figure E.4- in Appendix E.4.1.

The coefficient of determination for the training of the neural network is obtained as 0.935. For the simulation of the network, the coefficient of determination is 0.912. The correlation coefficient for the simulation is calculated as 0.955. The mean-squared error for the simulation of the neural network was calculated to be  $5.32 \times 10^{-7}$ . These values are close to the correlation coefficient and mean-squared error of the training, showing that the neural network performed as good as it could with the training it underwent.

The mean-squared errors and correlation coefficients of each neuron configuration were plotted against the number of neurons in the hidden layers. Figure 5.2-36 shows the graph.

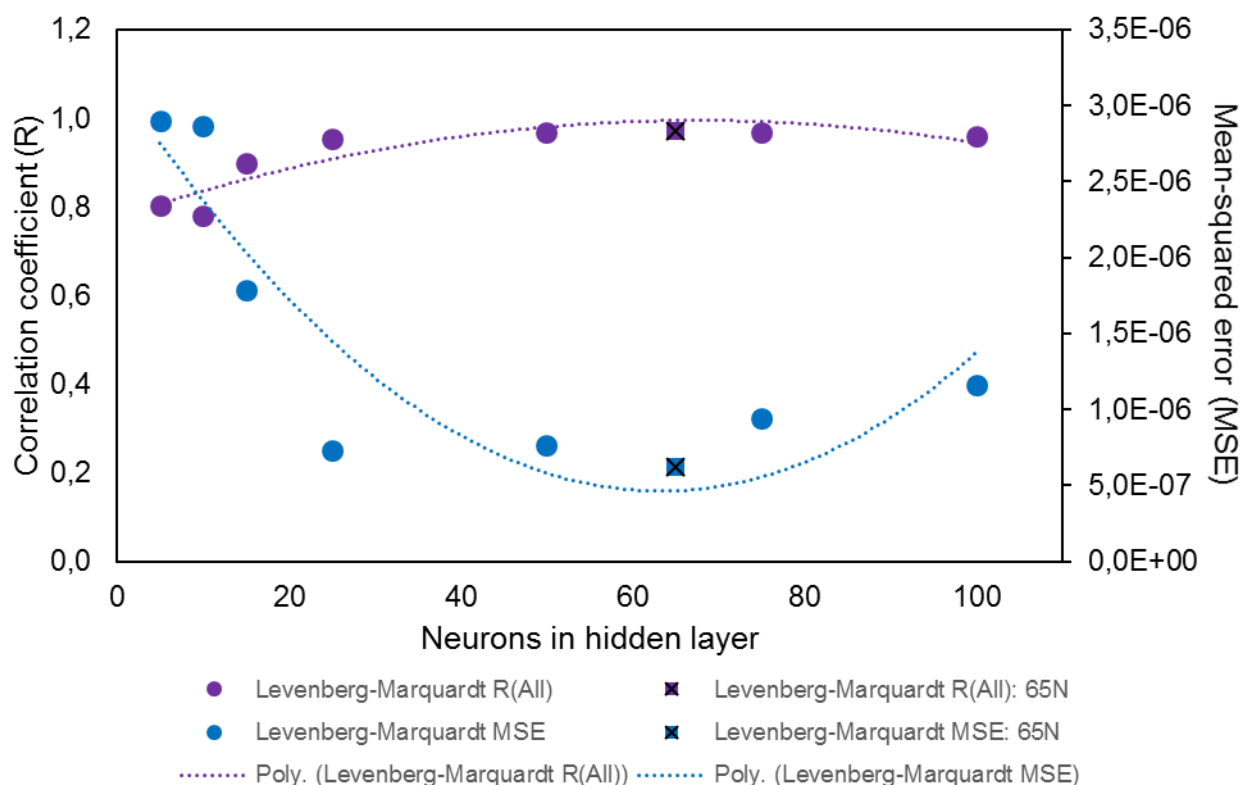


Figure 5.2-36: MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training algorithm and phase 2 sphericity data

The regression trend lines for the 5-neuron to 100-neuron networks were used to obtain the best neuron configuration. The trend line equation for the mean-squared error is given in Equation 5.2-12:

$$y = (6.75 \times 10^{-10})x^2 + (-8.54 \times 10^{-8})x + 3.17 \times 10^{-6} \quad 5.2-12$$

The minimum for the equation is calculated to be at 67 neurons. Equation 5.2-13 gives the trend line equation for the correlation coefficient.

$$y = (-4.82 \times 10^{-5})x^2 + (6.51 \times 10^{-3})x + 7.75 \times 10^{-1} \quad 5.2-13$$

The maximum of the equation yielded 63 neurons. This results in an average neuron configuration of 65. The configuration was tested, resulting in the regression plot given in Figure 5.2-37.

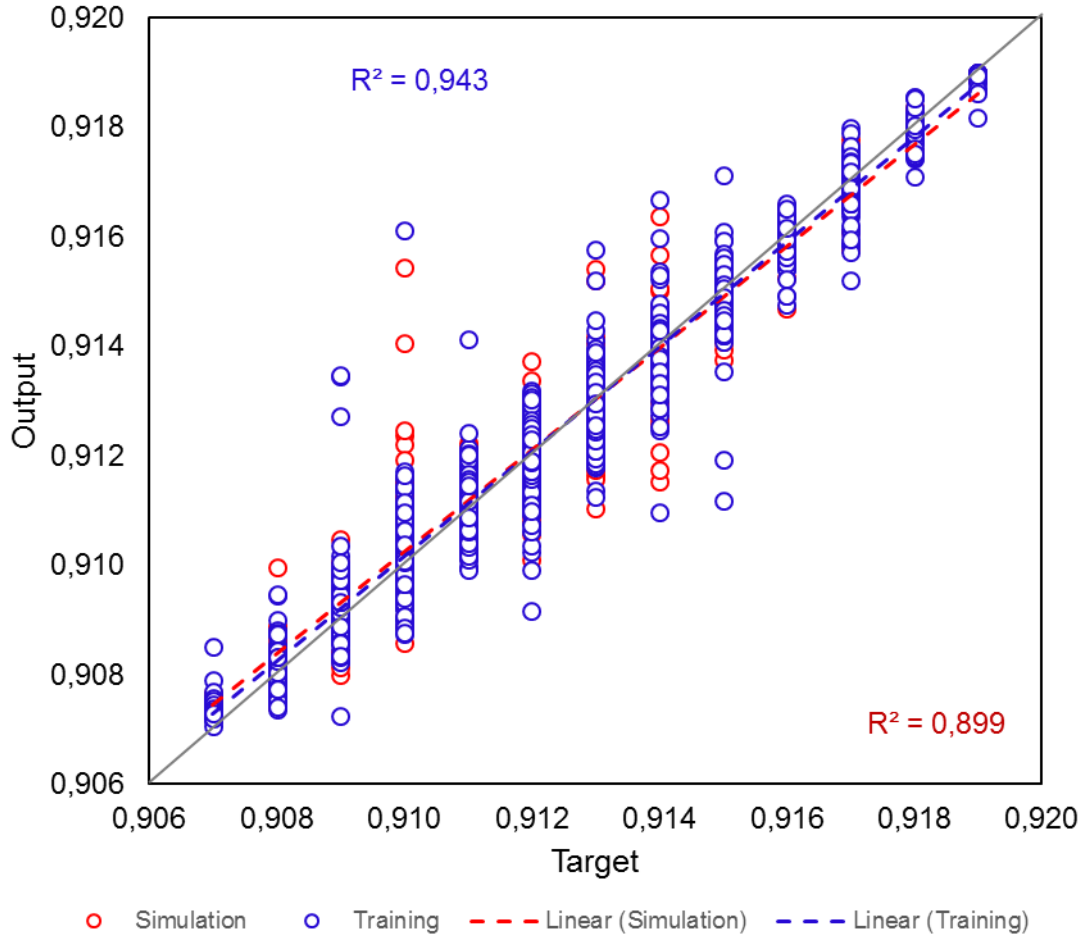


**Figure 5.2-37: Regression plots of the training and simulation of a 65-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data**

The mean-squared error for the training of the abovementioned neural network is  $6.2 \times 10^{-7}$  while the correlation coefficient for training is 0.971. The regression plots obtained for the different phases of assessing this neural network is given in Appendix E.4.1 as Figure E.4-3.

The figure above gives the coefficients of determination for training and simulation of the neural network as 0.943 and 0.899 respectively. The simulation correlation coefficient and mean-squared error is calculated to be 0.948 and  $6.1 \times 10^{-7}$  respectively. This shows that the neural network trains and simulates the estimation of sphericity of particles very well.

Figure 5.2-36 shows that the 65-neuron neural network did not perform as predicted in terms of the mean-squared error or the correlation coefficient. This could be due to an oversimplification of the regression approach as seen in the trend lines not fitting the data very well in Figure 5.2-36. The 65-neuron network did, however still perform the best amongst all the configurations.

The model performance of the abovementioned neural network is depicted in Figure 5.2-38 wherein the operational sphericity data points are combined with their corresponding artificial neural network data points. The figure shows only a part of the data for training as well as simulation, however the full graph is given as raw data in Appendix F.2. The figure shows that the neural network data follows the general tendency of the plant data to a fair extent but leaves room for improvement in light of the variations observed in the neural network's output. The simulation of the neural network seems to perform well in general with only a few outliers (spikes) being very large and thus influencing the performance figures (e.g. the MSE) of the neural network.

#### Bayesian regularisation training algorithm

The Bayesian regularisation training algorithm was tested on different neuron configurations. Table 5.2-20 shows the results.

**Table 5.2-20: Performance of Bayesian regularisation algorithm on phase 2 sphericity data**

|                    | MSE                  | R(training) | R(test) | R(all) |
|--------------------|----------------------|-------------|---------|--------|
| <i>5 neurons</i>   | $1.7 \times 10^{-6}$ | 0.84        | 0.79    | 0.84   |
| <i>10 neurons</i>  | $9.4 \times 10^{-7}$ | 0.92        | 0.88    | 0.92   |
| <i>15 neurons</i>  | $5.2 \times 10^{-7}$ | 0.96        | 0.94    | 0.95   |
| <i>25 neurons</i>  | $4.9 \times 10^{-7}$ | 0.96        | 0.93    | 0.96   |
| <i>50 neurons</i>  | $2.2 \times 10^{-7}$ | 0.98        | 0.96    | 0.98   |
| <i>75 neurons</i>  | $1.6 \times 10^{-7}$ | 0.98        | 0.97    | 0.98   |
| <i>100 neurons</i> | $1.8 \times 10^{-7}$ | 0.99        | 0.96    | 0.98   |

The mean-squared error decreased, and the correlation coefficient increased showing an increase in neural network performance for each increase in the number of neurons up until the 75-neuron configuration. The performance decreased for the 100-neuron configuration, making it possible to perform a regression analysis in order to identify a theoretical best-performing neuron configuration.

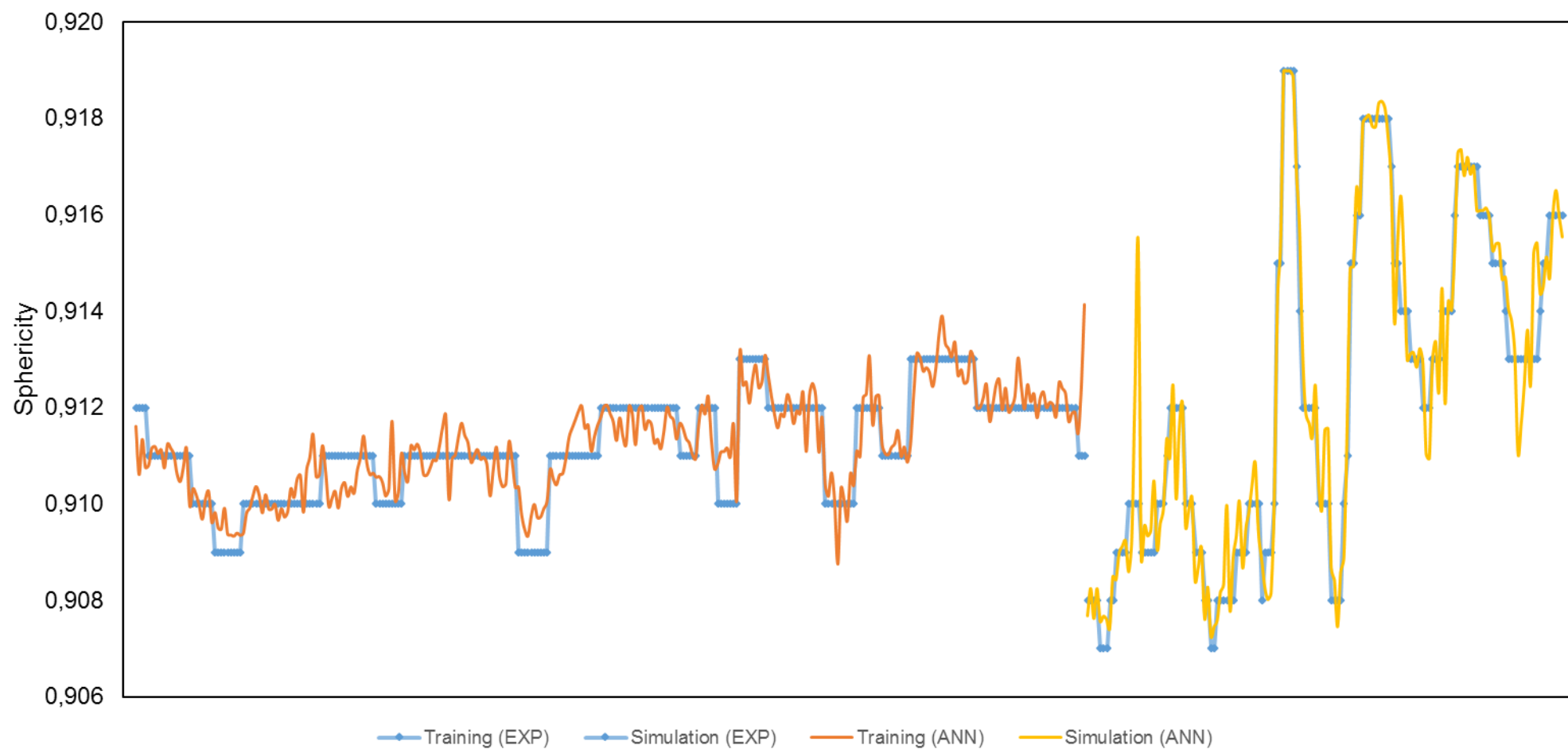


**Figure 5.2-38: Model performance of a 65-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data**

The mean-squared error was calculated as  $1.6 \times 10^{-7}$  and the correlation coefficient of all the data is obtained as 0.984, which indicates that a very good correlation exists between the actual plant data and the data estimated by the neural network and concludes the performance of the network as very good. The coefficient of determination for the training of the 75-neuron neural network is 0.969 while that of the simulation of the network is 0.927. This results in a correlation coefficient of 0.963 for the simulation of the neural network. The mean-squared error was calculated to be  $4.46 \times 10^{-7}$ . The performance of the simulation data is observed to be close to that of the training data, showing that the neural network can predict the sphericity using unseen data very well.

Figure 5.2-9 shows the correlation coefficient and mean-squared error of each of the tested neuron configurations for the Bayesian regularisation training algorithm using phase 2 sphericity estimation data.

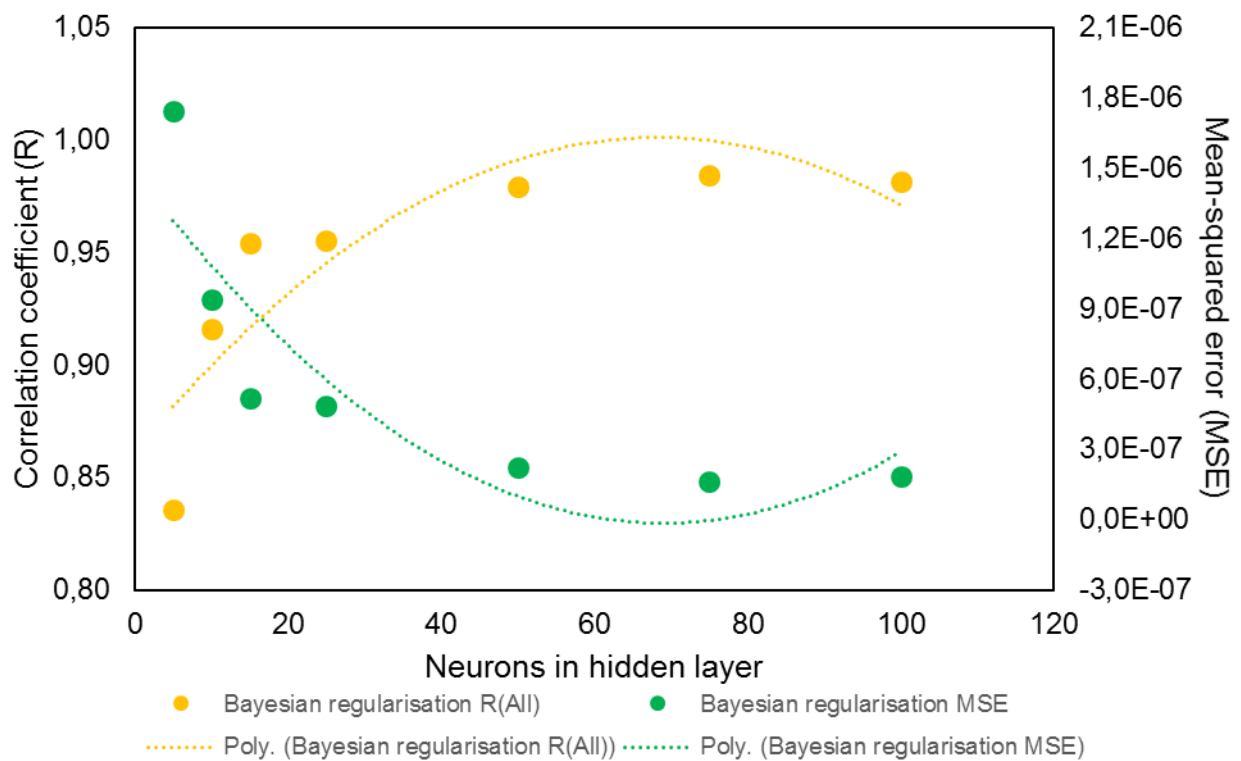


**Figure 5.2-39 MSE-values and R-values for neuron-configurations using the Levenberg-Marquardt training algorithm and phase 2 sphericity data**

As seen with the trend lines in the plot, the trends are quadratic and have a minimum and maximum respectively. The equation of the trend line of the correlation coefficient is given in Equation 5.2-14:

$$y = (-2.99 \times 10^{-5})x^2 + (4.08 \times 10^{-3})x + 8.62 \times 10^{-1} \quad 5.2-14$$

The maximum for the abovementioned quadratic equation is calculated to be at 75 neurons. The equation of the trend line of the mean-squared error is given in Equation 5.2-15:

$$y = (3.16 \times 10^{-10})x^2 + (-4.36 \times 10^{-8})x + 1.49 \times 10^{-6} \quad 5.2-15$$

The minimum for the abovementioned quadratic equation is calculated to be at 76 neurons. This results in an average of 76 neurons, which is close enough to the 75-neuron network already generated. Figure 5.2-40 shows the performance of the abovementioned neural network. The figure only shows a part of the data, while the graph of the full data set is given in Appendix F.2. The figure shows that the neural network data follows the trend of the plant data significantly closer in comparison to the Levenberg-Marquardt network. This shows that the network trained well enough to be a possible inference measurement model for the estimation of granule sphericity.

#### Training algorithm comparison

The comparison of training algorithms is based on the same performance evaluations as for the number of neurons in the hidden layer. Table 5.2-21 lists the best performing neural networks using the phase 2 sphericity data on both the Levenberg-Marquardt and Bayesian regularisation training algorithms.

**Table 5.2-21: Comparison of training algorithms using phase 2 sphericity data**

|                                                   | <b>MSE</b>           | <b>R(training)</b> | <b>R(validation)</b> | <b>R(test)</b> | <b>R(all)</b> |
|---------------------------------------------------|----------------------|--------------------|----------------------|----------------|---------------|
| <b>Levenberg-Marquardt training algorithm</b>     |                      |                    |                      |                |               |
| <i>50 neurons</i>                                 | $7.7 \times 10^{-7}$ | 0.98               | 0.93                 | 0.93           | 0.97          |
| <i>65 neurons</i>                                 | $6.2 \times 10^{-7}$ | 0.98               | 0.95                 | 0.93           | 0.97          |
| <i>75 neurons</i>                                 | $9.4 \times 10^{-7}$ | 0.98               | 0.92                 | 0.94           | 0.97          |
| <b>Bayesian regularisation training algorithm</b> |                      |                    |                      |                |               |
| <i>75 neurons</i>                                 | $1.6 \times 10^{-7}$ | 0.98               |                      | 0.97           | 0.98          |
| <i>100 neurons</i>                                | $1.8 \times 10^{-7}$ | 0.99               |                      | 0.96           | 0.98          |

As seen in the table above the Bayesian regularisation training algorithm fared better than the Levenberg-Marquardt training algorithm in every possible network topology. The dataset is quite large. Thus, favouring the Bayesian regularisation training algorithm.

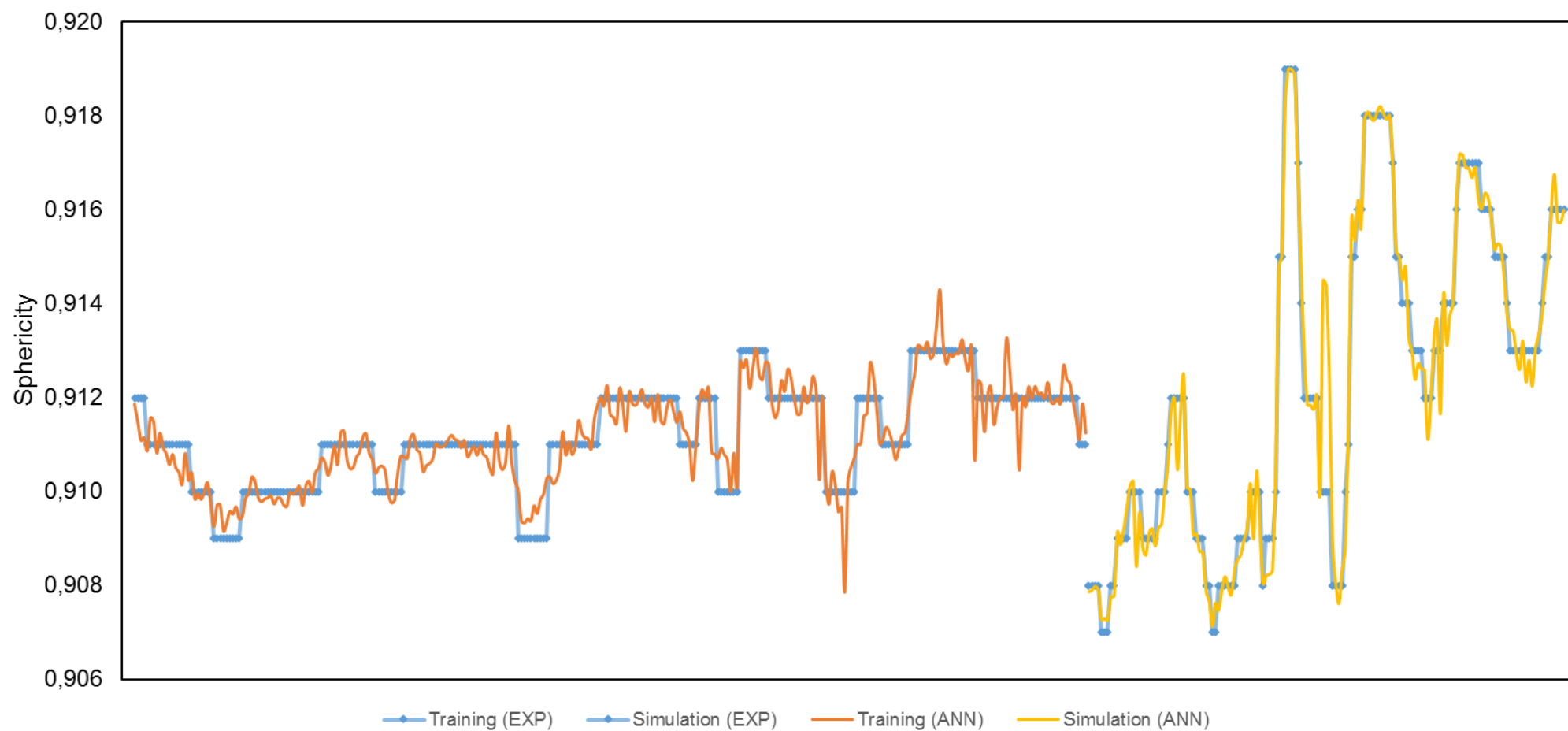


Figure 5.2-40: Model performance of a 75-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 sphericity data

The best trained neural network when using phase 2 sphericity estimation data is obtained with a 75-neuron hidden layer and the Bayesian regularisation training algorithm. The performance of the training as well as simulation of the neural network is very good. The mean-squared error is close to zero while the correlation coefficient is very close to one, showing that the artificial neural network performed very well.

#### Activation function comparison

The best performing network was subjected to an assessment using the sigmoidal and pure linear activation functions instead of the default hyperbolic tangent function. Table 5.2-22 shows the results of the 75-neuron network, trained using the Bayesian regularisation training algorithm and all the above-mentioned activation functions.

**Table 5.2-22: Activation function comparison using phase 2 sphericity data**

|                           | MSE                  | R(training) | R(test) | R(all) |
|---------------------------|----------------------|-------------|---------|--------|
| <i>Hyperbolic tangent</i> | $1.6 \times 10^{-7}$ | 0.98        | 0.97    | 0.98   |
| <i>Sigmoidal</i>          | $3.3 \times 10^{-7}$ | 0.97        | 0.95    | 0.97   |
| <i>Pure linear</i>        | $5.3 \times 10^{-6}$ | 0.41        | 0.43    | 0.41   |

As seen in the table above, the hyperbolic tangent function performed the best between the three activation functions. The results performed as predicted by literature, that the hyperbolic tangent activation function would perform best with the data at hand. The generated regression plots for the abovementioned activation function assessments are provided in Appendix E.4.3.

### **5.3 Chapter summary**

The selectivity variable measurement and validation resulted in the development of an advantages new method for porosity measurement as well as the confirmation that the sphericity measurement used by the industry partner is precise.

The results of the predictive models developed in the first phase of neural network testing was very informative for the acquisition of data for the second phase. The second phase model development resulted in possible usable models for the prediction of both porosity and sphericity.

## Chapter 6: Conclusion and recommendations

### 6.1 Conclusion

#### 6.1.1 Selectivity variable measurement and validation

The quality measurement of a product can only be as accurate as the method with which it is measured allows it to be (Hogekamp & Pohl, 2003). Two of the most important selectivity variables in the granulation of ammonium nitrate for agricultural and explosive purposes is the porosity and sphericity of the product granules. The porosity of the product is currently measured with an oil absorption technique while the sphericity is measured with image analysis.

It is concluded in this study that the current method for porosity measurement, namely the oil absorption method, correlates very well with a newly developed method, namely the optical porosity measurement method. The optical porosity measurement method is less expensive and less time consuming than the oil absorption method, concluding that it could be a viable alternative to the oil absorption method.

The repeatability of the sphericity measurements using image analysis was very good, showing that the results obtained from the optical analyser is precise. This concludes that the measurement currently in use for the determination of particle sphericity as one of the product quality variables should not be altered or replaced.

#### 6.1.2 Artificial neural network

Fluidised bed granulation is a process wherein the product quality variables show very little statistical relationships with any of the operational (input) variables; an observation from studies conducted by Eisenberg (2016) and Kok (2016). Another major problem with the porosity and sphericity of the product particles in the granulation of ammonium nitrate is that these variables cannot be measured on-line. A proven method to overcome these problems is to predict these quality variables using an inference measurement modelling technique. An artificial neural network can be trained to predict output (quality) values even with no apparent relation to the input (operational) values used for the training.

The first phase of predictive model testing concluded that the Levenberg-Marquardt training algorithm performed better with fewer data points, as seen in the discrete data set. The Bayesian regularisation training algorithm outperformed the Levenberg-Marquardt algorithm with larger

data sets such as the filtered data and the constant-time data. The study also concluded that there is no set pattern for the best neuron configuration for the hidden layer of the neural network, however, the larger data sets performed better with many neurons in the hidden layer. The phase 1 results concluded that a neural network trained with data with a constant time interval between new selectivity variables (as with the constant-time data set) as well as noise filtering on the input parameters performed the best.

The second phase of the predictive model testing concluded that a neural network with a large number of neurons in the hidden layer, trained with the Bayesian regularisation training algorithm still performed best with the data set used. The best artificial neural network obtained for the estimation of porosity used noise filtered data for training, the Bayesian regularisation training algorithm and had 61 neurons in the hidden layer. As with porosity estimation, the study concluded that a large number of neurons in the hidden layer combined with the Bayesian regularisation training algorithm performed the best for estimating the granule sphericity of porous granular ammonium nitrate. The best artificial neural network for sphericity estimation had 75 neurons in the hidden layer.

## **6.2 Recommendations**

### **6.2.1 Selectivity variable measurement and validation**

The oil absorption method for porosity measurement of the product particles can easily be replaced by the optical porosity measurement method as it is less time consuming, less expensive and makes use of a larger sample size than the oil absorption method. If mass-based porosity derived from the oil absorption technique is required by the plant, values can be determined using a conversion equation. To obtain a more accurate equation for the conversion between methods, a wider variety of samples must be analysed using both methods.

If the porosity continues to be measured using the oil absorption method, a recommendation is to keep particles in the chamber for longer to ensure full absorption of the oil. Microscopic analyses can be done on samples that have undergone oil absorption to establish if the oil penetrated to the centre of the particles. The particles should also be sieved to a smaller particle range, to ensure that the outer surface area of the particles do not influence the measurement due to adsorption of the oil. A last recommendation for the use of the oil absorption method for porosity measurement is to do multiple measurements from the same sample and use an average value. The reason for this is the sample size for oil absorption is relatively small and one run may not necessarily be representative of the whole sample.

### 6.2.2 Artificial neural network

The estimation ability of an artificial neural network is based on the limits of the input parameters it is trained with, since neural networks cannot extrapolate accurately. The neural networks in this study were trained with limited ranges on the input parameters and can thus only be used when the operating parameters are within those limits. A recommendation would be to improve the predictive ability of the neural networks by acquiring new experimental data with a wider range for all the input parameters and retraining the neural networks to improve their performance. The amount of data used for training also influences the performance of the neural network; the more data there is for training of the neural network, the better it can predict outputs. Thus, a larger dataset as well as wider range of data would be recommended for a better performing predictive model.

The predictive model will act as an inference measurement for on-line quality control of the granulator. A subsequent study can focus on this control strategy to develop a controller for the fluidised bed granulator with the predictive models' outputs as sensory inputs to the control loops.

## Chapter 7: References

- Abeykoon, C. 2018? Design and Applications of Soft Sensors in Polymer Processing: A Review. *IEEE Sensors Journal* (In press)
- Abraham. A. 2004. Meta learning evolutionary artificial neural networks. *Neurocomputing*. 56:1-38
- Aleksić. I., Đuriš. J., Illić. I., Ilbrić. S. & Parojčić. J. 2015. An investigation into the usefulness of different empirical modelling techniques for better control of spray-on fluidized bed melt granulation. *International journal of pharmaceutics*. 496:627-635
- Aleksić. I., Đuriš. J., Illić. I., Ilbrić. S., Parojčić. J. & Srčić. S. 2014. In silico modelling of in situ fluidised bed melt granulation. *International journal of pharmaceutics*. 466:21-30
- Andreola. F., Leonelli. C., Romagnoli. M. & Miselli. P. 2000. Techniques used to determine porosity. *American ceramic society bulletin*. 79(7):49-52
- Aqil. M., Kita. I., Yano. A. & Nishiyama. S. 2007. A comparative study of artificial neural networks and neuro-fuzzy in continuous modelling of the daily and hourly behaviour of runoff. *Journal of hydrology*. 337:22-34
- Azadi. S & Karimi-Jashni. A. 2016. Verifying the performance of artificial neural network and multiple linear regression in predicting the mean seasonal municipal solid waste generation rate: A case study of Fars province. Iran. *Waste Management*. 48(1):14-23
- Becher. R.D. & Schlünder. E.U. 1998. Fluidised bed granulation – the importance of a drying zone for the particle growth mechanism. *Chemical engineering and processing*. 37:1-6
- Bui. D.T., Pradhan. B., Lofman. O., Revhaug. I. & Dick. O.B. 2012. Landslide susceptibility assessment in the Hoa Binh province of Vietnam: A comparison of the Levenberg-Marquardt and Bayesian regularized neural networks. *Geomorphology*. 171-172: 12-29
- Burggraeve. A., Monteyne. T., Vervaet. C., Remon. J.P. & De Beer. T. 2013. Process analytical tools for monitoring, understanding, and control of pharmaceutical fluidized bed granulation: A review. *European journal of pharmaceutics and biopharmaceutics*. 83:2-15
- Burns. R.S. 2001. Advanced control engineering. 1<sup>st</sup> ed. Oxford: Butterworth-Heinemann.

- Chaipimonplin. T. 2016. Investigation internal variables of neural network model for flood forecasting at Upper River Ping, Chiang Mai. *Water Engineering*. 20(1): 478-484
- Chen. T & Chen. H. 1995. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*. 6(4): 911-917
- Cirstea. M.N., Dinu. A., Khor. J.G. & McCormick. M. 2002. Neural and fuzzy logic control of drives and power systems. 1<sup>st</sup> ed. Oxford: Newnes
- Demuth. H. & Beale. M. 2004. Neural network toolbox: For use with MATLAB. 4<sup>th</sup> ed. Natick, MA: The MathWorks Inc.
- Dongare. A.D., Kharde. R.R. & Kachare. A.D. 2012. Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology*. 2(1): 189-194
- Eisenberg, S. 2016. Modelling the influence of physicochemical material feed properties on ammonium nitrate product quality. Potchefstroom: North West University. (Dissertation – Masters)
- Hemati. M., Cherif. R., Saleh. K. & Pont. V. 2003. Fluidised bed coating and granulation: influence of process related variables and physicochemical properties on the growth kinetics. *Powder technology*. 130:18-34
- Hogekamp. S. & Pohl. M. 2003. Porosity measurement of fragile agglomerates. *Powder technology*. 130:385-392
- Kaur. H. & Salaria. D.S. 2013. Bayesian regularisation based neural network tool for software effort estimation. *Global journal of computer science and technology*. 8(2):45-50
- Kasiviswanathan. K.S., Sudheer. K.P. & He. J. 2016. Quantification of prediction uncertainty in artificial neural network models. (In Shanmuganathan S. & Samarasinghe. S. ed. *Artificial neural network modelling*. Switzerland: Springer international publishing p.145-159)
- Kayri. M. 2016. Predictive abilities of Bayesian regularisation and Levenberg-Marquardt algorithms in artificial neural networks: A comparative empirical study on social data. *Mathematical and computational applications*. 21(20)1-11

- Khafaf. N.A. & El-Hag. A. 2018. Bayesian regularisation of neural network to predict leakage current in salt fog environment. *IEEE Transactions on dielectrics and electrical insulation*. 25(2):686-693
- Klimasauskas. C.C. 1991. Applying neural networks. Part 3: Training a neural network. *PC-AI*. 5(3):20-24
- Klobes. P., Meyer. K. & Munro. R.G. 2006. Porosity and specific surface area measurement for solid materials. Special ed. Washington: US Government printing office.
- Kok. R.L. 2016. A statistical model of an ammonium nitrate fluidised bed granulator. Potchefstroom: North West University. (Dissertation – Masters)
- Levenberg. K. 1944. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*. 2(2):164-168
- Link. K.C. & Schlünder. E.U. 1997. Fluidised bed spray granulation: investigation of the coating process on a single sphere. *Chemical engineering and processing*. 36:443-457
- Lotspeich. E. & Petr. V. 2015. The characterisation of ammonium nitrate mini-prills. (In Song et al., eds. Dynamic behaviour of materials. volume 1: proceedings of the 2014 annual conference on experimental and applied mathematics organised by the Society for experimental mechanics series. p. 319-324
- Madsen. K., Nielsen. H.B. & Tingleff. O. 2004. Methods for non-linear least squared problems. 2<sup>nd</sup> ed. Denmark: Technical University of Denmark.
- Mahajan. O.P. & Walker. P.L. 1978. Porosity of coals and coal products. (In Analytical methods for coal and coal products. Volume 1. p. 68-84)
- Marquardt. D.W. 1963. An algorithm for least-squares estimation of nonlinear variables. *Journal of the society for industrial and applied mathematics*. 11(2):431-441
- Mel'gunov. M.S. & Ayupov. A.B. 2017. Direct method for evaluation of BET adsorbed monolayer capacity. *Micro-porous and meso-porous materials*. 243:147-153
- Moraga. S.V., Villa. M.P., Bertín. D.E., Cotabarren. I.M., Piña. J., Pedernera. M. & Bucalá. V. 2015. Fluidized-bed melt granulation: The effect of operating parameters on process performance and granule properties. *Powder technology*. 286:654-667

- Patel. T.B., Patel. L.D., Patel. T.B., Makwana. S.H. & Patel. T.R. 2010. Influence of process variables on physicochemical properties of the granulation mechanism of diclofenac sodium in fluid bed granulation. *International journal of pharmaceutical sciences review and research*. 3(1):61-65
- Payal. A., Rai. C.S. & Reddy. B.V.R. 2013. Comparative analysis of Bayesian regularisation and Levenberg-Marquardt training algorithm for localization in wireless sensor network. (*In The 14<sup>th</sup> International Conference on Advanced Communication Technology*. p. 191 – 194)
- Payal. A., Rai. C.S. & Reddy. B.V.R. 2015. Analysis of some feedforward artificial neural network training algorithms for developing localization framework in wireless sensor networks. *Wireless personal communications*. 82(4):2519-2536
- Pecht. M., Dasgupta. A., Evans. J.W. & Evans. J.Y. 1994. Quality conformance and qualification of microelectronic packages and interconnects. 1<sup>st</sup> ed. New York: John Wiley & Sons
- Perry. R.H. & Green. D.W. 1997. Perry's chemical engineer's handbook. 7<sup>th</sup> ed. New York: McGraw Hill
- Pont. V., Saleh. K., Steinmetz. D. & Hémati. M. 2001. Influence of the physicochemical properties on the growth of solid particles by granulation in fluidised bed. *Powder technology*. 120:97-104
- Rämet. P., Lehtola. S., Leiviskä. K. & Mannermaa. J.P. 1996. The fluidized bed granulation process modelling and control using neural networks and fuzzy control. Paper presented at the Fourth European congress on intelligent techniques and soft computing. [https://www.researchgate.net/profile/Kauko\\_Leiviskae/publication/306952669\\_The\\_Fluidized\\_Bed\\_Granulation\\_Process\\_Modelling\\_and\\_Control\\_Using\\_Neural\\_Networks\\_and\\_Fuzzy\\_Control/links/57c06ef808ae2f5eb3322153/The-Fluidized-Bed-Granulation-Process-Modelling-and-Control-Using-Neural-Networks-and-Fuzzy-Control.pdf?origin=publication\\_detail](https://www.researchgate.net/profile/Kauko_Leiviskae/publication/306952669_The_Fluidized_Bed_Granulation_Process_Modelling_and_Control_Using_Neural_Networks_and_Fuzzy_Control/links/57c06ef808ae2f5eb3322153/The-Fluidized-Bed-Granulation-Process-Modelling-and-Control-Using-Neural-Networks-and-Fuzzy-Control.pdf?origin=publication_detail). Date of access: 1 Feb 2017
- Rhodes. M. 2008. Introduction to Particle Technology. 2<sup>nd</sup> ed. England: John Wiley & Sons
- Riley. C.M., Rose. W.I. & Bluth. G.J.S. 2003. Quantitative shape measurements of distal volcanic ash. *Journal of geophysical research*. 108(B10):8-1 – 8-15

- Ronco. E. & Gawthrop. P.J. 1997. Neural networks for modelling and control: Technical report. <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=A3502FAB5EDC3B312AB7AB97C47EDC8F?doi=10.1.1.49.9247&rep=rep1&type=pdf>. Date of access: 9 Jan 2018
- Rouquerol. J., Avnir. D., Fairbridge. C.W., Everett. D.H., Haynes. J.H., Pernicone. N., Ramsay. J.D.F., Sing. K.S.W. & Unger. K.K. 1994. Recommendations for the characterisation of porous solids. *Pure and applied chemistry*. 66(8):1739-1758
- Samadian. H., Zakariaee. S.S., Adabi. M., Mobasheri. H., Azami. M. & Faridi-Majidi. R. 2016. Effective variables on conductivity of mineralized carbon nano-fibers: an investigation using artificial neural networks. *Royal society of chemistry*. 6:111908-111918
- Satish. S. & Setty. Y.P. 2005. Modelling of a continuous fluidized bed dryer using artificial neural networks. *International communications in heat and mass transfer*. 32(3-4): 539-547
- Sing. K. 2001. The use of nitrogen adsorption for the characterisation of porous materials. *Colloids and surfaces A: Physicochemical and engineering aspects*. 187-188:3-9
- Srinivasakannan. C. & Balasubramaniam. N. 2003. Particle growth in fluidised bed granulation. *Chemical and biochemical engineering*. 17(3):201-205
- Ulusoy. U. & Kursun. I. 2011. Comparison of different 2D image analysis measurement techniques for the shape of talc particles produces by different media milling. *Minerals engineering*. 24:91-97
- Vengateson. U. & Mohan. R. 2016. Experimental and modelling study of fluidised bed granulation: effect of binder flow rate and fluidising air velocity. *Resource-Efficient Technologies*. 2:S124-S135
- Yu. H. & Wilamowski. B. 2011. Chapter 12: Levenberg-Marquardt training. (In Yu. H. & Wilamowski. B. 2<sup>nd</sup> ed. Industrial electronics handbook: Intelligent systems Volume 5. CRC Press. p. 12-1 – 12-15)
- Zhang. G., Patuwo. B.E. & Hu. M.Y. 1998. Forecasting with artificial neural networks: A state of the art. *International journal of forecasting*. 14(1):35-62
- Zou. Y. & Malzbender. J. 2016. Development and optimisation of porosity measurement techniques. *Ceramics international*. 42:2861-2870

## Appendix A: Experimental calculations and procedures

### A.1 Oil absorption

#### A.1.1 Outliers test

AN outlier test was done on the oil absorption of samples 1 and 8. The test done using standard box-and-whisker diagrams. An outlier was defined as any value outside 1.5 times the interquartile range from the first and third quartile.

Table A.1-1 shows the experimental values of sample 1 and Table A.1-2 shows the calculations done to obtain the box plot. Figure A.1-1 shows the box-and-whisker diagram with the acceptable range of values.

**Table A.1-1: Sample 1 experimental values - mass oil absorbed**

| Sample 1 experimental values |       |
|------------------------------|-------|
|                              | 16.8% |
|                              | 45.4% |
|                              | 15.0% |
|                              | 15.8% |
|                              | 21.3% |

**Table A.1-2: Five-number summary for sample 1 - mass oil absorbed**

|                            |       |
|----------------------------|-------|
| <b>Mean</b>                | 22.9% |
| <b>Standard deviation</b>  | 12.8% |
| <b>Min</b>                 | 15.0% |
| <b>Q<sub>1</sub></b>       | 15.8% |
| <b>Median</b>              | 16.8% |
| <b>Q<sub>3</sub></b>       | 21.3% |
| <b>Max</b>                 | 45.4% |
| <b>Interquartile range</b> | 5.5%  |
| <b>Lower limit</b>         | 7.5%  |
| <b>Upper limit</b>         | 29.6% |

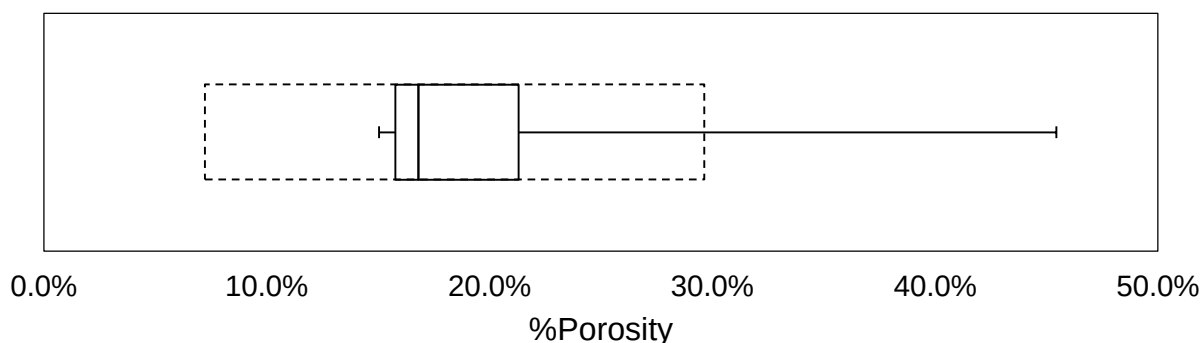


**Figure A.1-1: Box-and-whisker diagram of sample 1 - mass oil absorbed**

The box-and-whisker diagram shows a lower limit of 7.5% porosity and an upper limit of 29.6%. This means that the value of 45.4% must be excluded from any dataset.

Table A.1-3 and Table A.1-4 shows the data for the determination of outliers for sample 8 experimental values. Figure A.1-2 shows the box-and-whisker diagram for the sample.

**Table A.1-3: Sample 8 experimental values - mass oil absorbed**

| <b>Sample 8 experimental values</b> |       |
|-------------------------------------|-------|
|                                     | 20.0% |
|                                     | 21.3% |
|                                     | 28.3% |
|                                     | 21.9% |
|                                     | 23.2% |

**Table A.1-4: Five-number summary for sample 8 - mass oil absorbed**

|                            |       |
|----------------------------|-------|
| <b>Mean</b>                | 23.0% |
| <b>Standard deviation</b>  | 3.2%  |
| <b>Min</b>                 | 20.0% |
| <b>Q<sub>1</sub></b>       | 21.3% |
| <b>Median</b>              | 21.9% |
| <b>Q<sub>3</sub></b>       | 23.2% |
| <b>Max</b>                 | 28.3% |
| <b>Interquartile range</b> | 1.9%  |
| <b>Lower limit</b>         | 18.5% |
| <b>Upper limit</b>         | 26.0% |

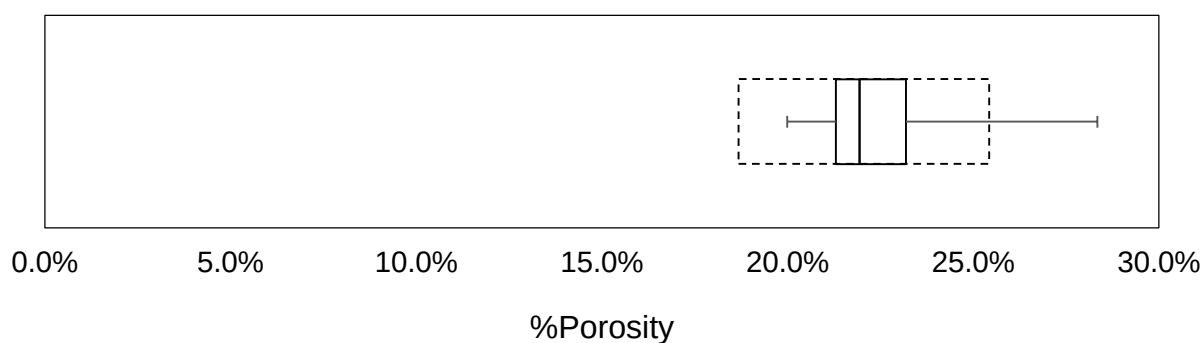


**Figure A.1-2: Box-and-whisker diagram of sample 8 - mass oil absorbed**

The box-and-whisker diagram for sample 8 shows one outlier namely the porosity value of 28.3%. The upper and lower limits for sample 8 experiments are 26.0% and 8.5% respectively.

### A.1.2 Repeatability calculations

Repeatability calculations were done on samples 1, 5 and 8. The data for each sample is given in Table A.1-5:

**Table A.1-5: Repeatability results for oil absorption - volume and mass**

| Sample | V <sub>liq absorbed</sub><br>[%] | m <sub>liq absorbed</sub><br>[%] |
|--------|----------------------------------|----------------------------------|
| 1.1    | 16.8%                            | 8.5%                             |
| 1.2    | 15.0%                            | 7.5%                             |
| 1.3    | 15.8%                            | 7.9%                             |
| 1.4    | 21.3%                            | 11.0%                            |
| 5.1    | 21.9%                            | 11.4%                            |
| 5.2    | 29.8%                            | 16.2%                            |
| 5.3    | 28.2%                            | 15.3%                            |
| 8.1    | 20.0%                            | 10.3%                            |
| 8.2    | 21.3%                            | 11.0%                            |
| 8.3    | 21.9%                            | 11.4%                            |
| 8.4    | 23.2%                            | 12.1%                            |

The repeatability of the samples is based on the mean (Equation A.1-1) and standard deviation (Equation A.1-2), both calculated using built-in Microsoft Excel functions.

## Appendix A: Experimental calculations and procedures

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad \text{A.1-1}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n - 1}} \quad \text{A.1-2}$$

The error and percentage error were calculated using Equation A.1-3 and A.1-4 respectively.

$$Error = \frac{\sigma}{\sqrt{n}} \times t_{n-1} \quad \text{A.1-3}$$

$$\%Error = \frac{Error}{\bar{x}} \quad \text{A.1-4}$$

The  $t_{n-1}$  is given in Table A.1-6. obtained from Figure A.2- in Appendix A.2. Table A.1-7 gives the repeatability results for both porosity percentage and mass of liquid absorbed percentage.

**Table A.1-6: Critical t-values for oil absorption repeatability**

| Constants |       |
|-----------|-------|
| $t_2$     | 4.303 |
| $t_3$     | 3.182 |

**Table A.1-7: Repeatability calculations and results for oil absorption**

| Sample | Repeatability - %V <sub>liq</sub> absorbed |                        |       |        | Repeatability - %m <sub>liq</sub> absorbed |                        |       |        |
|--------|--------------------------------------------|------------------------|-------|--------|--------------------------------------------|------------------------|-------|--------|
|        | Mean ( $\bar{x}$ )                         | Standard deviation (s) | Error | %Error | Mean ( $\bar{x}$ )                         | Standard deviation (s) | Error | %Error |
| 1      | 17%                                        | 2.8%                   | 4%    | 25.9%  | 9%                                         | 1.6%                   | 3%    | 28.9%  |
| 5      | 27%                                        | 4.1%                   | 9%    | 33.5%  | 14%                                        | 2.6%                   | 6%    | 38.5%  |
| 8      | 22%                                        | 1.3%                   | 2%    | 9.8%   | 11%                                        | 0.8%                   | 1%    | 11.1%  |

## A.2 Critical values for confidence interval

Figure A.2- shows the t critical values for confidence interval (Devore & Farnum. cited by Kok. 2016).

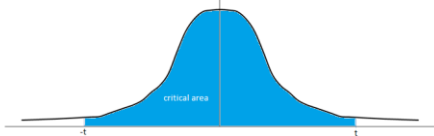
|  |                                          |       |        |        |        |         |         |
|------------------------------------------------------------------------------------|------------------------------------------|-------|--------|--------|--------|---------|---------|
| Degrees of freedom                                                                 | Confidence level for two-sided interval: |       |        |        |        |         |         |
|                                                                                    | 80%                                      | 90%   | 95%    | 98%    | 99%    | 99,8%   | 99,9%   |
|                                                                                    | Confidence level for one-sided interval: |       |        |        |        |         |         |
|                                                                                    | 90%                                      | 95%   | 98%    | 99%    | 99,8%  | 99,9%   | 99,95%  |
| 1                                                                                  | 3.078                                    | 6.314 | 12.706 | 31.821 | 63.657 | 318.313 | 636,620 |
| 2                                                                                  | 1.886                                    | 2.920 | 4.303  | 6.965  | 9.925  | 22.327  | 31,598  |
| 3                                                                                  | 1.638                                    | 2.353 | 3.182  | 4.541  | 5.841  | 10.215  | 12,924  |
| 4                                                                                  | 1.533                                    | 2.132 | 2.776  | 3.747  | 4.604  | 7.173   | 8,610   |
| 5                                                                                  | 1.476                                    | 2.015 | 2.571  | 3.365  | 4.032  | 5.893   | 6,869   |
| 6                                                                                  | 1.440                                    | 1.943 | 2.447  | 3.143  | 3.707  | 5.208   | 5,959   |
| 7                                                                                  | 1.415                                    | 1.895 | 2.365  | 2.998  | 3.499  | 4.782   | 5,408   |
| 8                                                                                  | 1.397                                    | 1.860 | 2.306  | 2.896  | 3.355  | 4.499   | 5,041   |
| 9                                                                                  | 1.383                                    | 1.833 | 2.262  | 2.821  | 3.250  | 4.296   | 4,781   |
| 10                                                                                 | 1.372                                    | 1.812 | 2.228  | 2.764  | 3.169  | 4.143   | 4,587   |

Figure A.2-1: t-critical values for confidence interval

## Appendix B: Results

### B.1 Oil absorption

Table B.1-1 gives the constants used in the calculation of liquid absorption data. while Table B.1-2 shows the results obtained for all samples. including repeated samples.

**Table B.1-1: Constants for oil absorption calculations**

| Constants                |                         |
|--------------------------|-------------------------|
| $m_{\text{granules}}$    | 2.1 g                   |
|                          | 0.0021 kg               |
| $\rho_{\text{particle}}$ | 1725 kg/m <sup>3</sup>  |
| $V_{\text{granules}}$    | 1.22E-06 m <sup>3</sup> |
| $\rho_{\text{liquid}}$   | 789 kg/m <sup>3</sup>   |

**Table B.1-2: Oil absorption calculations**

| Sample | $m_{\text{container}}$<br>[g] | $m_{\text{after}}$<br>[g] | $m_{\text{liq absorbed}}$<br>[g] | $m_{\text{liq absorbed}}$<br>[kg] | $V_{\text{pore}}$<br>[m <sup>3</sup> ] | Porosity<br>[%] | $m_{\text{liq absorbed}}$<br>[%] |
|--------|-------------------------------|---------------------------|----------------------------------|-----------------------------------|----------------------------------------|-----------------|----------------------------------|
| 1.1    | 6.760                         | 9.054                     | 0.194                            | 0.000194                          | 2.46E-07                               | 16.8%           | 8.5%                             |
| 1.2    | 6.840                         | 9.110                     | 0.170                            | 0.000170                          | 2.15E-07                               | 15.0%           | 7.5%                             |
| 1.3    | 8.000                         | 10.280                    | 0.180                            | 0.000180                          | 2.28E-07                               | 15.8%           | 7.9%                             |
| 1.4    | 8.340                         | 10.700                    | 0.260                            | 0.000260                          | 3.30E-07                               | 21.3%           | 11.0%                            |
| 2      | 6.870                         | 9.250                     | 0.280                            | 0.000280                          | 3.55E-07                               | 22.6%           | 11.8%                            |
| 3      | 6.870                         | 9.373                     | 0.403                            | 0.000403                          | 5.11E-07                               | 29.6%           | 16.1%                            |
| 4      | 6.670                         | 9.190                     | 0.420                            | 0.000420                          | 5.32E-07                               | 30.4%           | 16.7%                            |
| 5.1    | 6.760                         | 9.130                     | 0.270                            | 0.000270                          | 3.42E-07                               | 21.9%           | 11.4%                            |
| 5.2    | 6.043                         | 8.550                     | 0.407                            | 0.000407                          | 5.16E-07                               | 29.8%           | 16.2%                            |
| 5.3    | 6.042                         | 8.520                     | 0.378                            | 0.000378                          | 4.79E-07                               | 28.2%           | 15.3%                            |
| 6      | 6.049                         | 8.580                     | 0.431                            | 0.000431                          | 5.46E-07                               | 31.0%           | 17.0%                            |
| 7      | 6.670                         | 9.230                     | 0.460                            | 0.000460                          | 5.83E-07                               | 32.4%           | 18.0%                            |
| 8.1    | 6.750                         | 9.090                     | 0.240                            | 0.000240                          | 3.04E-07                               | 20.0%           | 10.3%                            |
| 8.2    | 6.750                         | 9.110                     | 0.260                            | 0.000260                          | 3.30E-07                               | 21.3%           | 11.0%                            |
| 8.3    | 8.240                         | 10.610                    | 0.270                            | 0.000270                          | 3.42E-07                               | 21.9%           | 11.4%                            |
| 8.4    | 8.320                         | 10.710                    | 0.290                            | 0.000290                          | 3.68E-07                               | 23.2%           | 12.1%                            |

## B.2 Optical porosity measurement method

The constant used in the calculation of porosity using the optical porosity measurement method is given in Table B.2-1. while the results are given in Table B.2-2.

**Table B.2-1: Constants for optical porosity measurement calculations**

| Constant              |                        |
|-----------------------|------------------------|
| $\rho_{\text{solid}}$ | 1725 kg/m <sup>3</sup> |

**Table B.2-2: optical porosity measurement calculations**

| Sample | d <sub>50</sub><br>[mm] | V <sub>particle</sub><br>[m <sup>3</sup> ] | Particle<br>count | Mean<br>circularity | V <sub>total</sub><br>[m <sup>3</sup> ] | Mass<br>[g] | Mass<br>[kg] | Porosity<br>[%] |
|--------|-------------------------|--------------------------------------------|-------------------|---------------------|-----------------------------------------|-------------|--------------|-----------------|
| 1      | 2.586                   | 9.05E-09                                   | 7181              | 0.750               | 4.88E-05                                | 78.425      | 0.078425     | 6.8%            |
| 2      | 2.561                   | 8.79E-09                                   | 6831              | 0.730               | 4.39E-05                                | 68.456      | 0.068456     | 9.5%            |
| 3      | 2.515                   | 8.33E-09                                   | 7231              | 0.741               | 4.46E-05                                | 65.456      | 0.065456     | 15.0%           |
| 4      | 2.593                   | 9.13E-09                                   | 7381              | 0.743               | 5.01E-05                                | 74.036      | 0.074036     | 14.3%           |
| 5      | 2.540                   | 8.58E-09                                   | 7706              | 0.727               | 4.81E-05                                | 74.376      | 0.074376     | 10.3%           |
| 6      | 2.562                   | 8.81E-09                                   | 8345              | 0.680               | 5.00E-05                                | 76.298      | 0.076298     | 11.5%           |
| 7      | 2.595                   | 9.15E-09                                   | 6702              | 0.701               | 4.30E-05                                | 61.177      | 0.061177     | 17.5%           |
| 8      | 2.390                   | 7.15E-09                                   | 9030              | 0.804               | 5.19E-05                                | 79.619      | 0.079619     | 11.1%           |

## B.3 Sphericity

The mean circularity repeatability results are shown in Table B.3-1.

**Table B.3-1: Sphericity repeatability results**

| Sample<br>number | Mean circularity          |                           |                           |                       |                              | Error  | %Error |
|------------------|---------------------------|---------------------------|---------------------------|-----------------------|------------------------------|--------|--------|
|                  | RUN1<br>(x <sub>1</sub> ) | RUN2<br>(x <sub>2</sub> ) | RUN3<br>(x <sub>3</sub> ) | Mean<br>( $\bar{x}$ ) | Standard<br>deviation<br>(s) |        |        |
| 1                | 0.705                     | 0.711                     | 0.711                     | 0.709                 | 0.0035                       | 0.0086 | 1.2%   |
| 2                | 0.732                     | 0.731                     | 0.727                     | 0.730                 | 0.0026                       | 0.0066 | 0.9%   |
| 6                | 0.707                     | 0.708                     | 0.706                     | 0.707                 | 0.0010                       | 0.0025 | 0.4%   |
| 7                | 0.697                     | 0.701                     | 0.700                     | 0.699                 | 0.0021                       | 0.0052 | 0.7%   |
| 8                | 0.814                     | 0.813                     | 0.813                     | 0.813                 | 0.0006                       | 0.0014 | 0.2%   |

## Appendix C: Phase 1 data processing

### C.1 Raw data: Statistical filtering

Table C.1-1 to Table C.1-7 gives the five-number summary used for the box-and-whisker diagrams (Figure C.1- to Figure C.1-7) for each variable.

#### C.1.1 Fluidising air flow rate

Table C.1-1: Five-number summary - Fluidising air flow rate

|                               |       |             |
|-------------------------------|-------|-------------|
| <b>Standard deviation</b>     | 1425  |             |
| <b>Five-number summary</b>    |       |             |
| <b>Minimum</b>                | 505   |             |
| <b>-2× Standard deviation</b> | 8845  | Lower limit |
| <b>Mean</b>                   | 11695 |             |
| <b>+2× Standard deviation</b> | 14544 | Upper limit |
| <b>Maximum</b>                | 16328 |             |

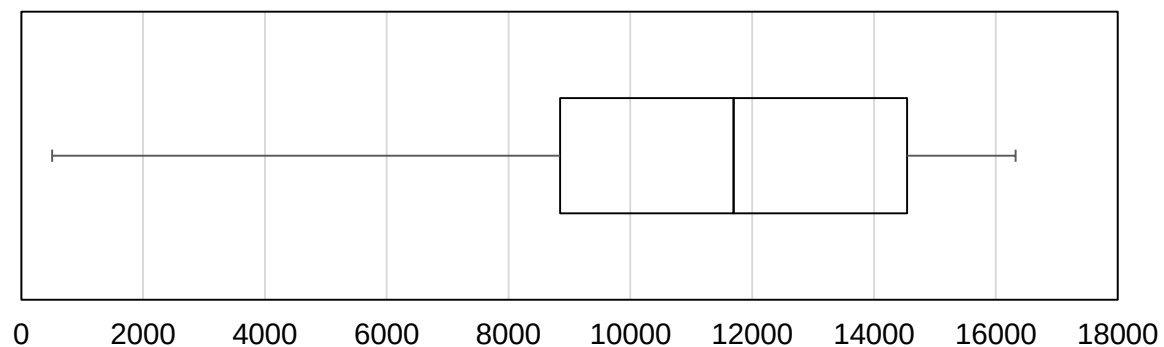


Figure C.1-1: Box-and-whisker diagram - Fluidising air flow rate

The lower and upper limits for fluidising air flow rate are 8845 and 14544 respectively.

### C.1.2 Fluidising air temperature

Table C.1-2: Five-number summary - Fluidising air temperature

|                               |     |             |
|-------------------------------|-----|-------------|
| <b>Standard deviation</b>     | 11  |             |
| <b>Five-number summary</b>    |     |             |
| <b>Minimum</b>                | 80  |             |
| <b>-2× Standard deviation</b> | 98  | Lower limit |
| <b>Mean</b>                   | 120 |             |
| <b>+2× Standard deviation</b> | 143 | Upper limit |
| <b>Maximum</b>                | 147 |             |

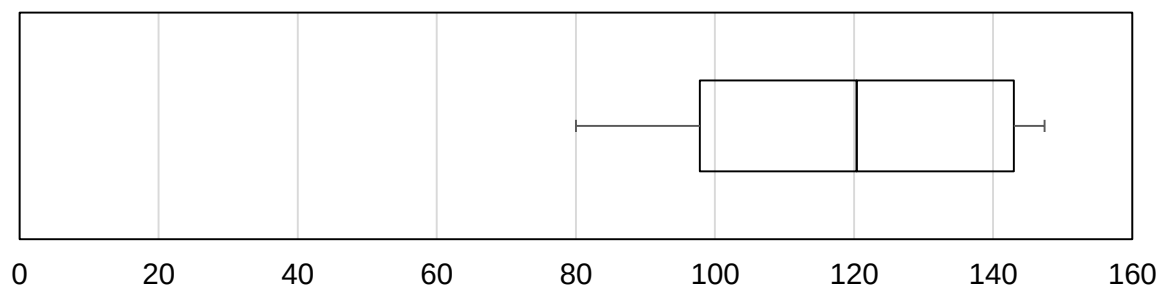


Figure C.1-2: Box-and-whisker diagram - Fluidising air temperature

The lower and upper limits for fluidising air temperature are 98 and 143 respectively.

### C.1.3 Liquid AN density

Table C.1-3: Five-number summary - Liquid ammonium nitrate density

|                               |      |             |
|-------------------------------|------|-------------|
| <b>Standard deviation</b>     | 0.02 |             |
| <b>Five-number summary</b>    |      |             |
| <b>Minimum</b>                | 1.04 |             |
| <b>-2× Standard deviation</b> | 1.35 | Lower limit |
| <b>Mean</b>                   | 1.39 |             |
| <b>+2× Standard deviation</b> | 1.43 | Upper limit |
| <b>Maximum</b>                | 1.48 |             |

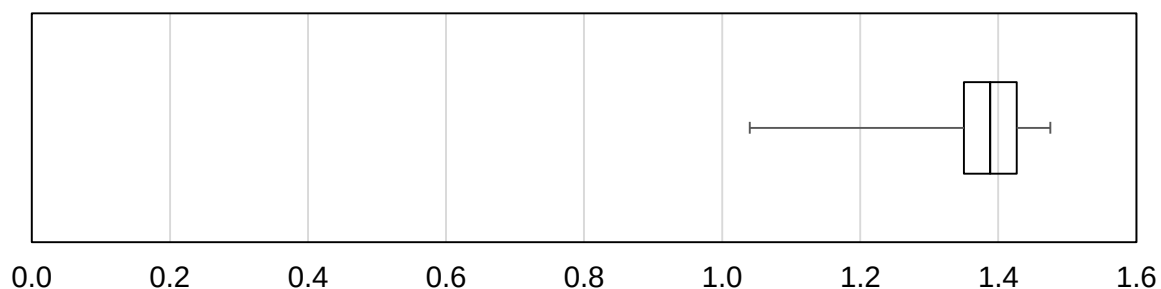


**Figure C.1-3: Box-and-whisker diagram - Liquid ammonium nitrate density**

The lower and upper limits for liquid ammonium nitrate density are 1.35 and 1.43 respectively.

#### C.1.4 Liquid AN temperature

**Table C.1-4: Five-number summary - Liquid ammonium nitrate temperature**

|                               |     |             |
|-------------------------------|-----|-------------|
| <b>Standard deviation</b>     | 10  |             |
| <b>Five-number summary</b>    |     |             |
| <b>Minimum</b>                | 100 |             |
| <b>-2× Standard deviation</b> | 118 | Lower limit |
| <b>Mean</b>                   | 138 |             |
| <b>+2× Standard deviation</b> | 158 |             |
| <b>Maximum</b>                | 151 | Upper limit |

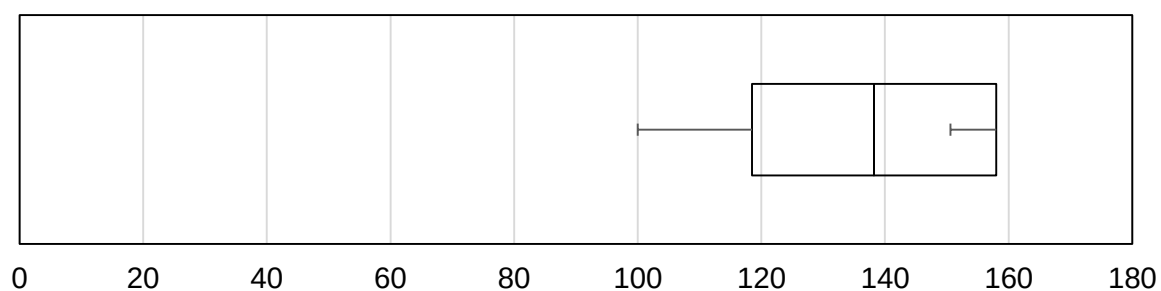


**Figure C.1-4: Box-and-whisker diagram - Liquid ammonium nitrate temperature**

The lower and upper limits for liquid ammonium nitrate temperature are 118 and 151 respectively.

### C.1.5 Liquid AN flow rate

Table C.1-5: Five-number summary - Liquid ammonium nitrate flow rate

|                               |      |             |
|-------------------------------|------|-------------|
| <b>Standard deviation</b>     | 2.9  |             |
| <b>Five-number summary</b>    |      |             |
| <b>Minimum</b>                | 2.3  |             |
| <b>-2× Standard deviation</b> | 2.3  | Lower limit |
| <b>Mean</b>                   | 8.1  |             |
| <b>+2× Standard deviation</b> | 13.8 | Upper limit |
| <b>Maximum</b>                | 50.9 |             |

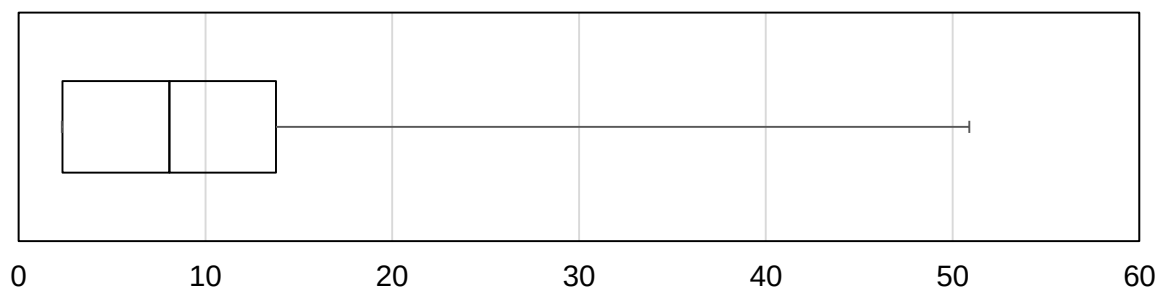


Figure C.1-5: Box-and-whisker diagram - Liquid ammonium nitrate flow rate

The lower and upper limits for liquid ammonium nitrate flow rate are 2.3 and 13.8 respectively.

### C.1.6 Atomising air flow rate

Table C.1-6: Five-number summary - Atomising air flow rate

|                               |     |             |
|-------------------------------|-----|-------------|
| <b>Standard deviation</b>     | 52  |             |
| <b>Five-number summary</b>    |     |             |
| <b>Minimum</b>                | 81  |             |
| <b>-2× Standard deviation</b> | 322 | Lower limit |
| <b>Mean</b>                   | 425 |             |
| <b>+2× Standard deviation</b> | 529 | Upper limit |
| <b>Maximum</b>                | 821 |             |

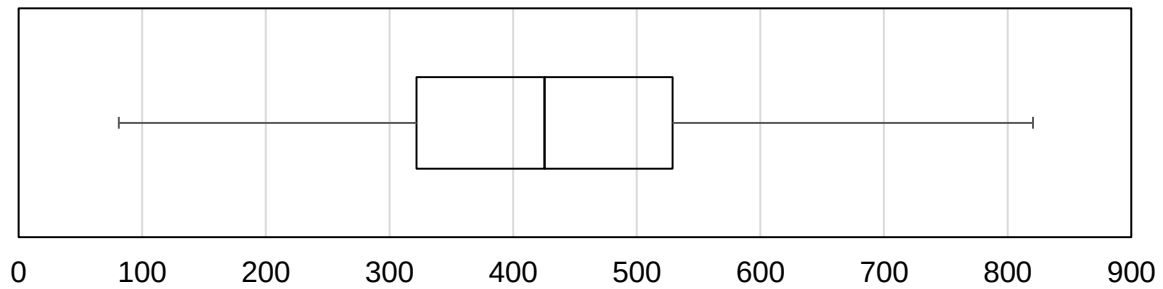


Figure C.1-6: Box-and-whisker diagram - Atomising air flow rate

The lower and upper limits for atomising air flow rate are 322 and 529 respectively.

### C.1.7 Oil absorption

Table C.1-7: Five-number summary - Oil absorption

|                               |      |             |
|-------------------------------|------|-------------|
| <b>Standard deviation</b>     | 1.3  |             |
| <b>Five-number summary</b>    |      |             |
| <b>Minimum</b>                | 9.3  |             |
| <b>-2× Standard deviation</b> | 10.1 | Lower limit |
| <b>Mean</b>                   | 12.7 |             |
| <b>+2× Standard deviation</b> | 15.3 | Upper limit |
| <b>Maximum</b>                | 20.8 |             |

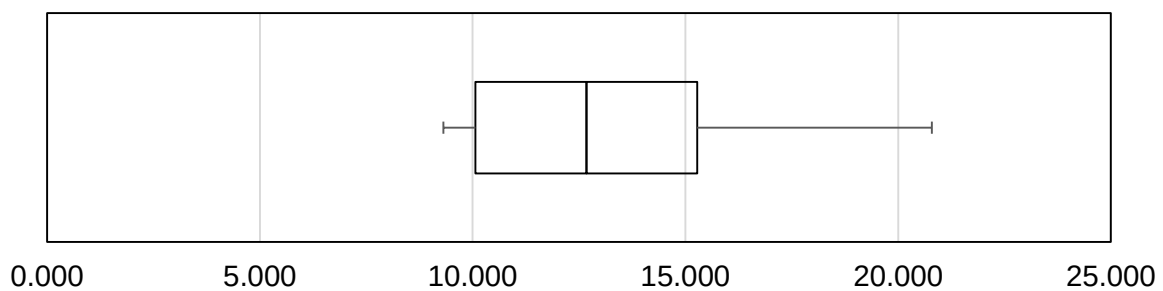


Figure C.1-7: Box-and-whisker diagram - Oil absorption

The lower and upper limits for oil absorption are 10.1 and 15.3 respectively.

## C.2 Phase 1 data graphs

### C.2.1 Original data graphs

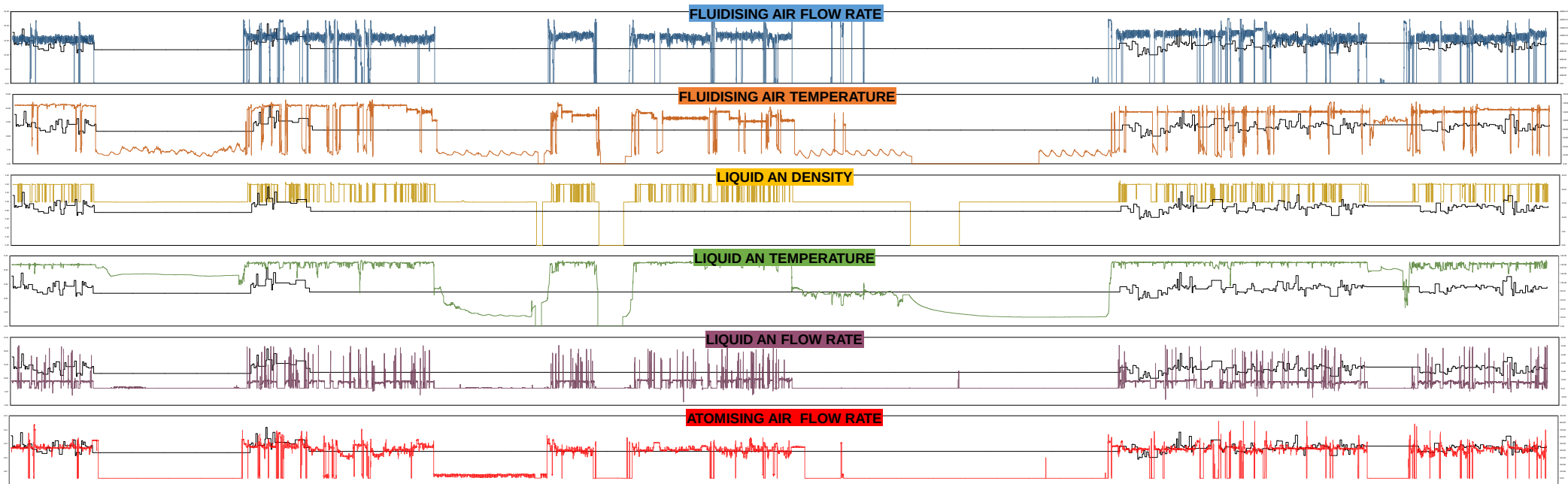


Figure C.2-1: Phase 1 – original data graphs

## C.2.2 Filtered data graphs

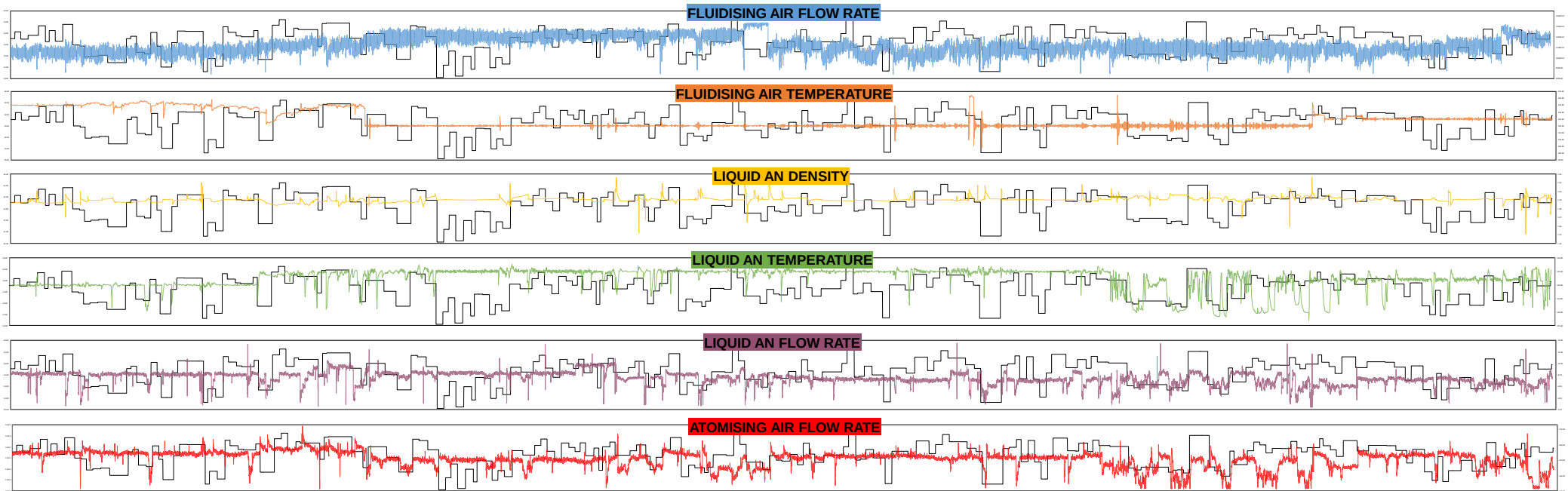


Figure C.2-2: Phase 1 – filtered data graphs

### C.2.3 Discrete data graphs

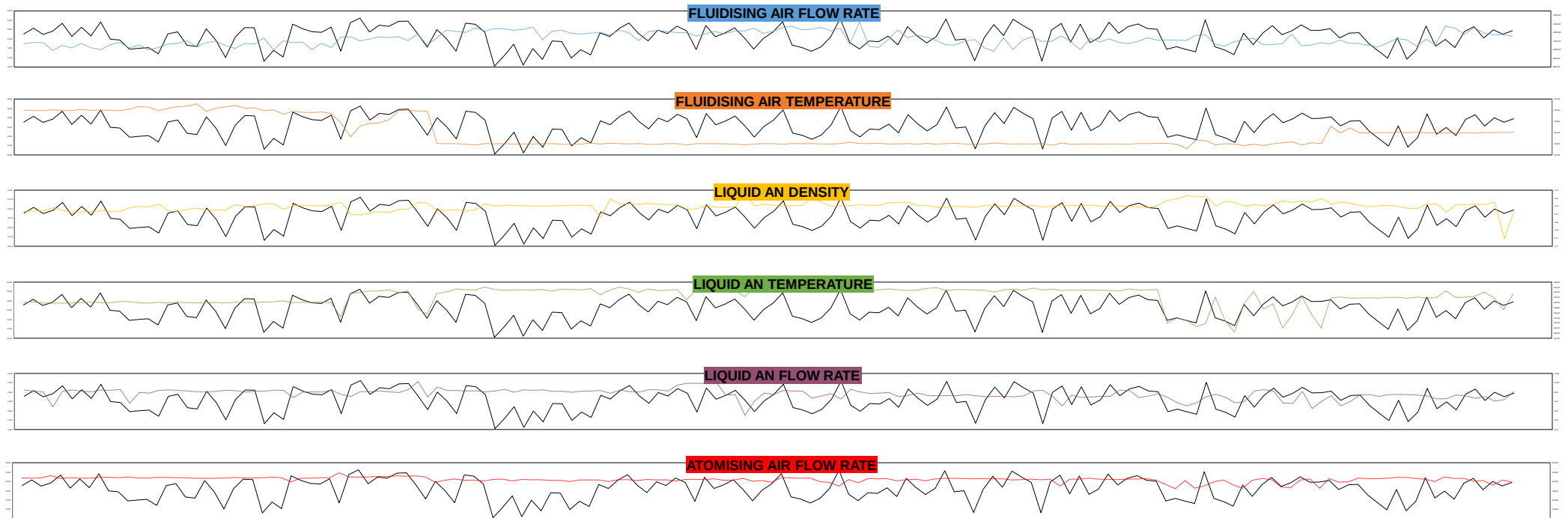


Figure C.2-3: Phase 1 – discrete data graphs

### C.2.4 Constant-time data graphs

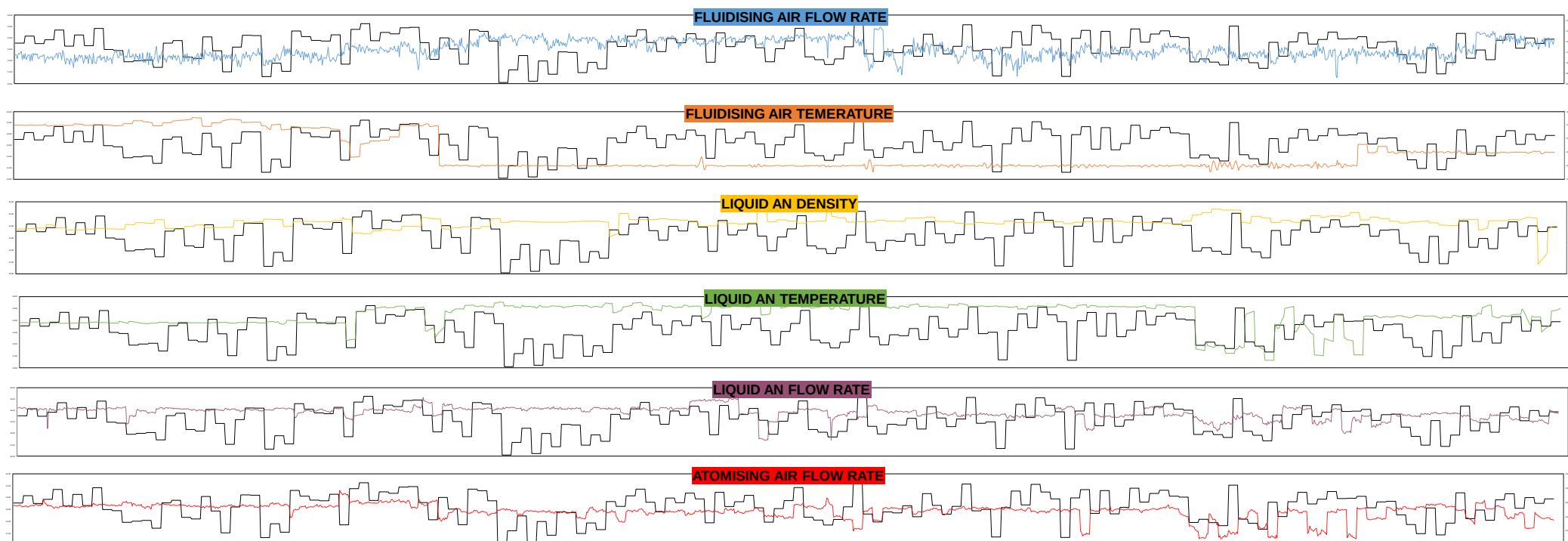


Figure C.2-4: Phase 1 – Constant-time data graphs

### C.3 Regression plots

#### C.3.1 Filtered data using Levenberg-Marquardt training algorithm

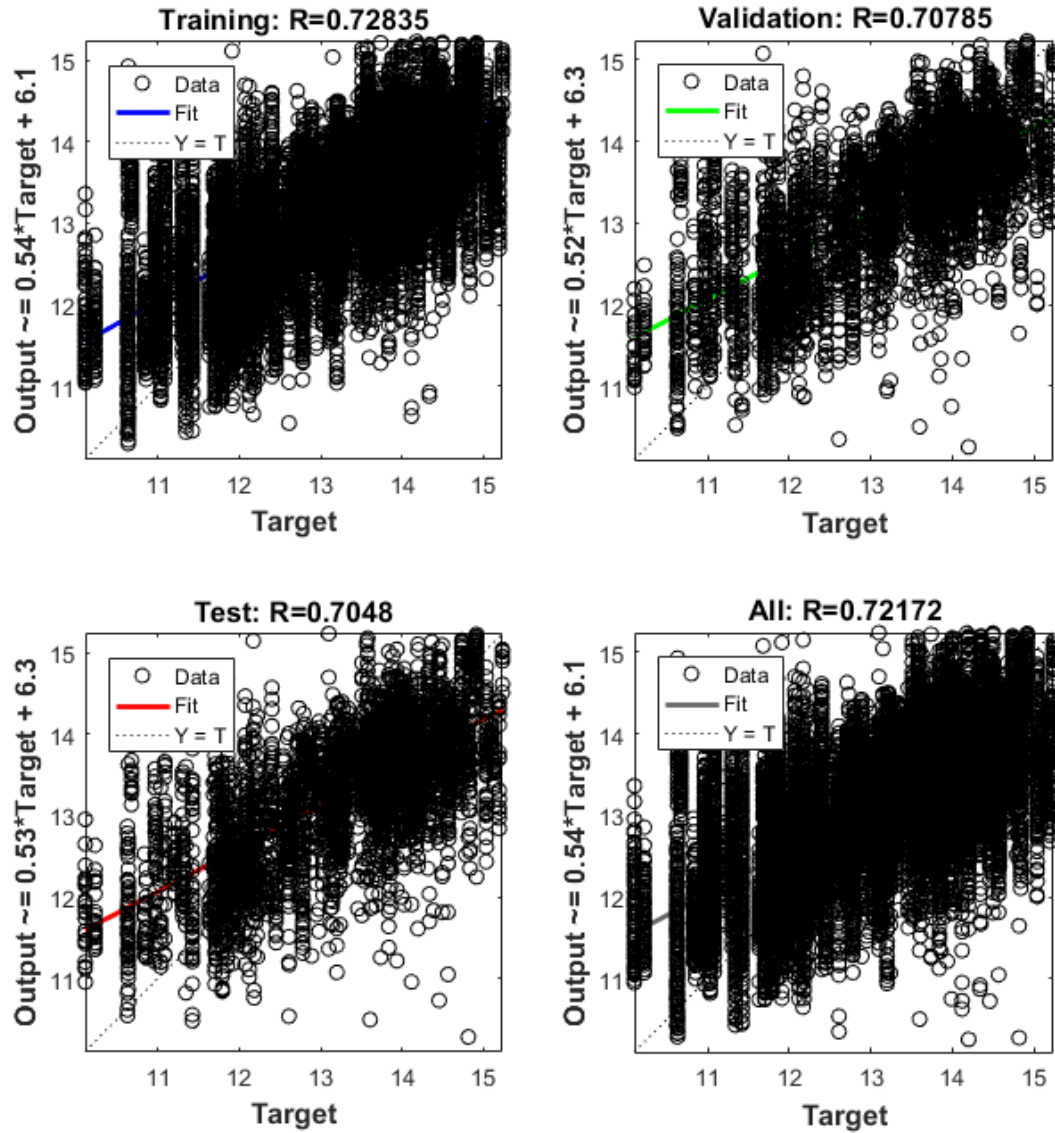


Figure C.3-1: Regression plots of the training of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data

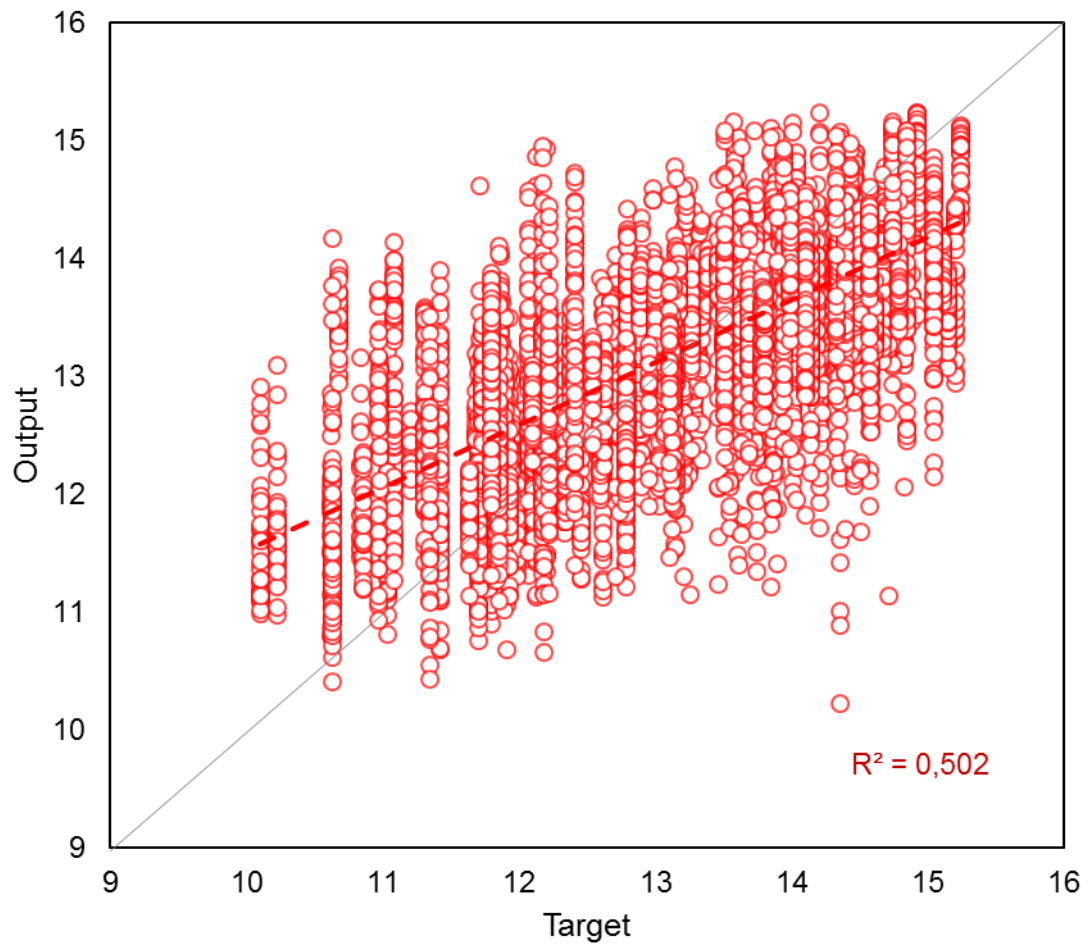


Figure C.3-2: Regression plot of the simulation of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data

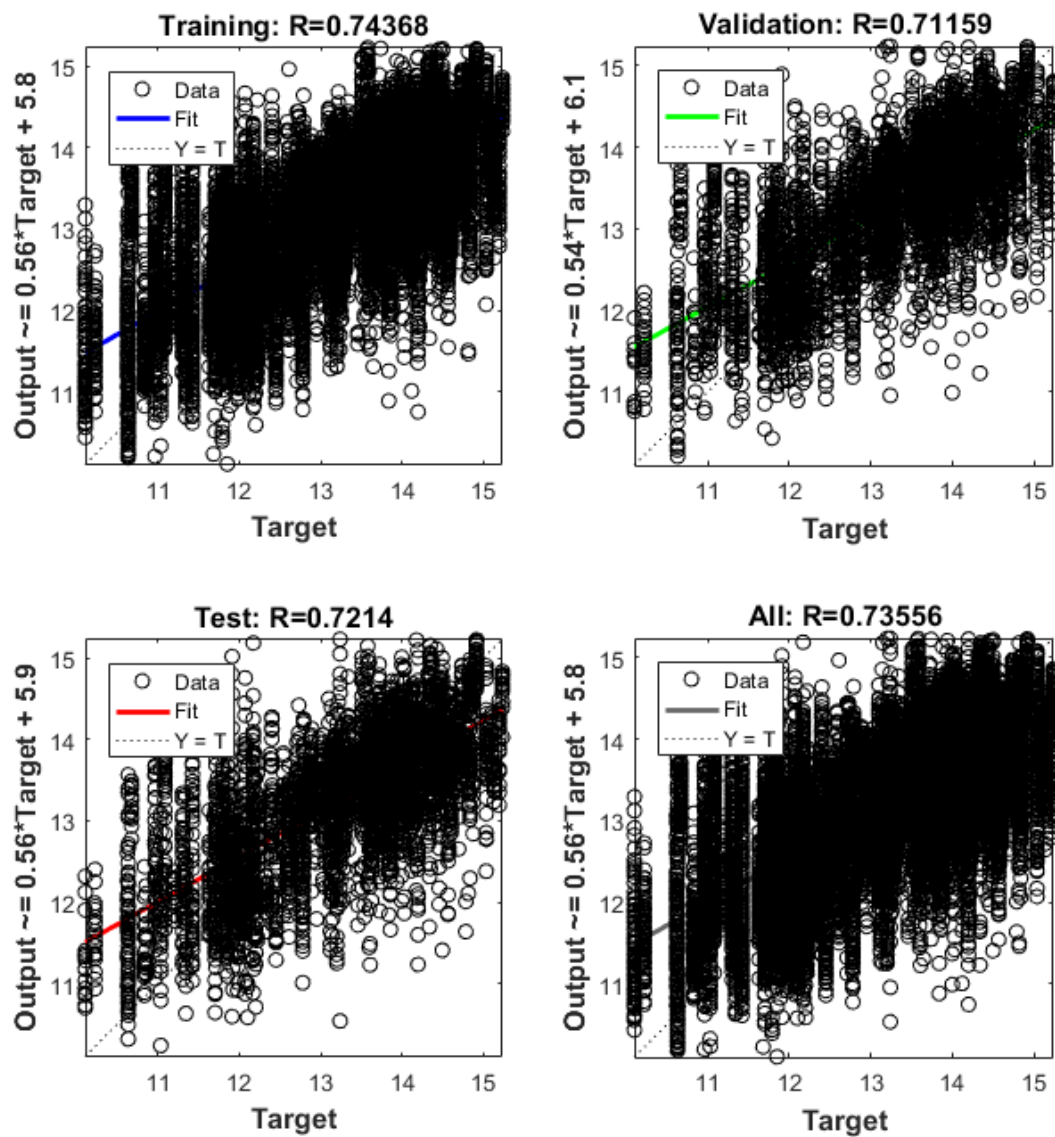


Figure C.3-3: Regression plots of the training of a 72-neuron neural network trained with the Levenberg-Marquardt algorithm using filtered data

## C.3.2 Filtered data using Bayesian regularisation training algorithm

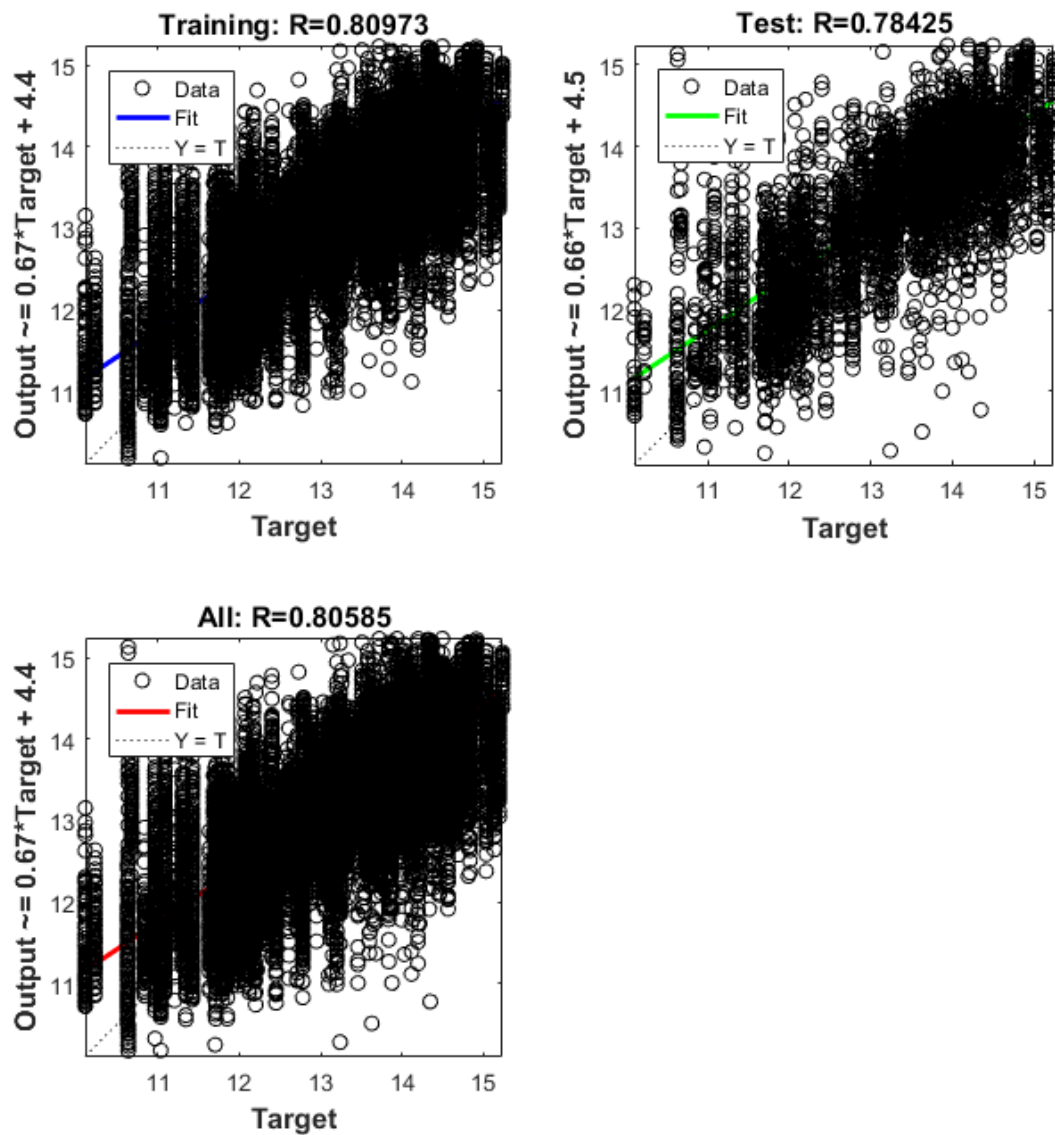


Figure C.3-4: Regression plots of the training of a 125-neuron neural network trained with the Bayesian regularisation algorithm using filtered data

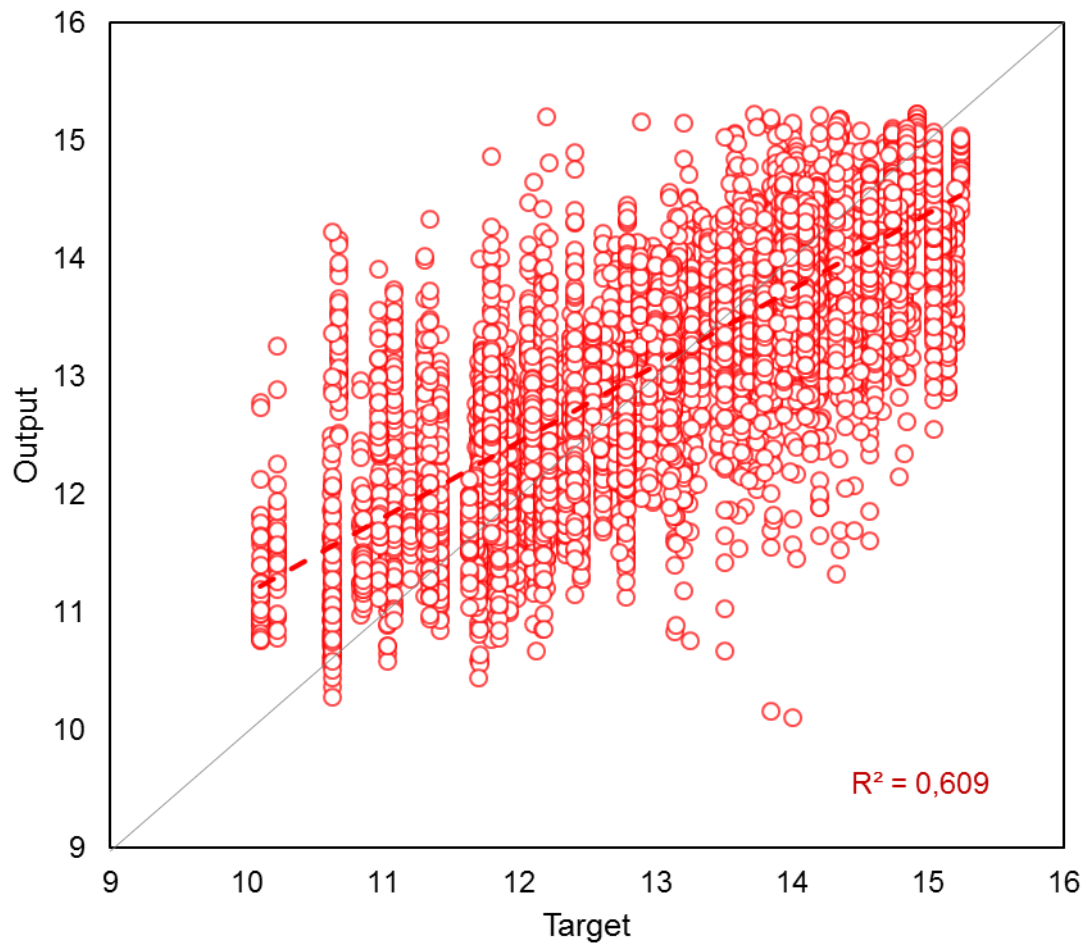


Figure C.3-5: Regression plot of the simulation of a 125-neuron neural network trained with the Bayesian regularisation algorithm using filtered data

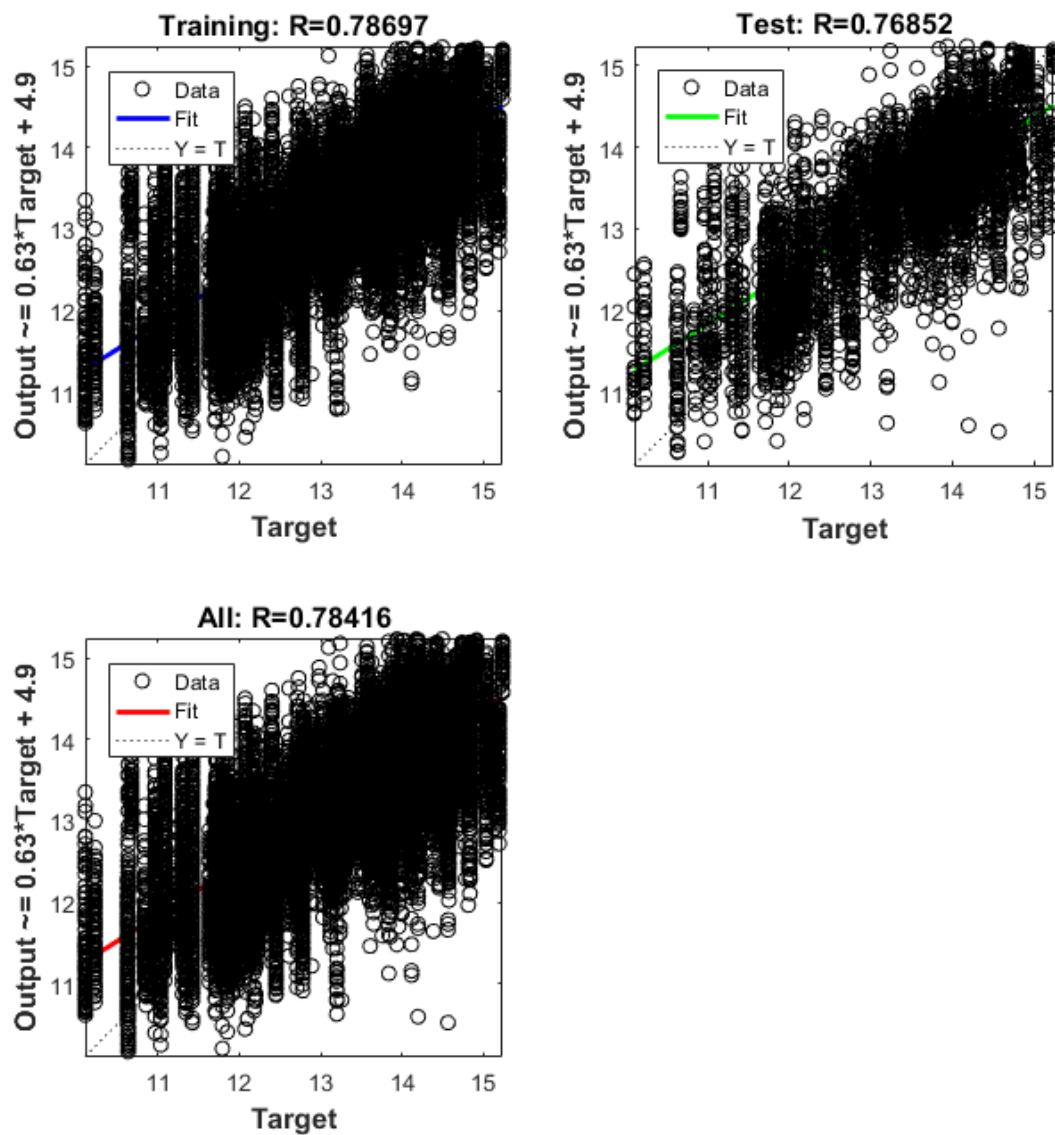


Figure C.3-6: Regression plots of the training of a 113-neuron neural network trained with the Bayesian regularisation algorithm using filtered data

## C.3.3 Discrete data using Levenberg-Marquardt training algorithm

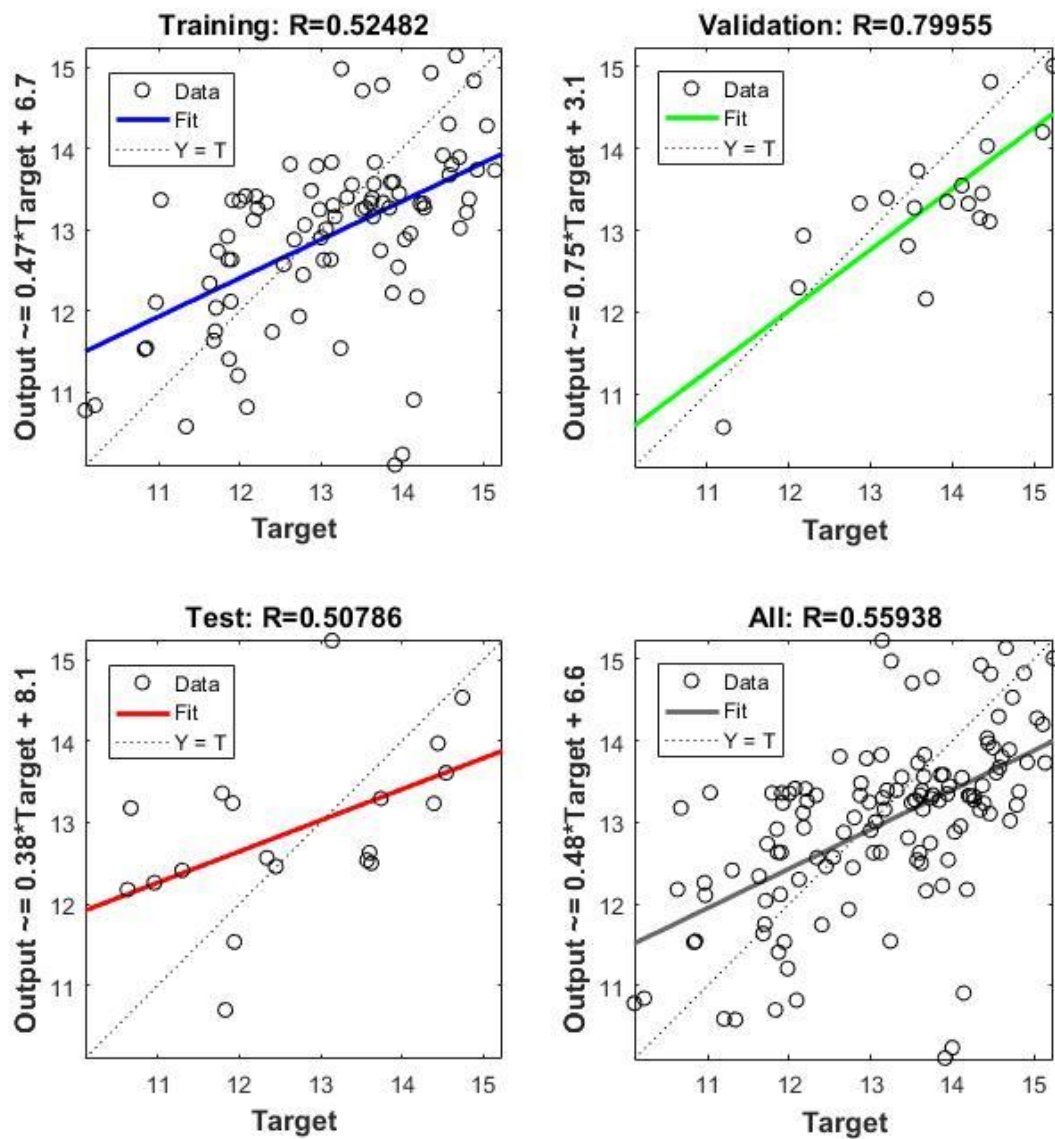


Figure C.3-7: Regression plots of the training of a 100-neuron neural network trained with the Levenberg-Marquardt algorithm using discrete data

## C.3.4 Discrete data using Bayesian regularisation training algorithm

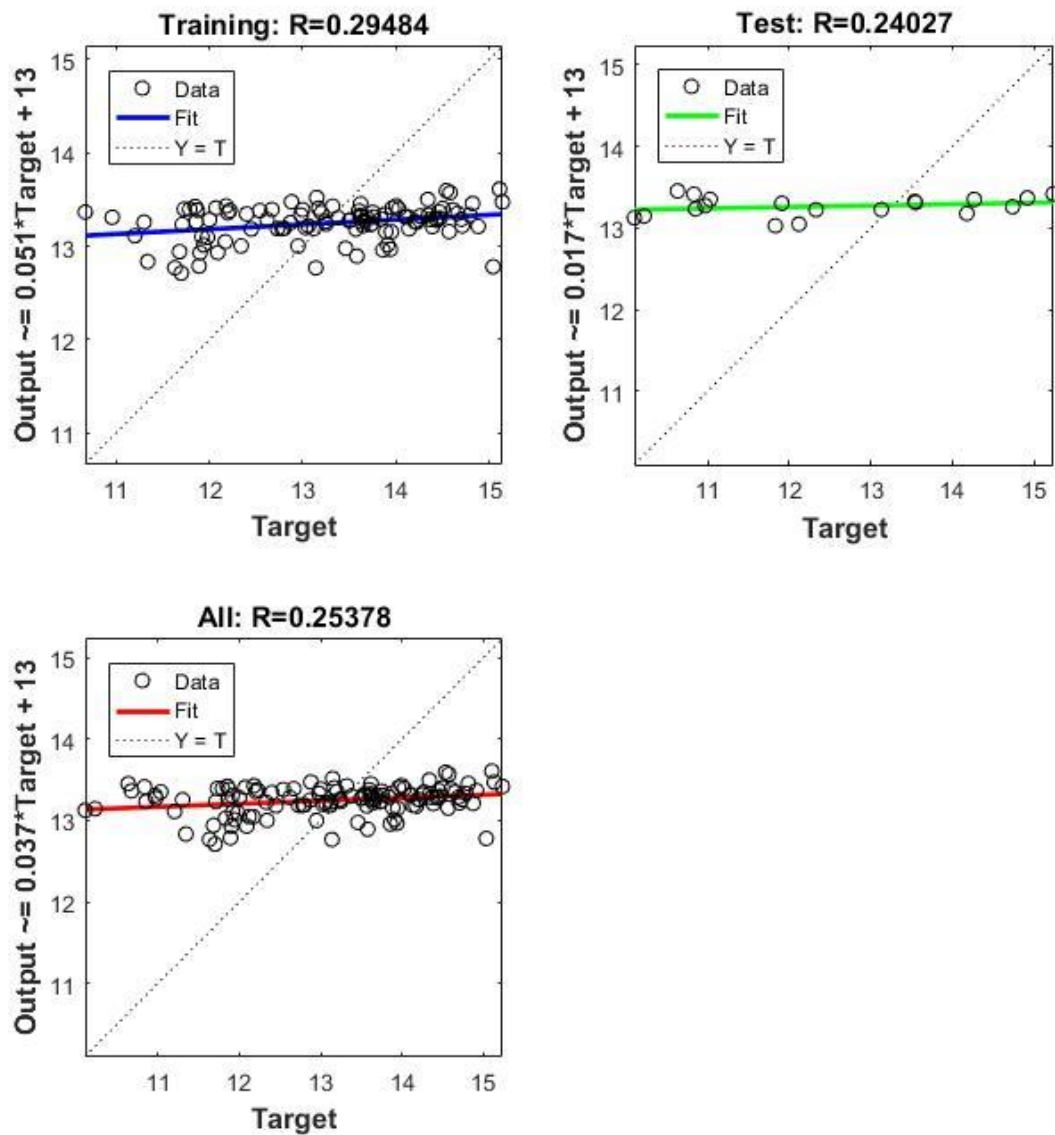


Figure C.3-8: Regression plots of the training of a 100-neuron neural network trained with the Bayesian regularisation algorithm using discrete data

## C.3.5 Constant-time data using Levenberg-Marquardt training algorithm

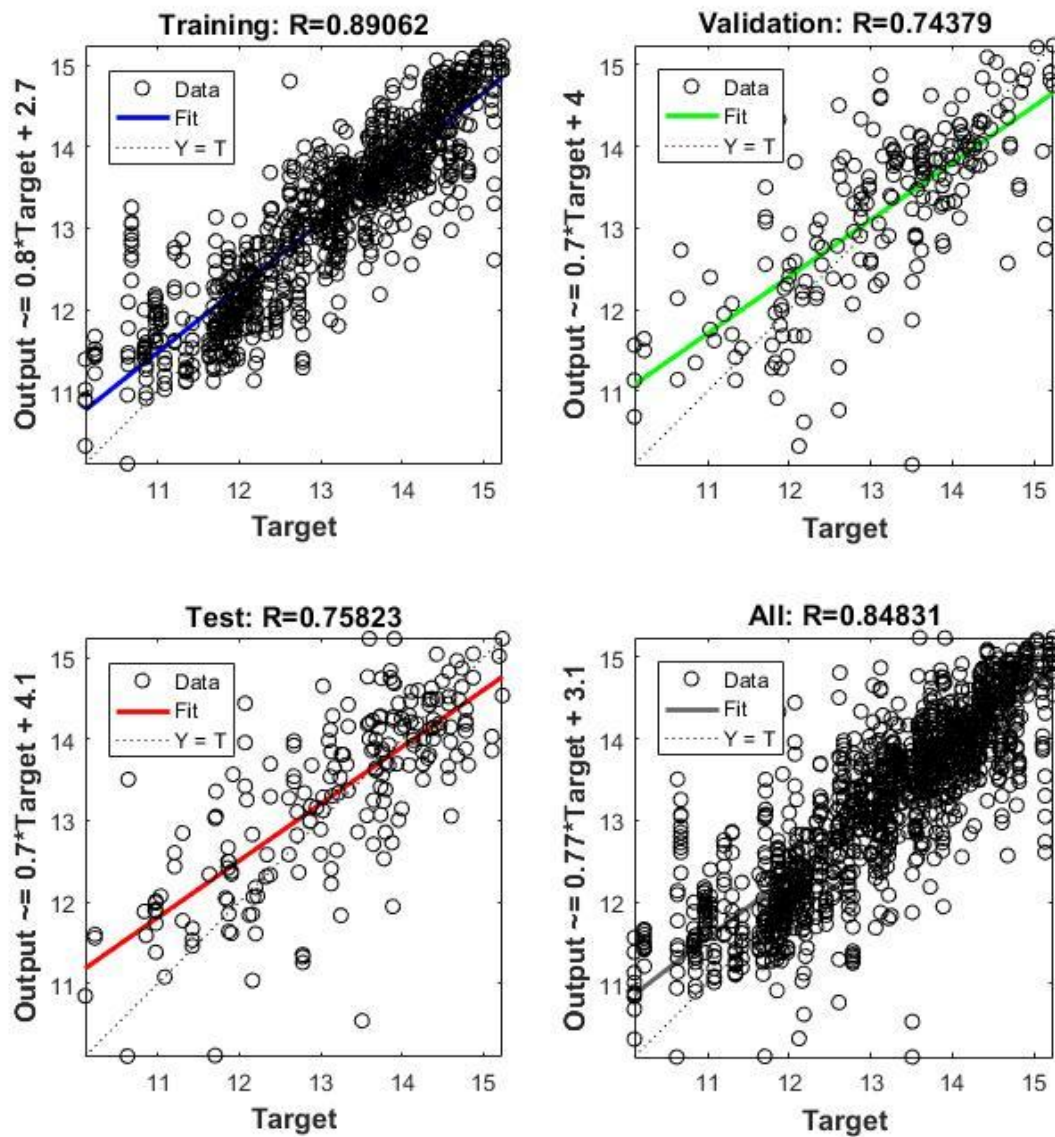


Figure C.3-9: Regression plots of the training of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data

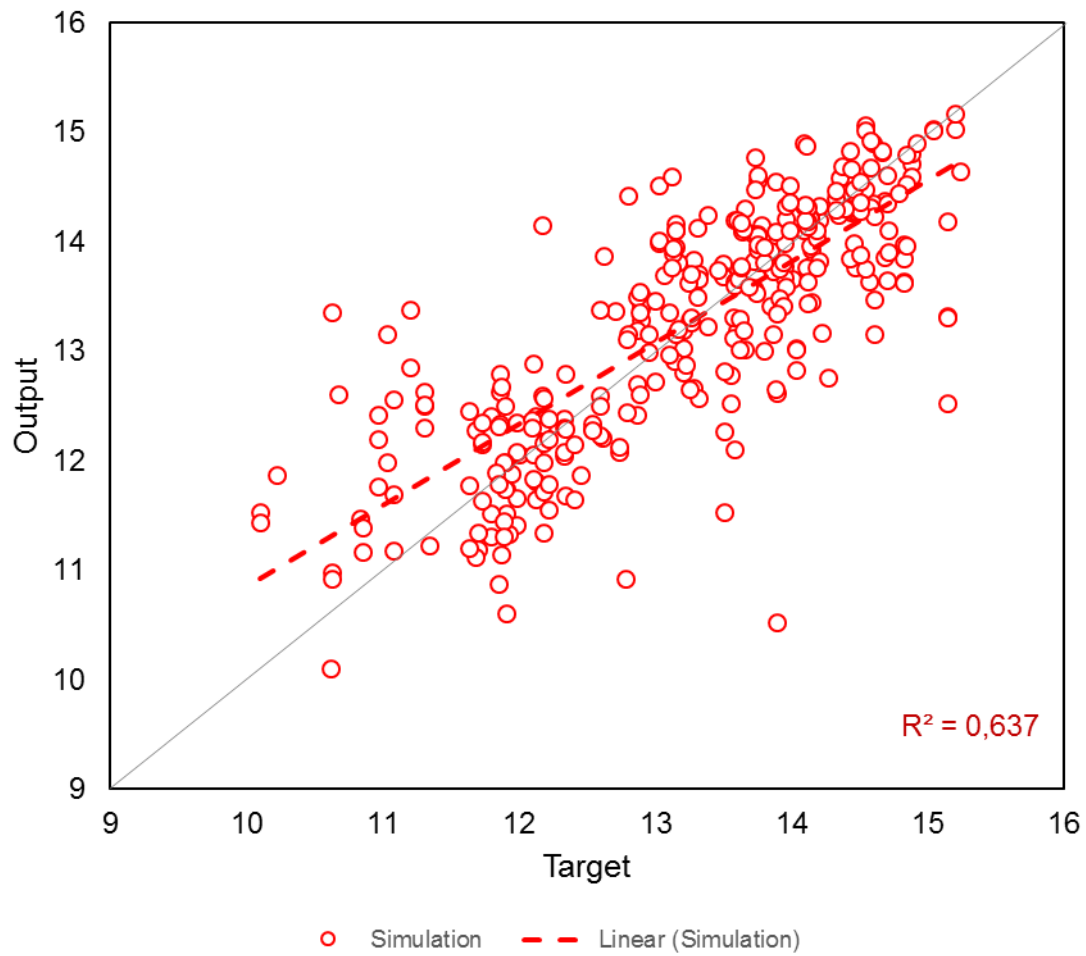


Figure C.3-10: Regression plot of the simulation of a 75-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data

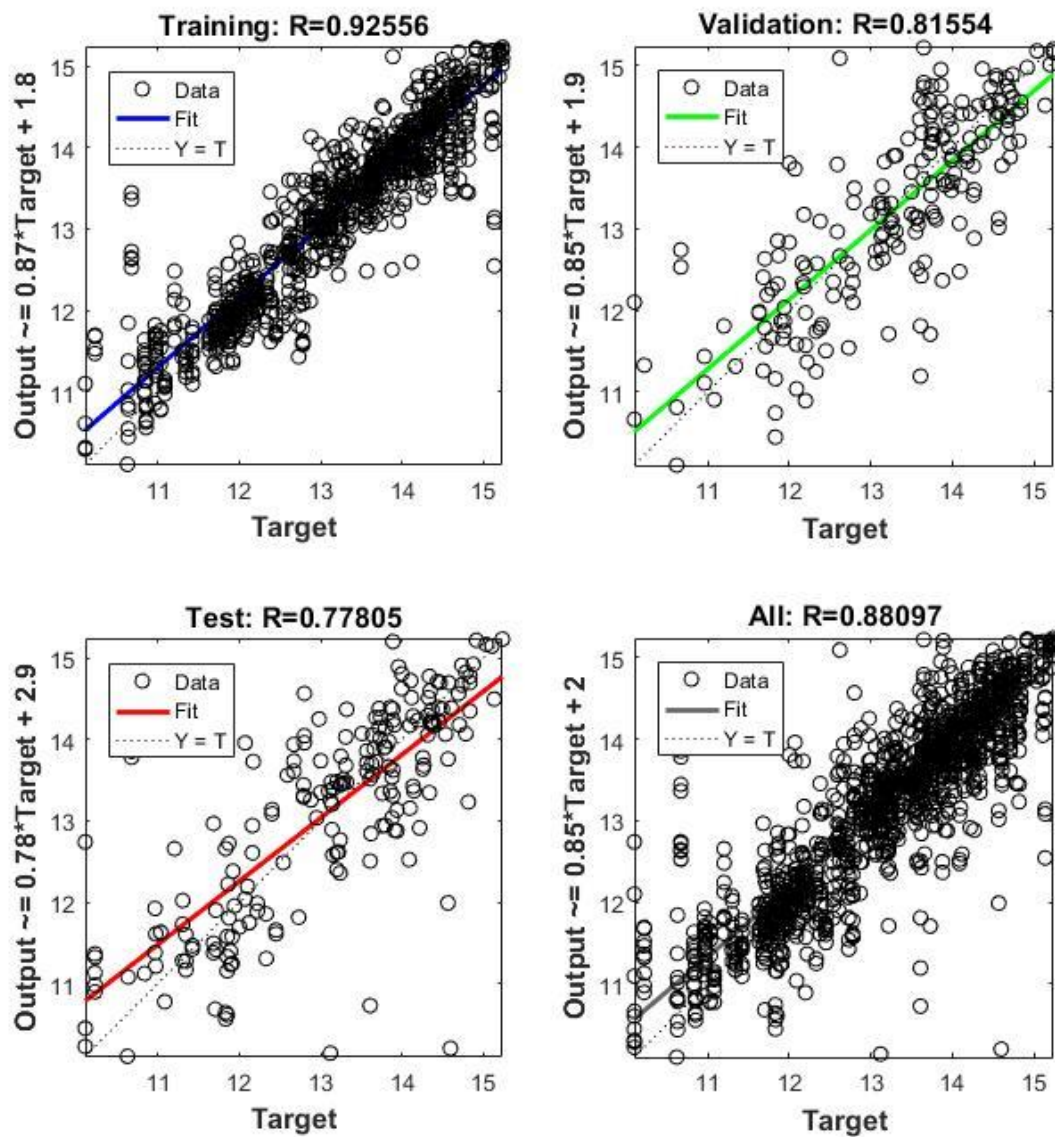


Figure C.3-11: Regression plots of the training of a 69-neuron neural network trained with the Levenberg-Marquardt algorithm using 10-minute constant-time data

## C.3.6 Constant-time data using Bayesian regularisation training algorithm

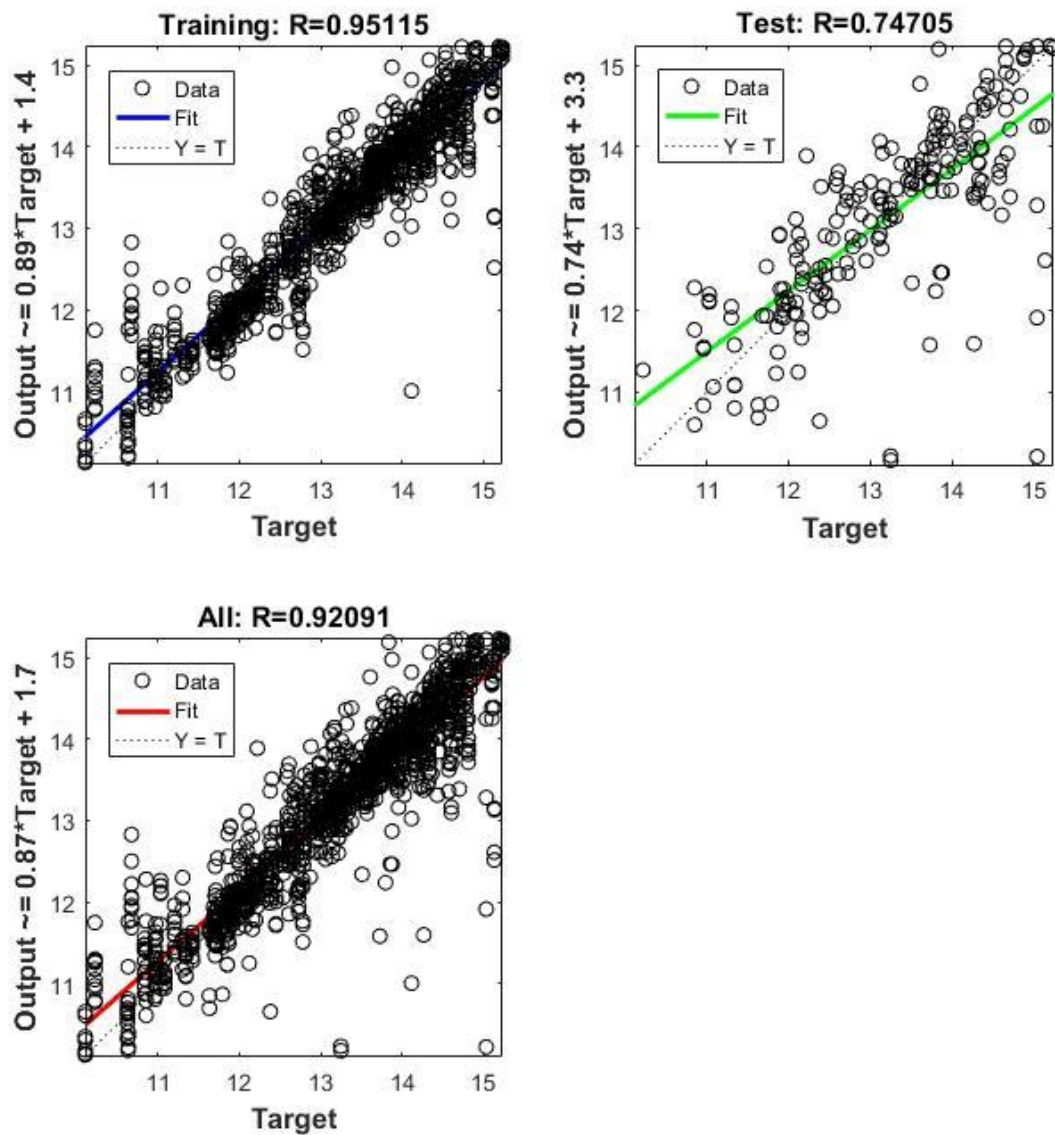


Figure C.3-12: Regression plots of the training of a 75-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data

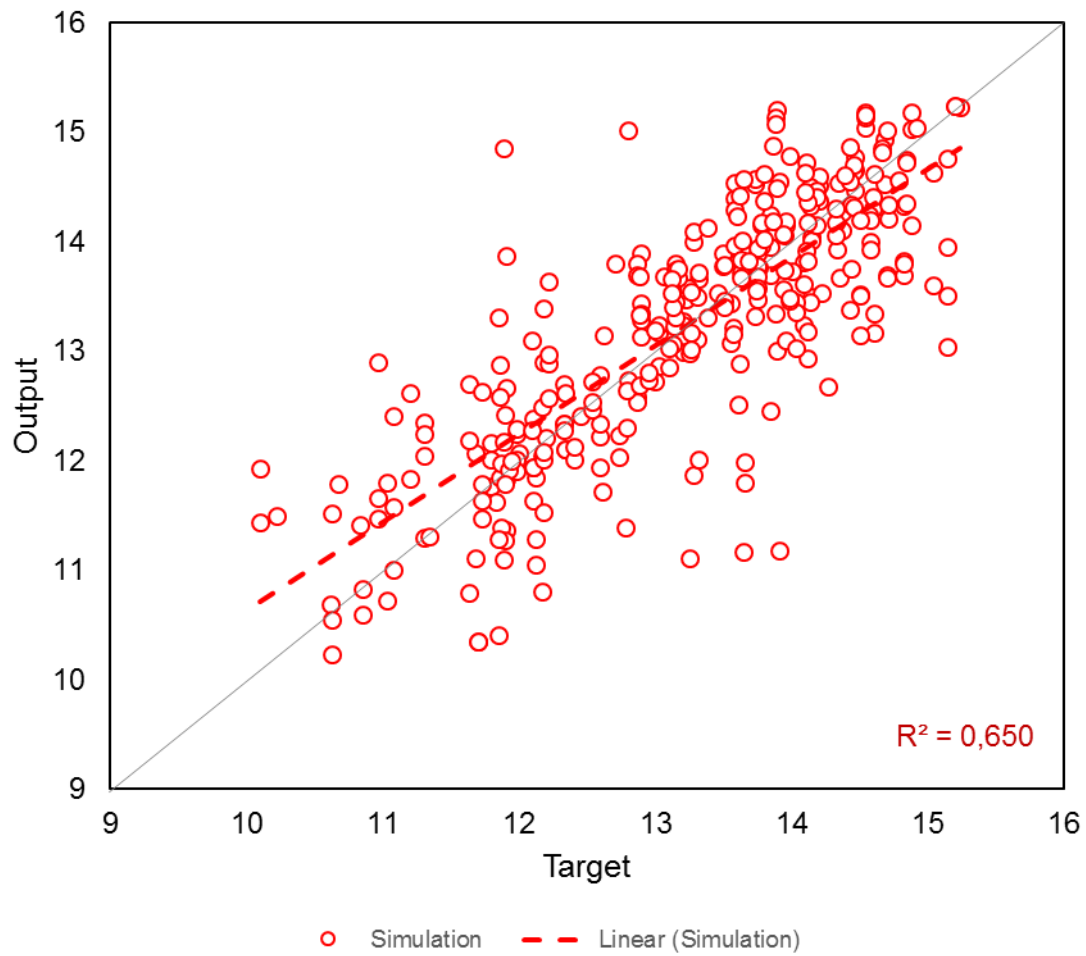


Figure C.3-13: Regression plot of the simulation of a 75-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data

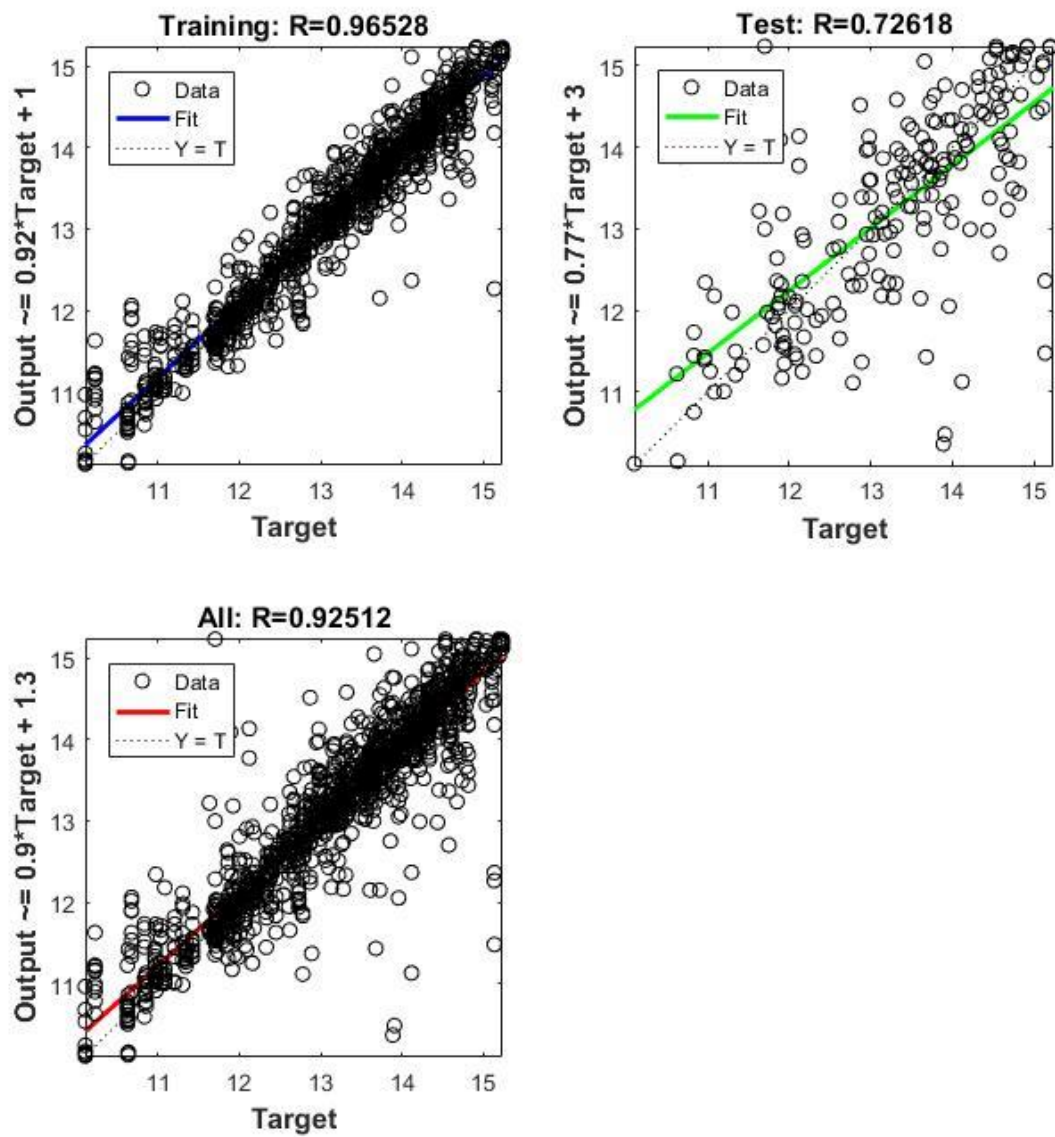


Figure C.3-14: Regression plots of the training of a 69-neuron neural network trained with the Bayesian regularisation algorithm using 10-minute constant-time data

## Appendix D: Experimental calculations and procedures

### D.1 Phase 2 porosity calculations

#### D.1.1 Optical porosity measurement and oil absorption results

Table D.1-1: Optical porosity measurement and oil absorption values for phase 2

| Sample number | Oil absorption [%] | Optical porosity [%] |
|---------------|--------------------|----------------------|
| 1             | 9.0%               | 9.3%                 |
| 2             | 10.4%              | 11.8%                |
| 3             | 5.0%               | 7.0%                 |
| 4             | 9.9%               | 6.7%                 |
| 5             | 8.4%               | 9.5%                 |
| 6             | 7.9%               | 9.4%                 |
| 7             | 16.9%              | 11.8%                |
| 8             | 10.8%              | 12.4%                |
| 9             | 10.0%              | 10.4%                |
| 10            | 8.4%               | 7.1%                 |
| 11            | 9.3%               | 8.7%                 |
| 12            | 9.5%               | 7.9%                 |
| 13            | 8.7%               | 7.6%                 |
| 14            | 8.6%               | 10.7%                |
| 15            | 9.3%               | 9.9%                 |
| 16            | 11.2%              | 13.1%                |
| 17            | 9.5%               | 13.8%                |
| 18            | 10.3%              | 11.9%                |
| 19            | 8.7%               | 10.1%                |
| 20            | 8.2%               | 11.6%                |

**D.1.2 Repeatability calculations on optical porosity measurement method****Table D.1-2: Repeatability on optical porosity measurement method on 05/09 17:10 sample**

| Sample      | Sample number | $d_{50}$<br>[mm] | $V_{\text{bulk.p}}$<br>[m <sup>3</sup> ] | Particle count | Mean roundness | $V_{\text{bulk.T}}$<br>[m <sup>3</sup> ] | Mass [g] | Mass [kg] | $\rho_{\text{bulk}}$<br>[kg/m <sup>3</sup> ] | Porosity [%] |
|-------------|---------------|------------------|------------------------------------------|----------------|----------------|------------------------------------------|----------|-----------|----------------------------------------------|--------------|
| 05/09 17:10 | 1             | 2.824            | 1.18E-08                                 | 9064           | 0.911          | 9.74E-05                                 | 157.2    | 0.1572    | 1614                                         | 6.4%         |
|             | 2             | 2.821            | 1.18E-08                                 | 9119           | 0.910          | 9.75E-05                                 | 157.2    | 0.1572    | 1612                                         | 6.6%         |
|             | 3             | 2.826            | 1.18E-08                                 | 9136           | 0.910          | 9.82E-05                                 | 157.2    | 0.1572    | 1600                                         | 7.2%         |
| 05/09 17:10 | AVG           | 2.824            | 1.18E-08                                 | 9106           | 0.910          | 9.77E-05                                 | 157.2    | 0.1572    | 1609                                         | 6.7%         |
| 05/09 17:10 | STD           |                  |                                          |                |                |                                          |          |           |                                              | 0.4%         |
| 05/09 17:10 | ε             |                  |                                          |                |                |                                          |          |           |                                              | 1.1%         |
| 05/09 17:10 | % Error       |                  |                                          |                |                |                                          |          |           |                                              | 16.2%        |

Table D.1-3: Repeatability on optical porosity measurement method on 05/10 18:00 sample

| Sample      | Sample number | $d_{50}$<br>[mm] | $V_{\text{bulk.p}}$<br>[m <sup>3</sup> ] | Particle count | Mean roundness | $V_{\text{bulk.T}}$<br>[m <sup>3</sup> ] | Mass [g] | Mass [kg] | $\rho_{\text{bulk}}$<br>[kg/m <sup>3</sup> ] | Porosity [%] |
|-------------|---------------|------------------|------------------------------------------|----------------|----------------|------------------------------------------|----------|-----------|----------------------------------------------|--------------|
| 05/10 18:00 | 1             | 2.771            | 1.11E-08                                 | 9082           | 0.912          | 9.23E-05                                 | 145.8    | 0.1458    | 1580                                         | 8.4%         |
|             | 2             | 2.769            | 1.11E-08                                 | 9111           | 0.912          | 9.24E-05                                 | 145.8    | 0.1458    | 1578                                         | 8.5%         |
|             | 3             | 2.773            | 1.12E-08                                 | 9148           | 0.912          | 9.31E-05                                 | 145.8    | 0.1458    | 1565                                         | 9.3%         |
| 05/10 18:00 | AVG           | 2.771            | 1.11E-08                                 | 9114           | 0.912          | 9.26E-05                                 | 145.8    | 0.1458    | 1575                                         | 8.7%         |
| 05/10 18:00 | STD           |                  |                                          |                |                |                                          |          |           |                                              | 0.5%         |
| 05/10 18:00 | ε             |                  |                                          |                |                |                                          |          |           |                                              | 1.2%         |
| 05/10 18:00 | % Error       |                  |                                          |                |                |                                          |          |           |                                              | 13.4%        |

Table D.1-4: Repeatability on optical porosity measurement method on 05/11 05:50 sample

| Sample     | Sample number | $d_{50}$<br>[mm] | $V_{\text{bulk.p}}$<br>[m <sup>3</sup> ] | Particle count | Mean roundness | $V_{\text{bulk.T}}$<br>[m <sup>3</sup> ] | Mass [g] | Mass [kg] | $\rho_{\text{bulk}}$<br>[kg/m <sup>3</sup> ] | Porosity [%] |
|------------|---------------|------------------|------------------------------------------|----------------|----------------|------------------------------------------|----------|-----------|----------------------------------------------|--------------|
| 05/11 5:50 | 1             | 2.726            | 1.06E-08                                 | 10379          | 0.911          | 1.00E-04                                 | 152.8    | 0.1528    | 1524                                         | 11.7%        |
|            | 2             | 2.726            | 1.06E-08                                 | 10379          | 0.912          | 1.00E-04                                 | 152.8    | 0.1528    | 1522                                         | 11.8%        |
|            | 3             | 2.721            | 1.05E-08                                 | 10433          | 0.911          | 1.00E-04                                 | 152.8    | 0.1528    | 1524                                         | 11.6%        |
| 05/11 5:50 | AVG           | 2.724            | 1.06E-08                                 | 10397          | 0.911          | 1.00E-04                                 | 152.8    | 0.1528    | 1523                                         | 11.7%        |
| 05/11 5:50 | STD           |                  |                                          |                |                |                                          |          |           |                                              | 0.1%         |
| 05/11 5:50 | ε             |                  |                                          |                |                |                                          |          |           |                                              | 0.2%         |
| 05/11 5:50 | % Error       |                  |                                          |                |                |                                          |          |           |                                              | 1.4%         |

**D.1.3 Repeatability calculations on oil absorption****Table D.1-5: Repeatability on oil absorption on 05/09 17:10 sample**

| <b>Sample</b> | <b>Sample number</b> | <b>m<sub>container</sub><br/>[g]</b> | <b>m<sub>granules</sub><br/>[g]</b> | <b>m<sub>after</sub><br/>[g]</b> | <b>m<sub>liq absorbed</sub><br/>[g]</b> | <b>m<sub>liq absorbed</sub><br/>[kg]</b> | <b>m<sub>liq absorbed</sub><br/>[%]</b> |
|---------------|----------------------|--------------------------------------|-------------------------------------|----------------------------------|-----------------------------------------|------------------------------------------|-----------------------------------------|
| 05/09 17:10   | 1                    | 9.412                                | 2.497                               | 12.183                           | 0.274                                   | 0.000274                                 | 9.9%                                    |
|               | 2                    | 9.413                                | 2.508                               | 12.186                           | 0.265                                   | 0.000265                                 | 9.6%                                    |
|               | 3                    | 9.841                                | 2.504                               | 12.609                           | 0.264                                   | 0.000264                                 | 9.5%                                    |
| 05/09 17:10   | AVG                  | 9.555                                | 2.503                               | 12.326                           | 0.268                                   | 0.000268                                 | 0.097                                   |
| 05/09 17:10   | STD                  |                                      |                                     |                                  |                                         |                                          | 0.2%                                    |
| 05/09 17:10   | €                    |                                      |                                     |                                  |                                         |                                          | 0.5%                                    |
| 05/09 17:10   | % Error              |                                      |                                     |                                  |                                         |                                          | 5.1%                                    |

Table D.1-6: Repeatability on oil absorption on 05/10 18:00 sample

| Sample      | Sample number | m <sub>container</sub><br>[g] | m <sub>granules</sub><br>[g] | m <sub>after</sub><br>[g] | m <sub>liq absorbed</sub><br>[g] | m <sub>liq absorbed</sub><br>[kg] | m <sub>liq absorbed</sub><br>[%] |
|-------------|---------------|-------------------------------|------------------------------|---------------------------|----------------------------------|-----------------------------------|----------------------------------|
| 05/10 18:00 | 1             | 9.026                         | 2.497                        | 11.780                    | 0.257                            | 0.000257                          | 9.3%                             |
|             | 2             | 9.438                         | 2.499                        | 12.231                    | 0.294                            | 0.000294                          | 10.5%                            |
|             | 3             | 9.848                         | 2.509                        | 12.626                    | 0.269                            | 0.000269                          | 9.7%                             |
| 05/10 18:00 | AVG           | 9.437                         | 2.502                        | 12.212                    | 0.273                            | 0.000273                          | 0.098                            |
| 05/10 18:00 | STD           |                               |                              |                           |                                  |                                   | 0.6%                             |
| 05/10 18:00 | €             |                               |                              |                           |                                  |                                   | 1.5%                             |
| 05/10 18:00 | % Error       |                               |                              |                           |                                  |                                   | 15.5%                            |

Table D.1-7: Repeatability on oil absorption on 05/11 18:00 sample

| Sample     | Sample number | m <sub>container</sub><br>[g] | m <sub>granules</sub><br>[g] | m <sub>after</sub><br>[g] | m <sub>liq absorbed</sub><br>[g] | m <sub>liq absorbed</sub><br>[kg] | m <sub>liq absorbed</sub><br>[%] |
|------------|---------------|-------------------------------|------------------------------|---------------------------|----------------------------------|-----------------------------------|----------------------------------|
| 05/11 5:50 | 1             | 9.049                         | 2.498                        | 11.769                    | 0.222                            | 0.000222                          | 8.2%                             |
|            | 2             | 9.718                         | 2.504                        | 12.419                    | 0.197                            | 0.000197                          | 7.3%                             |
|            | 3             | 9.660                         | 2.504                        | 12.378                    | 0.214                            | 0.000214                          | 7.9%                             |
| 05/11 5:50 | AVG           | 9.476                         | 2.502                        | 12.189                    | 0.211                            | 0.000211                          | 0.078                            |
| 05/11 5:50 | STD           |                               |                              |                           |                                  |                                   | 0.4%                             |
| 05/11 5:50 | €             |                               |                              |                           |                                  |                                   | 1.1%                             |
| 05/11 5:50 | % Error       |                               |                              |                           |                                  |                                   | 14.1%                            |

## Appendix E: Phase 2 data processing

### E.1 Phase 2 porosity data graphs

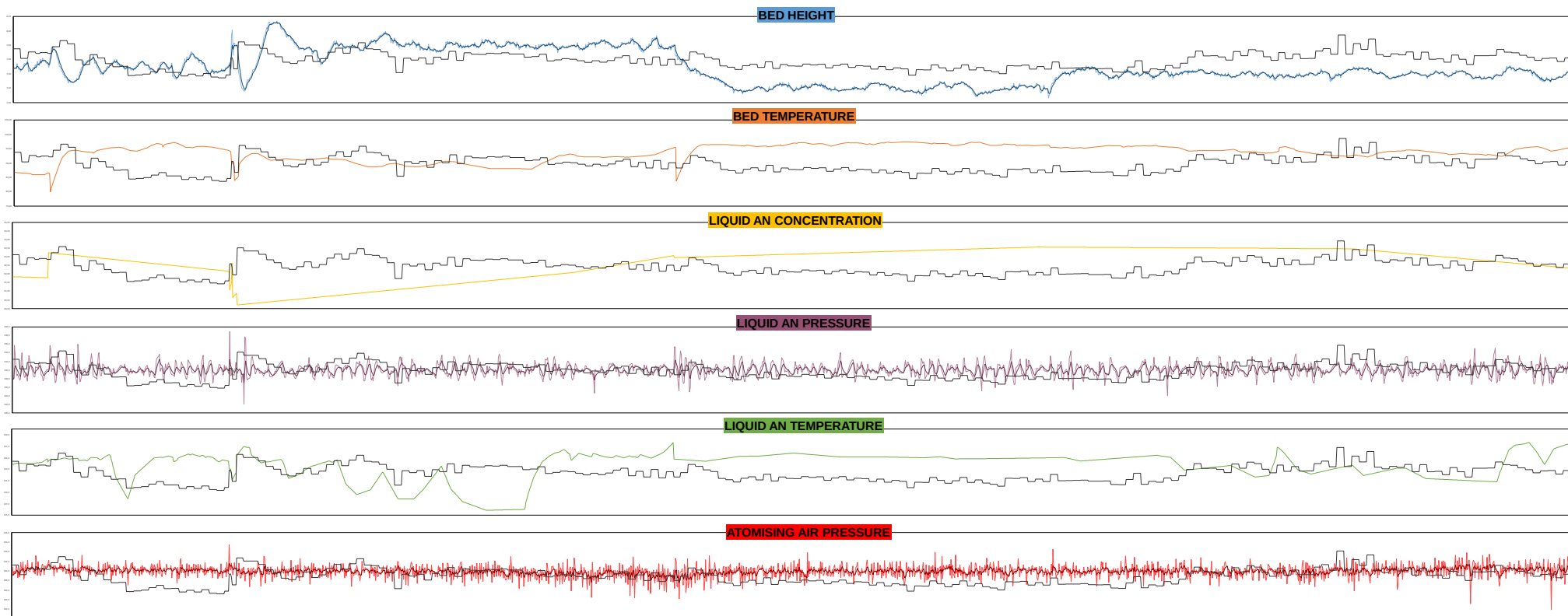


Figure E.1-1: Phase 2 sensor inputs and porosity data

## E.2 Regression plots for phase 2 porosity estimation

### E.2.1 Bayesian regularisation training algorithm

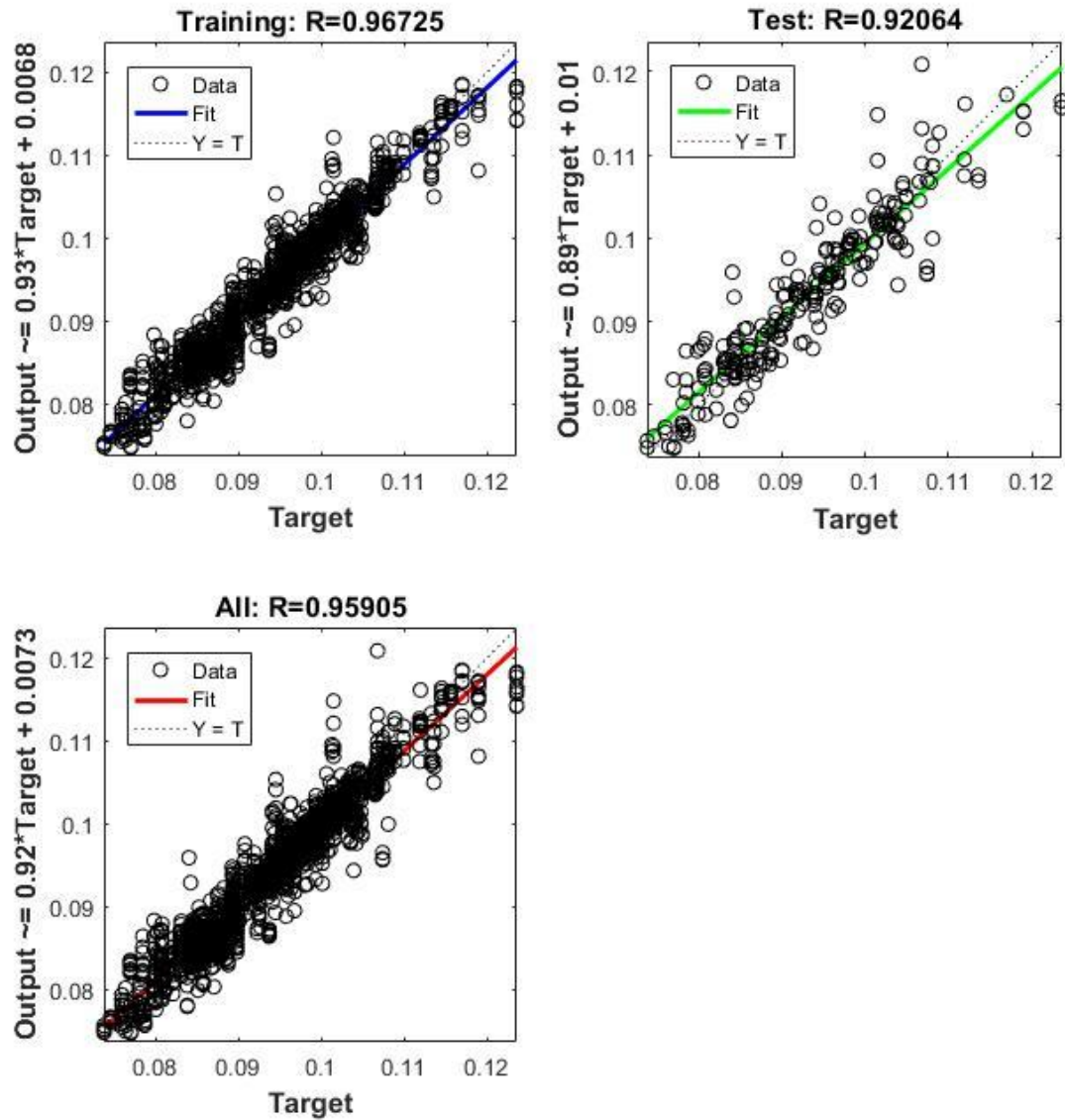


Figure E.2-1: Regression plots of the training of a 50-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data



Figure E.2-2: Regression plot of the simulation of a 50-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data

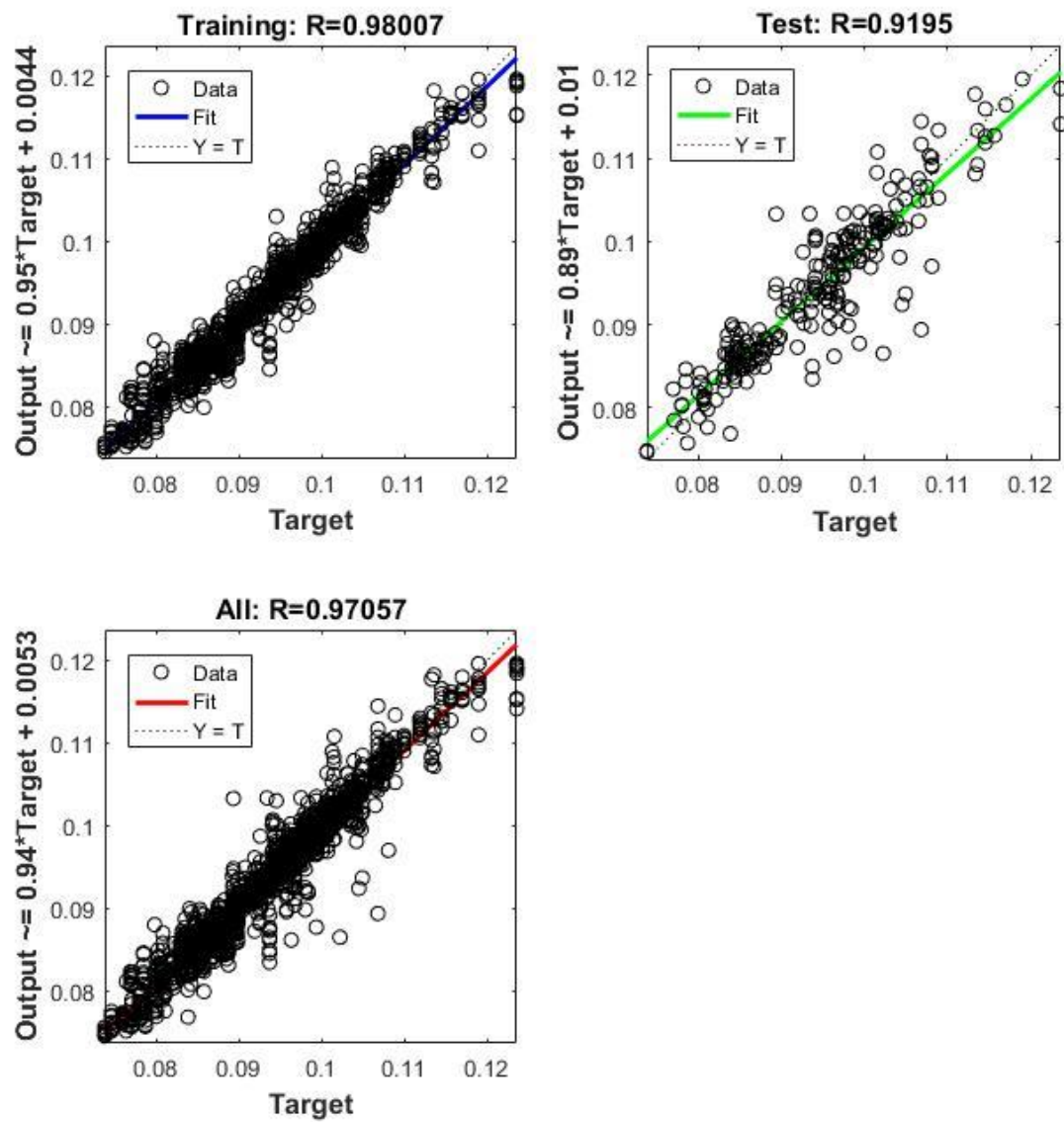


Figure E.2-3: Regression plots of the training of a 61-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 porosity data

## E.2.2 Activation function comparison

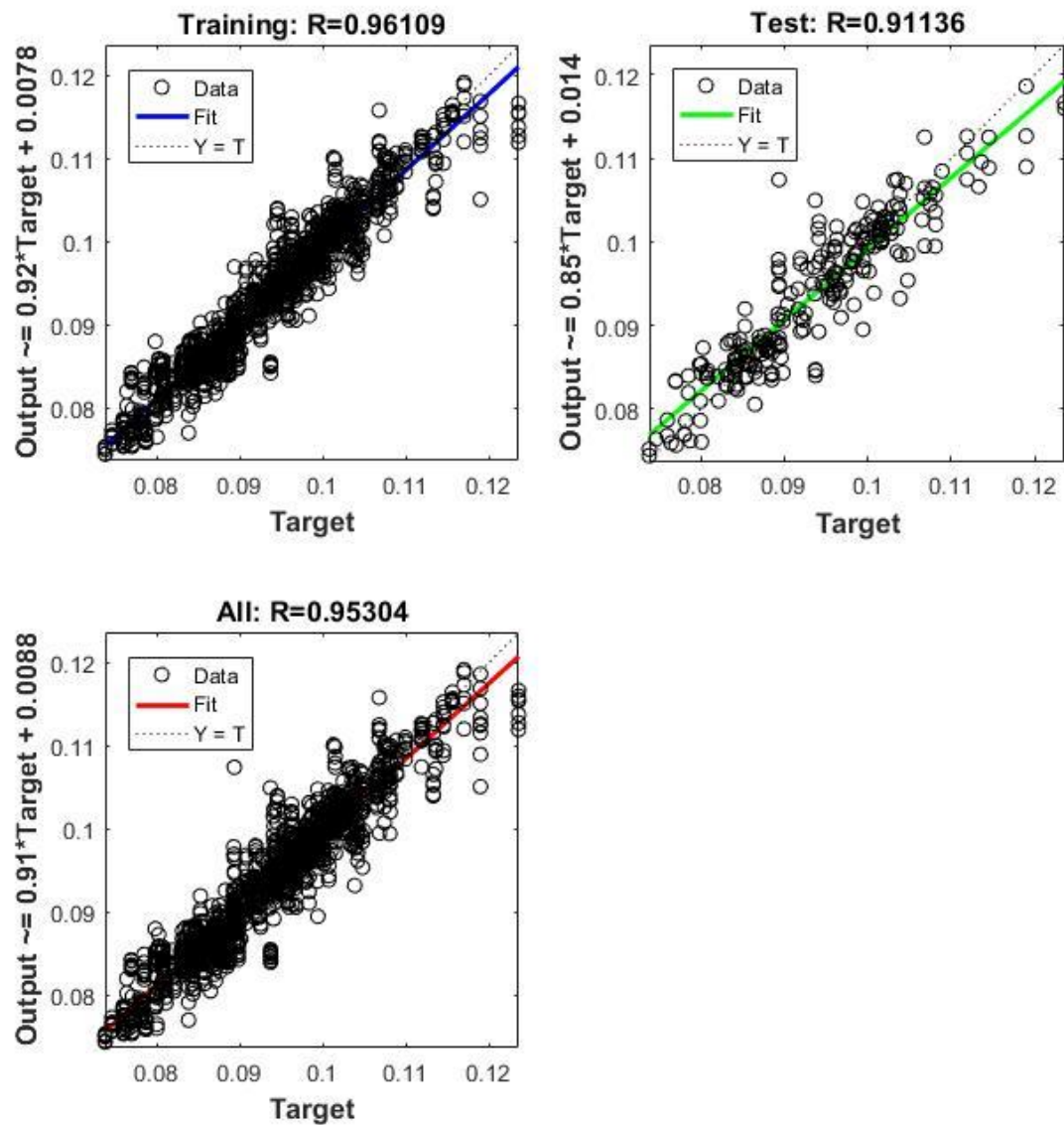


Figure E.2-4: Regression plots of the training of a 61-neuron neural network using phase 2 porosity data and a sigmoidal activation function

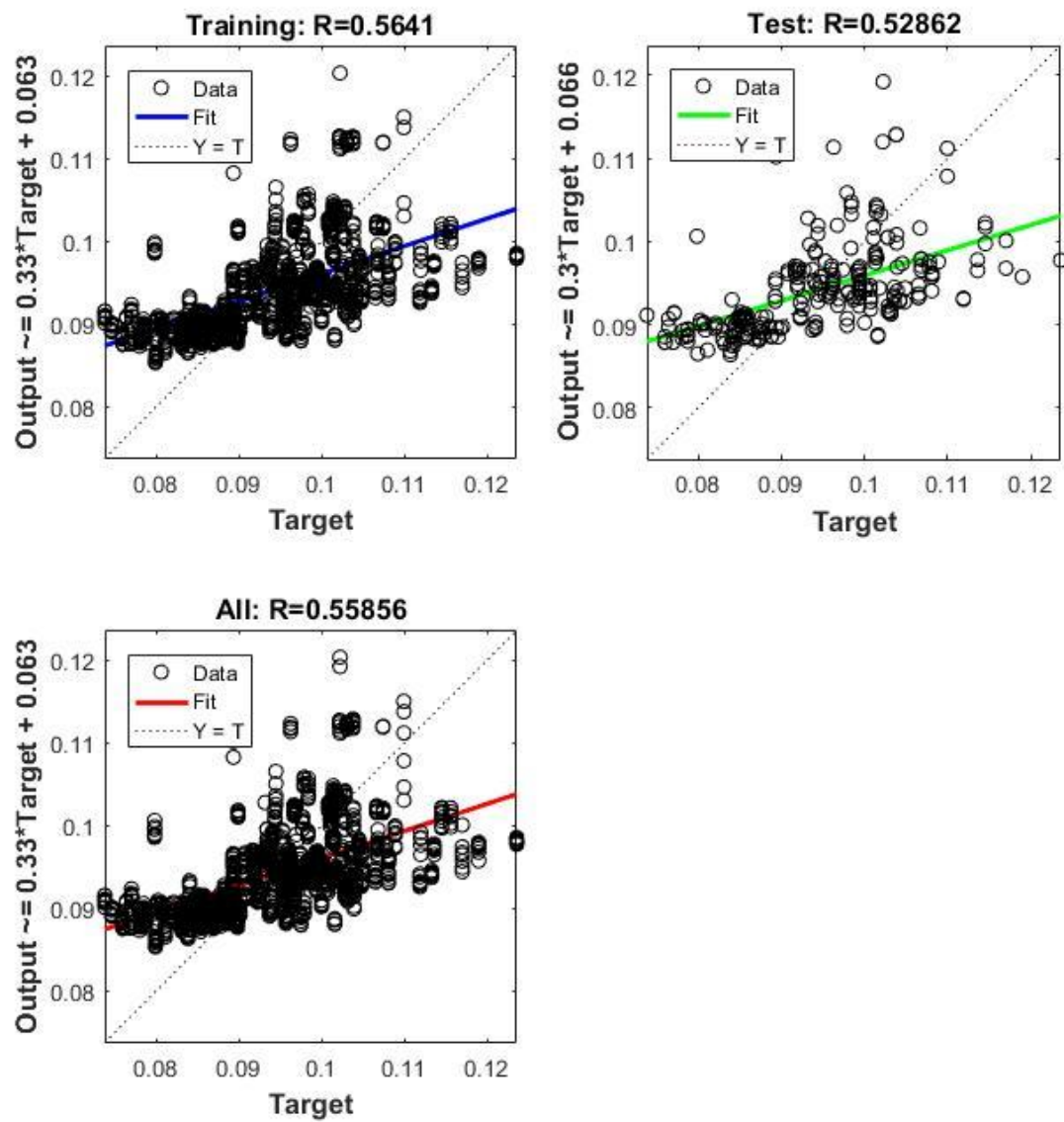


Figure E.2-5: Regression plots of the training of a 61-neuron neural network using phase 2 porosity data and a linear activation function

### E.3 Phase 2 sphericity data graphs

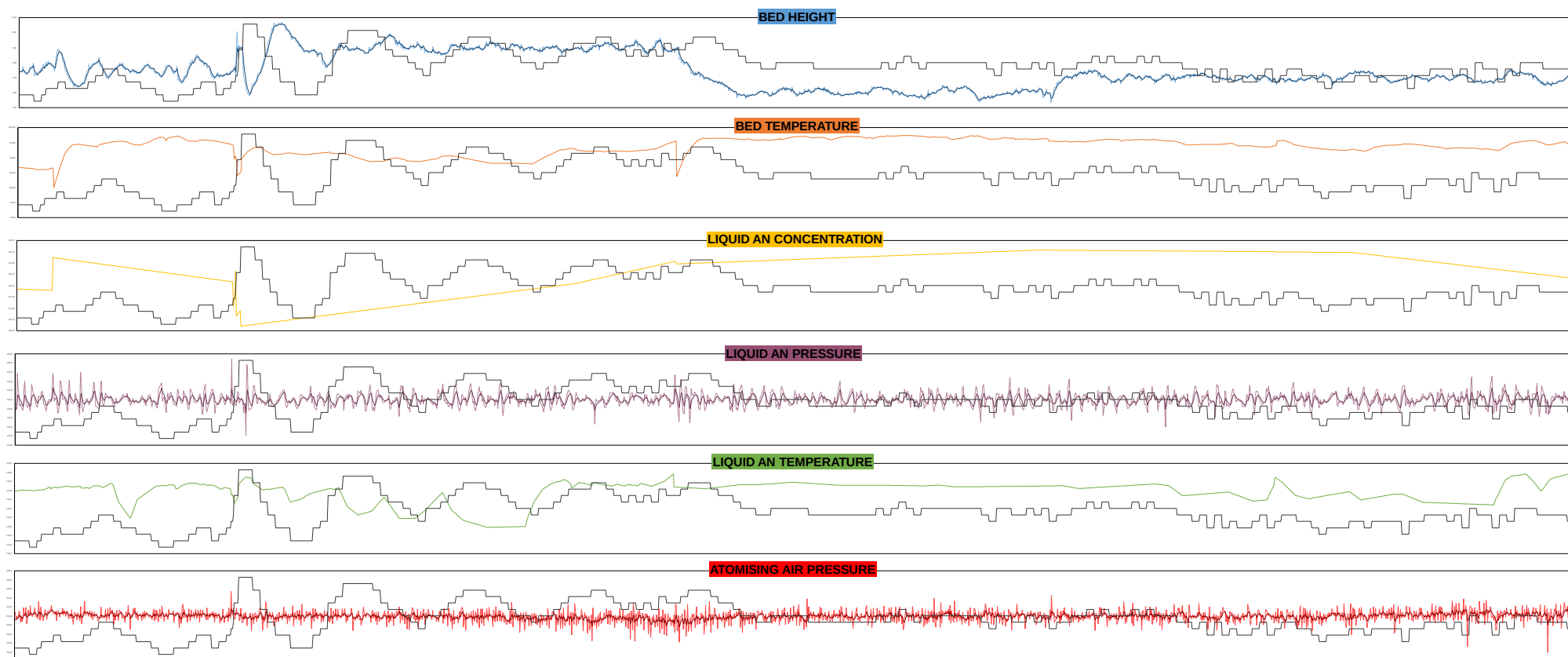


Figure E.3-1: Phase 2 sensor inputs and sphericity data

## E.4 Regression plots for phase 2 sphericity estimation

### E.4.1 Levenberg-Marquardt training algorithm

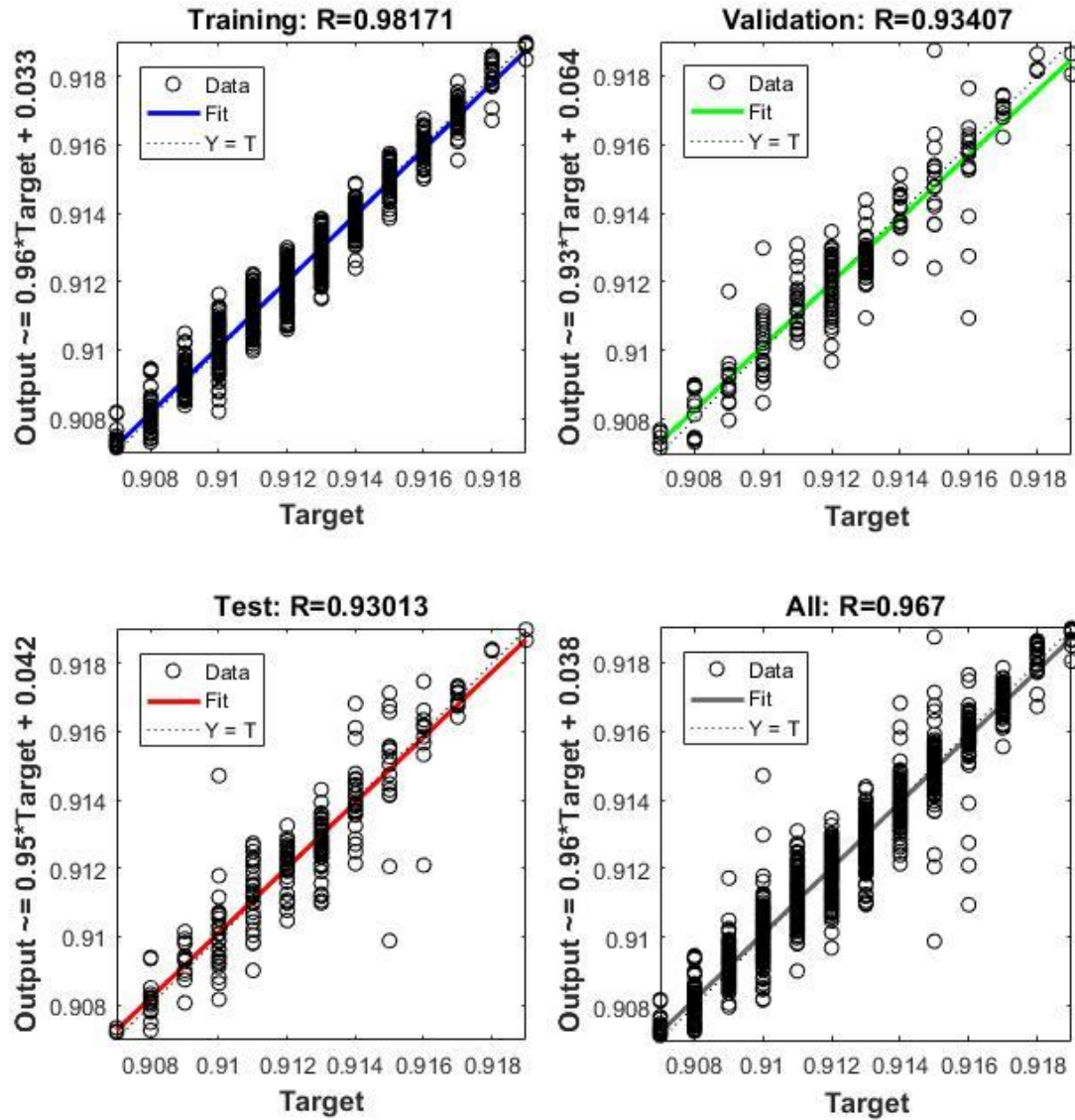


Figure E.4-1: Regression plots of the training of a 50-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data

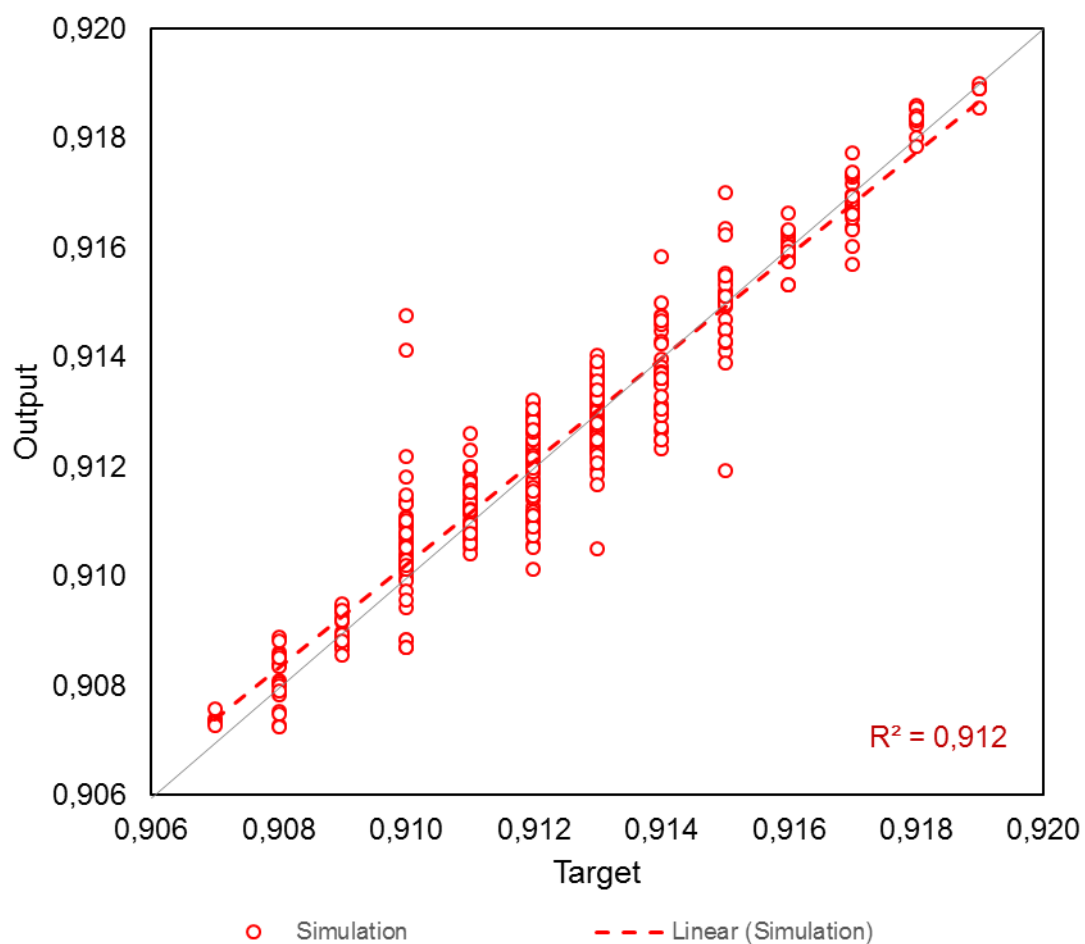


Figure E.4-2: Regression plot of the simulation of a 50-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data

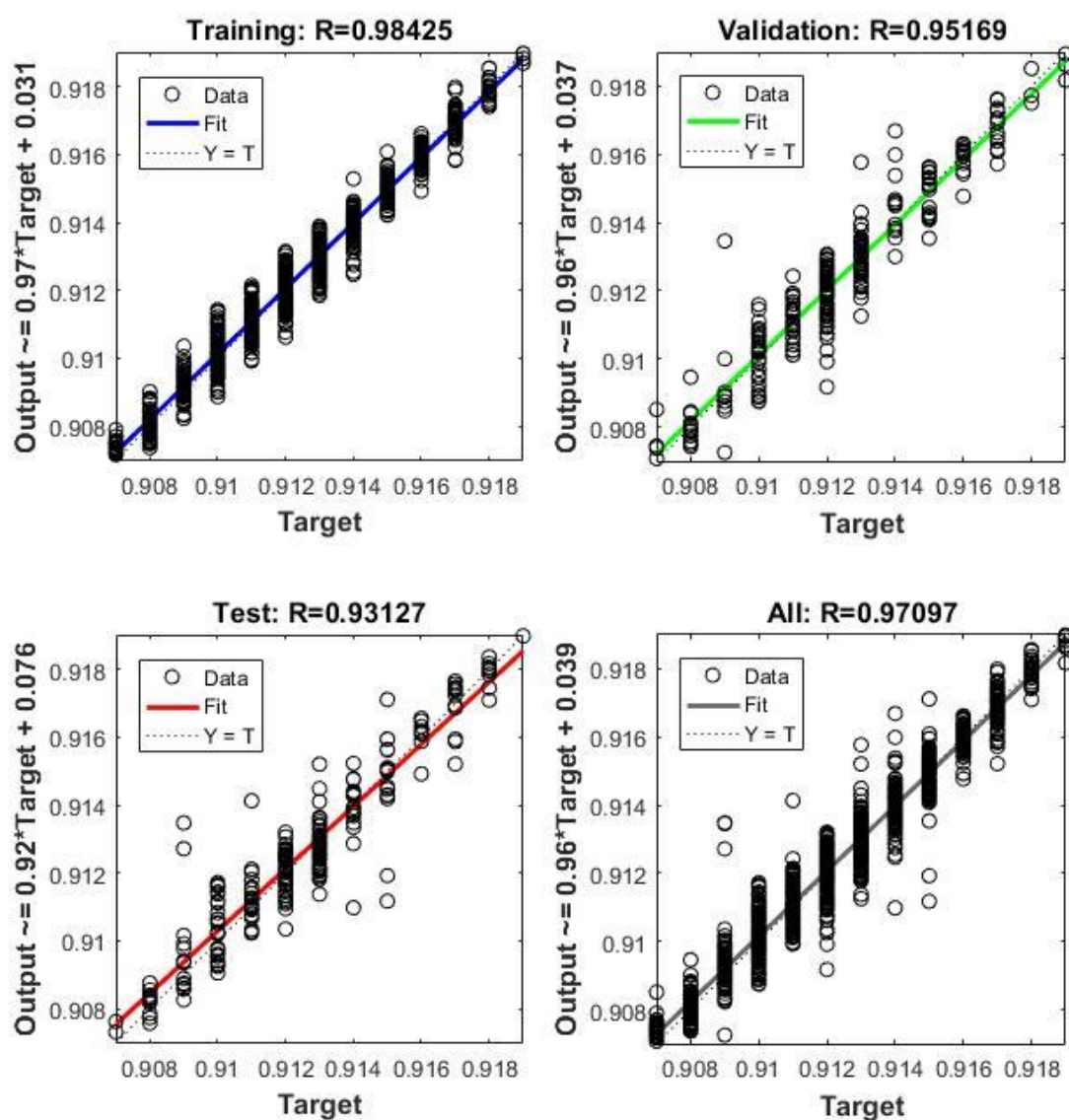


Figure E.4-3: Regression plots of the training of a 65-neuron neural network trained with the Levenberg-Marquardt algorithm using phase 2 sphericity data

## E.4.2 Bayesian regularisation training algorithm

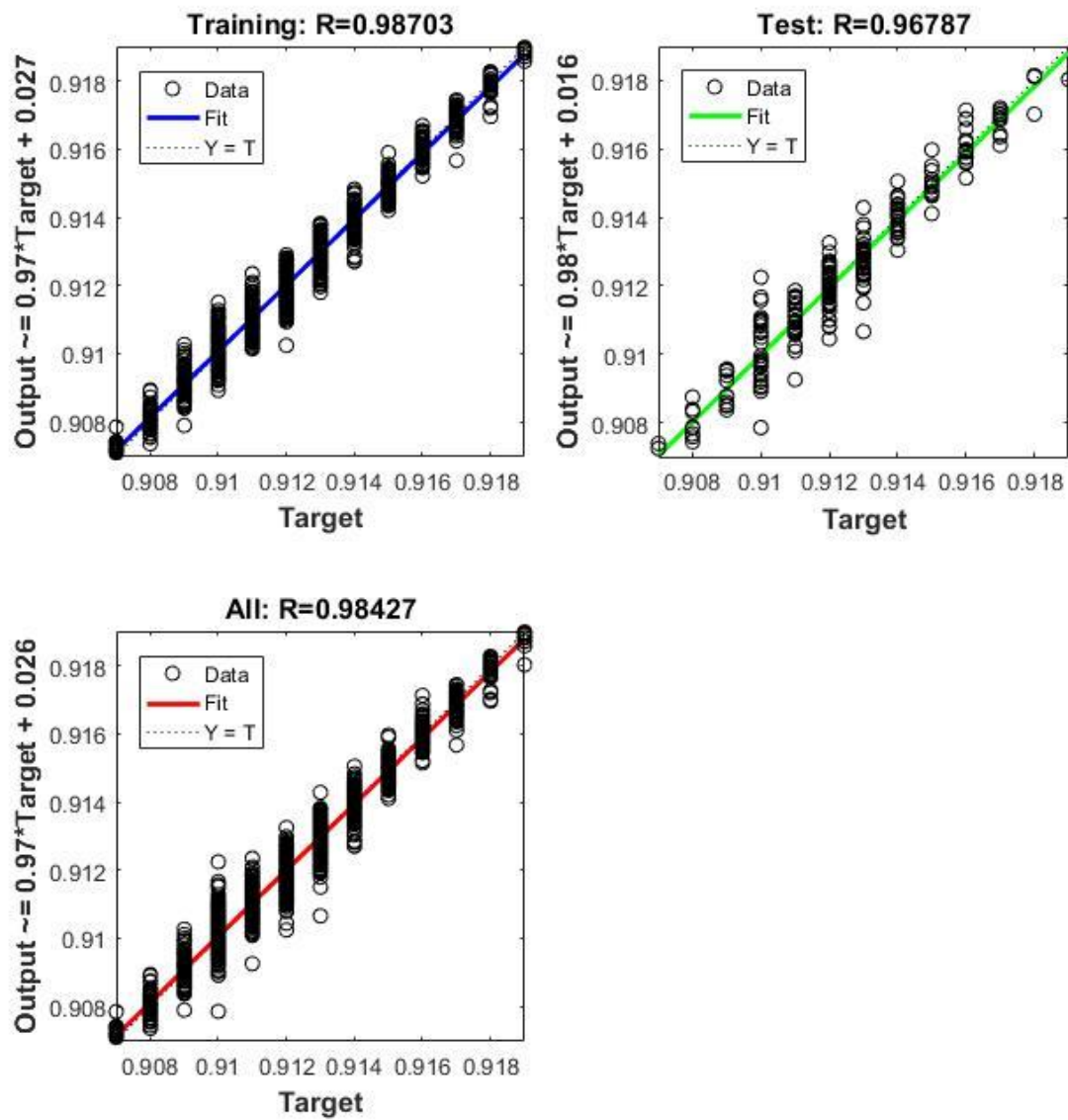


Figure E.4-4: Regression plots of the training of a 75-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 sphericity data

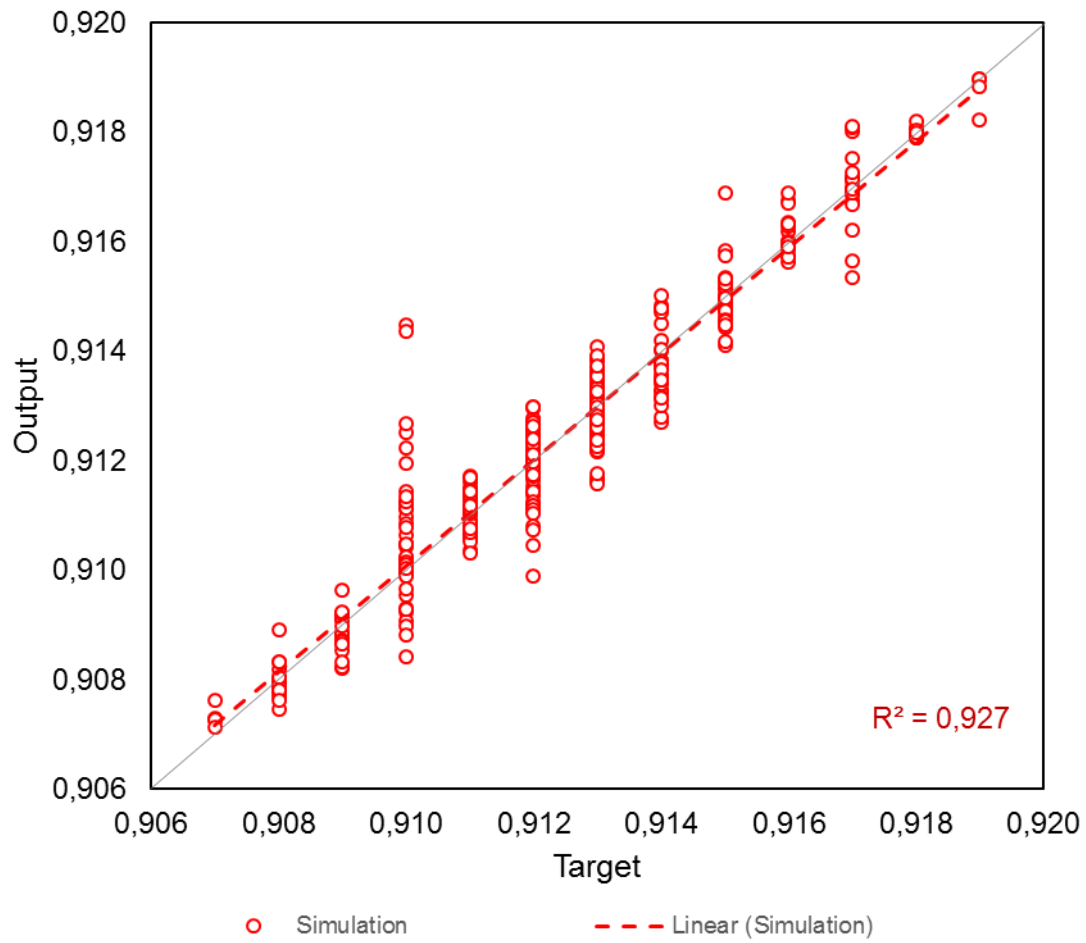


Figure E.4-5: Regression plot of the simulation of a 75-neuron neural network trained with the Bayesian regularisation algorithm using phase 2 sphericity data

## E.4.3 Activation function comparison

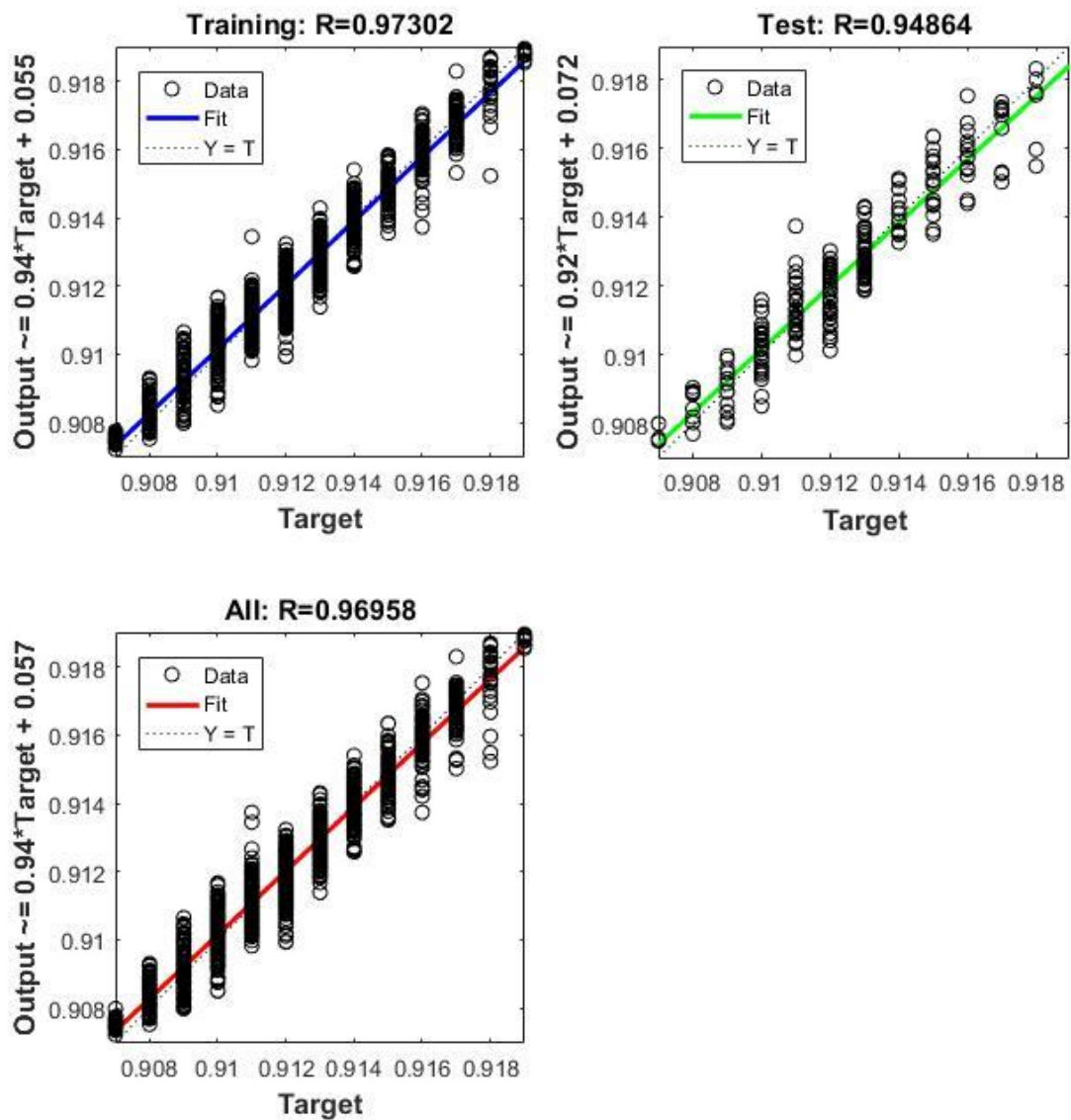


Figure E.4-6: Regression plots of the training of a 75-neuron neural network using phase 2 sphericity data and a sigmoidal activation function

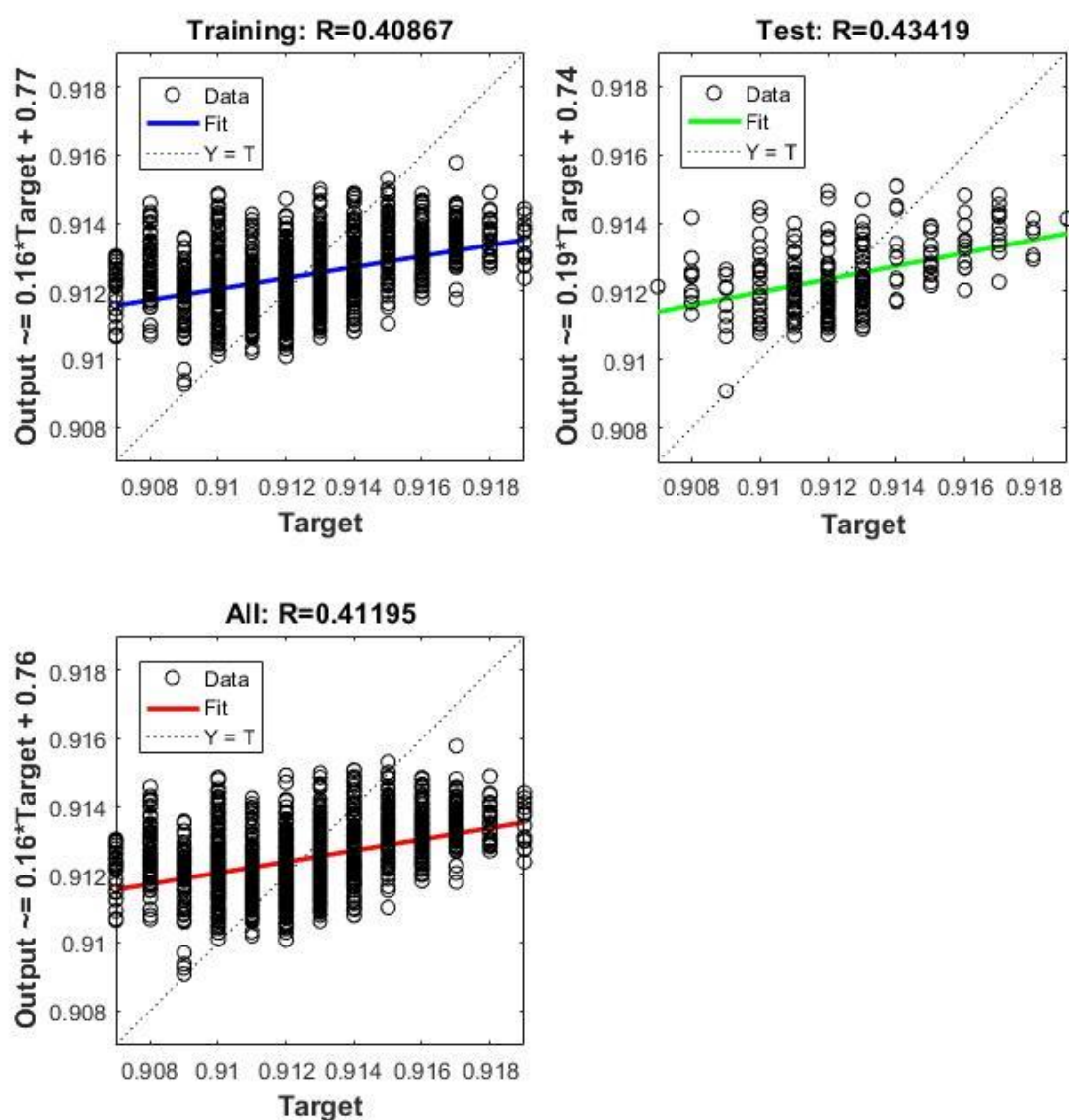


Figure E.4-7: Regression plots of the training of a 75-neuron neural network using phase 2 sphericity data and a linear activation function

## Appendix F: Raw data

The raw data is too much to show in the dissertation and can thus be found on the Excel spreadsheets provided electronically

### F.1 Phase 1 raw data

The Excel spreadsheets available are:

- MEng\_23603844\_L Botha\_Phase 1 - All data
- MEng\_23603844\_L Botha\_Phase 1 - Box-and-whisker diagrams
- MEng\_23603844\_L Botha\_Phase 1 ANN - Filtered data
- MEng\_23603844\_L Botha\_Phase 1 ANN - Discrete data
- MEng\_23603844\_L Botha\_Phase 1 ANN - Constant-time data

Table F.1-1 gives the layout of each spreadsheet mentioned above:

**Table F.1-1: Raw data - Layout of Excel spreadsheets**

| Excel spreadsheet | Work sheet                       | Description                                                           |
|-------------------|----------------------------------|-----------------------------------------------------------------------|
| All data          | Original data                    | Historical data as obtained from plant                                |
|                   | Original data plots              | Plots of original data                                                |
|                   | Filtered data                    | Data with statistical filtering                                       |
|                   | Filtered data plots              | Plots of filtered data                                                |
|                   | Discrete data                    | Data of discrete oil absorption values                                |
|                   | Discrete data plots              | Plots of discrete data                                                |
|                   | Constant-time data (10min)       | Data filtered for a constant-time interval of 10 minutes              |
|                   | Constant-time data plots (10min) | Constant-time data with linear sensor filtering                       |
|                   | Constant-time data plots (10min) | Plots of 10-minute constant-time data as well as sensor filtered data |
|                   | Constant-time data (5min)        | Data filtered for a constant-time interval of 5 minutes               |
|                   | Constant-time data (15min)       | Data filtered for a constant-time interval of 15 minutes              |
|                   | Constant-time data (20min)       | Data filtered for a constant-time interval of 20 minutes              |
|                   | Constant-time data (30min)       | Data filtered for a constant-time interval of 30 minutes              |

|                                 | Constant-time data<br>(40min) | Data filtered for a constant-time interval of 40<br>minutes                                                                                                  |
|---------------------------------|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                 | PerfReg comparison            | Comparison of best performing neuron<br>configuration of different datasets                                                                                  |
| Box-and-<br>whisker<br>diagrams | All data                      | Historical data as obtained from plant                                                                                                                       |
|                                 | FAF – Box plot                | Fluidising air flow rate data<br>Normal distribution and box-and-whisker diagram<br>of fluidising air flow rate using original data                          |
|                                 | FAF – Filtered data           | Filtered fluidising air flow rate data<br>Normal distribution of fluidising air flow rate data<br>after statistical filtering                                |
|                                 | FAT – Box plot                | Fluidising air temperature data<br>Normal distribution and box-and-whisker diagram<br>of fluidising air temperature using original data                      |
|                                 | FAT – Filtered data           | Filtered fluidising air temperature data<br>Normal distribution of fluidising air temperature<br>data after statistical filtering                            |
|                                 | LAND – Box plot               | Liquid ammonium nitrate density data<br>Normal distribution and box-and-whisker diagram<br>of liquid ammonium nitrate density using original<br>data         |
|                                 | LAND – Filtered data          | Filtered liquid ammonium nitrate density data<br>Normal distribution of liquid ammonium nitrate<br>density data after statistical filtering                  |
|                                 | LANT – Box plot               | Liquid ammonium nitrate temperature data<br>Normal distribution and box-and-whisker diagram<br>of liquid ammonium nitrate temperature using<br>original data |
|                                 | LANT – Filtered data          | Filtered liquid ammonium nitrate temperature data<br>Normal distribution of liquid ammonium nitrate<br>temperature data after statistical filtering          |
|                                 | LANF – Box plot               | Liquid ammonium nitrate flow rate data<br>Normal distribution and box-and-whisker diagram<br>of liquid ammonium nitrate flow rate using original<br>data     |
|                                 | LANF – Filtered data          | Filtered liquid ammonium nitrate flow rate data<br>Normal distribution of liquid ammonium nitrate<br>flow rate data after statistical filtering              |
|                                 | AAF – Box plot                | Atomising air flow rate data<br>Normal distribution and box-and-whisker diagram<br>of atomising air flow rate using original data                            |
|                                 | AAF – Filtered data           | Filtered atomising air flow rate data                                                                                                                        |

|                    |                                                                                                                                                                                                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                    | OA – Box plot                                                                                                                                                                                        | Normal distribution of atomising air flow rate data after statistical filtering                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|                    | OA – Filtered data                                                                                                                                                                                   | Oil absorption data<br>Normal distribution and box-and-whisker diagram of oil absorption using original data<br>Filtered oil absorption data<br>Normal distribution of oil absorption data after statistical filtering                                                                                                                                                                                                                                                                                                                                                                                                 |
| Filtered data      | Filtered data<br>Training set<br>Simulation set<br>ANN comparison table<br>ANN-SIM regression graphs<br>ANN-TRAIN regression graphs<br>ANN regression graphs<br>PerfReg comparison<br>Top ANN graphs | Data with statistical filtering<br>80% of filtered data set for training of ANN<br>20% of filtered data set for simulation of ANN<br>Training and simulation output values for best performing neural networks<br>Regression graphs for the simulation of best performing neural networks<br>Regression graphs for the training of best performing neural networks<br>Regression graphs for the training and simulation of best performing neural networks<br>Table and graphs of MSE- and R-values for all tested configurations<br>Model performance graphs for best performing neural network configurations        |
| Discrete data      | Discrete data<br>Training set<br>Simulation set<br>ANN comparison table<br>ANN-SIM regression graphs<br>ANN-TRAIN regression graphs<br>ANN regression graphs<br>PerfReg comparison<br>Top ANN graphs | Data of discrete oil absorption values<br>80% of discrete data set for training of ANN<br>20% of discrete data set for simulation of ANN<br>Training and simulation output values for best performing neural networks<br>Regression graphs for the simulation of best performing neural networks<br>Regression graphs for the training of best performing neural networks<br>Regression graphs for the training and simulation of best performing neural networks<br>Table and graphs of MSE- and R-values for all tested configurations<br>Model performance graphs for best performing neural network configurations |
| Constant-time data | Constant-time data (10min)                                                                                                                                                                           | Data filtered for a constant-time interval of 10 minutes                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

|                             |                                                                                                                                          |
|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| Training set (10min)        | 80% of constant-time data (10min) set for training of ANN                                                                                |
|                             | 10-minute constant-time data for training with linear sensor filtering                                                                   |
| Simulation set (10min)      | 20% of constant-time data (10min) set for simulation of ANN                                                                              |
|                             | 10-minute constant-time data for simulation with linear sensor filtering                                                                 |
| ANN comparison table        | Training and simulation output values for best performing neural networks including that of data with linear sensor filtering            |
| ANN-SIM regression graphs   | Regression graphs for the simulation of best performing neural networks including that of data with linear sensor filtering              |
| ANN-TRAIN regression graphs | Regression graphs for the training of best performing neural networks including that of data with linear sensor filtering                |
| ANN regression graphs       | Regression graphs for the training and simulation of best performing neural networks including that of data with linear sensor filtering |
| PerfReg comparison          | Table and graphs of MSE- and R-values for all tested configurations                                                                      |
|                             | Table and graph of time interval comparison                                                                                              |
|                             | Table and graph of linear sensor comparison                                                                                              |
| Top ANN graphs              | Model performance graphs for best performing neural network configurations including that of data with linear sensor filtering           |
| Constant-time data (5min)   | Data filtered for a constant-time interval of 5 minutes                                                                                  |
| Training set (5min)         | 80% of constant-time data (5min) set for training of ANN                                                                                 |
| Simulation set (5min)       | 20% of constant-time data (5min) set for simulation of ANN                                                                               |
| Constant-time data (15min)  | Data filtered for a constant-time interval of 15 minutes                                                                                 |
| Training set (15min)        | 80% of constant-time data (15min) set for training of ANN                                                                                |
| Simulation set (15min)      | 20% of constant-time data (15min) set for simulation of ANN                                                                              |
| Constant-time data (20min)  | Data filtered for a constant-time interval of 20 minutes                                                                                 |
| Training set (20min)        | 80% of constant-time data (20min) set for training of ANN                                                                                |

|  |                            |                                                             |
|--|----------------------------|-------------------------------------------------------------|
|  | Simulation set (20min)     | 20% of constant-time data (20min) set for simulation of ANN |
|  | Constant-time data (30min) | Data filtered for a constant-time interval of 30 minutes    |
|  | Training set (30min)       | 80% of constant-time data (30min) set for training of ANN   |
|  | Simulation set (30min)     | 20% of constant-time data (30min) set for simulation of ANN |
|  | Constant-time data (40min) | Data filtered for a constant-time interval of 40 minutes    |
|  | Training set (40min)       | 80% of constant-time data (40min) set for training of ANN   |
|  | Simulation set (40min)     | 20% of constant-time data (40min) set for simulation of ANN |

## F.2 Phase 2 raw data

The Excel spreadsheets available are:

- MEng\_23603844\_L Botha\_Phase 2 - All data\_Oil absorption
- MEng\_23603844\_L Botha\_Phase 2 - All data\_Sphericity
- MEng\_23603844\_L Botha\_Phase 2 - Porosity calculations
- MEng\_23603844\_L Botha\_Phase 2 - Advances
- MEng\_23603844\_L Botha\_Phase 2 - ANN data\_Oil absorption
- MEng\_23603844\_L Botha\_Phase 2 - ANN data\_Sphericity

Table F.2-1 gives the layout of each spreadsheet given above:

**Table F.2-1: Raw data - Layout of Excel spreadsheets**

| Excel spreadsheet         | Work sheet      | Description                                           |
|---------------------------|-----------------|-------------------------------------------------------|
| All data – Oil absorption | All data        | All data obtained for phase 2 porosity                |
|                           | All data graphs | Graphs of all data obtained for phase 2 porosity      |
|                           | Data – Advances | Data without downtime and with time-advance           |
|                           | Advanced graphs | Graphs of data without downtime and with time-advance |
| All data – Sphericity     | All data        | All data obtained for phase 2 sphericity              |
|                           | All data graphs | Graphs of all data obtained for phase 2 sphericity    |

|                           |                                                                                                                                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                           | Data – Advances<br>Advanced graphs                                                                                                             | Data without downtime and with time-advance<br>Graphs of data without downtime and with time-advance                                                                                                                                                                                                                                                                                                                                                              |
| Porosity calculations     | Vol-to-Mass<br>Oil absorption<br>Repeatability                                                                                                 | Calculating porosity using the optical porosity measurement method for each sample taken<br>Calculating porosity using the Oil absorption method for 20 of the samples taken<br>Repeatability calculations for 3 of the samples                                                                                                                                                                                                                                   |
| Advances                  | Fluidising air pressure<br>Bed temperature<br>Liquid AN concentration<br>Liquid AN pressure<br>Liquid AN temperature<br>Atomising air pressure | Advance calculations on Fluidising air pressure<br>Advance calculations on Bed temperature<br>Advance calculations on Liquid AN concentration<br>Advance calculations on Liquid AN pressure<br>Advance calculations on Liquid AN temperature<br>Advance calculations on Atomising air pressure                                                                                                                                                                    |
| ANN data – Oil absorption | Training<br>Simulation<br>ANN comparison table<br>ANN regression graphs<br>PerfReg comparison<br>Top ANN graphs                                | 80% of phase 2 porosity estimation data et for training of ANN<br>20% of phase 2 porosity estimation data set for simulation of ANN<br>Training and simulation output values for best performing neural networks<br>Regression graphs for the training and simulation of best performing neural networks<br>Table and graphs of MSE- and R-values for all tested configurations<br>Model performance graphs for best performing neural network configurations     |
| ANN data – Oil absorption | Training<br>Simulation<br>ANN comparison table<br>ANN regression graphs<br>PerfReg comparison<br>Top ANN graphs                                | 80% of phase 2 sphericity estimation data et for training of ANN<br>20% of phase 2 sphericity estimation data set for simulation of ANN<br>Training and simulation output values for best performing neural networks<br>Regression graphs for the training and simulation of best performing neural networks<br>Table and graphs of MSE- and R-values for all tested configurations<br>Model performance graphs for best performing neural network configurations |