

Investigation of factorisation machines and its extensions to predictive models in a statistical context

E Slabber

 orcid.org/0000-0002-7301-7001

Thesis accepted in fulfilment of the requirements for the degree
Doctor of Philosophy in Science with Business Mathematics at
the North-West University

Promoter: Prof T Verster

Co-promoter: Prof PJ de Jongh

Graduation May 2023

21139032

Acknowledgements

First and foremost, I would like to thank my Heavenly Father for His love and grace and for granting me the wisdom and strength to complete my PhD study.

I wish to express my appreciation and gratitude to the following individuals:

- My promoter, Prof T Verster for her invaluable advice, assistance and continuous support and patience during my research. Her knowledge and experience have encouraged me during this time. Thank you for mentoring me, encouraging my research and for allowing me to grow as a research scientist.
- My co-promoter and father, Prof PJ de Jongh. Never as a young kid have I thought I will learn this much from you in the field of statistical research. It has been an honour and something I will take with me forever. Thanks for sharing your knowledge and for your love and support throughout my life.
- A special thanks to my husband, Kobus, for his support, patience, encouragement, and endless love. Thanks for your motivation, believing in me and being someone I can look up to.
- Finally, my mom for her love and support throughout my life. You are a pillar in my life and your prayers sustained me thus far.

Abstract

Factorisation machines originated from the field of machine learning and have gained popularity because of the high accuracy obtained in several prediction problems, particularly in the field of recommender systems. As will be seen in the introductory discussion statisticians are largely unaware of factorisation machines and their capabilities and are seldomly considered by financial companies in their model-building endeavours. On the other hand, generalised linear models are frequently used by financial companies, and given the close relation of factorisation machines to these models, it is rather unfortunate that factorisation machines receive little attention. It is also surprising that factorisation machines are seldom if ever, mentioned as an alternative technique to benchmark models when problems in the financial services industry are discussed. Other than our papers, we have not been able to find a paper on factorisation machines published in a statistical journal. One of the reasons is probably since machine learners use terminology unfamiliar to statisticians. Almost all of the papers on factorisation machines are published in conference proceedings and in engineering and artificial intelligence journals. Although banks and insurance companies make use of open-source software, their production systems are usually coded in commercial software. Currently, commercial software is rather restricted in the routines that are offered to fit factorisation machines. So, to facilitate application of factorisation machines to a wide range of problems, a suite of fitting routines is needed.

In this thesis, we introduce factorisation machines to the statistical community and show that they can be applied to a wide range of modelling problems. Specifically, we compare and relate factorisation machines to generalised linear models and we develop several fitting routines in popular commercial software. The accuracy of the routines is thoroughly checked employing simulation studies. The performance of factorisation machines is then compared to that of popular generalised linear models by using Monte Carlo simulation and several real-world data sets. Specific contributions of the thesis include the introduction of a new robust factorisation machine, based on mean absolute deviation, and logistic factorisation machines that perform well on binary classification problems where the number of events is much less than the number of non-events. We further motivate why and in what setting it will be beneficial to include factorisation machines as part of the Statistician's model-building toolkit. During our research, understanding and analysis of FMs, we concluded that FMs can potentially be used to solve a range of exciting business problems, and we include some of these as recommendations for future research. Lastly, we believe that the routines developed in this

thesis will result in the widespread application of factorisation machines in business. Also, we hope that the publications resulting from this thesis will stimulate research interest on this topic by statisticians.

Keywords: *Factorisation machines, Machine Learning, Recommender systems, Linear regression, Parameter estimation, Logistic regression, Credit scoring*

Preface

This thesis is written in article format. The research reported in this thesis was done in conjunction with my promoter, Professor Tanja Verster and co-promoter, Professor Riaan de Jongh and consists of two published articles and one submitted article.

In Chapter 1 a general overview of this thesis is given and it includes a literature study. The three articles can be found in Chapters 2, 3 and 4. The articles were written for this thesis and were submitted to the indicated journals for publication. In Chapter 5 the main findings of the thesis are reported along with conclusions and avenues for future research. Appendix A contains the author guidelines of the South African Statistical Journal and Risks.

The three articles are:

- The first article, *Advantages of using factorisation machines as a statistical modelling technique*, has been published in the South African Statistical Journal 2021, Vol. 55, No. 2, 125 - 144. DOI: <https://doi.org/10.37920/sasj.2021.55.2.3> Authors: E Slabber, T Verster and PJ de Jongh.
- The second article, *Algorithms for estimating the parameters of factorisation machines*, has been published for publication in the South African Statistical Journal 2022, Vol. 56, No. 2, 69 - 89. DOI: <https://doi.org/10.37920/sasj.2022.56.2.1> Authors: E Slabber, T Verster and PJ de Jongh.
- The third article, *Some insights about the applicability of logistic factorisation machines in banking*, has been submitted to Risks. Authors: E Slabber, T Verster and PJ de Jongh.

The editor confirmed that there is no copyright involved in the publishing of the articles as part of the thesis. The promoters agreed on co-authorship and gave consent for the use of these articles as part of the final thesis. The student, together with the supervisors, planned the thesis and the student was then responsible for all literature searches, coding of algorithms, interpretation of results, and the initial drafting of the papers.

CONTENTS

Chapter 1: Introduction (including literature study)	1
Chapter 2: Article 1 - Advantages of using factorisation machines as a statistical modelling technique.....	12
Chapter 3: Article 2 - Algorithms for estimating the parameters of factorisation machines	33
Chapter 4: Article 3 - Some insights about the applicability of logistic factorisation machines in banking	55
Chapter 5: Conclusion	76
Appendix A: Author guidelines for SASJ and Risks	78

Chapter 1

Introduction

Since the first paper on factorisation machines (FMs) appeared in 2010, the research in and application of these methods exploded. A recent Google Scholar search on ‘factorisation machines’ yielded 101 000 papers of which 5 600 appeared in 2022. It is therefore a daunting task to provide a comprehensive overview of FMs, their extensions, and their applications. Yet, before getting into the objectives of this thesis and our contributions to the field, we will attempt to provide an overview of the latest developments in a form of a literature study. The references listed at the end of this chapter serve as an extract of the latest contributions in the field. This bibliography clearly shows that most of the papers are published in conference proceedings and journals in the machine learning, engineering, and computer science fields. Very few, if any of these papers have been published in statistical journals giving the impression that statisticians are largely unaware of FMs, their capabilities and application areas. The terminology used in the machine learning literature is not familiar to statisticians and thus a contributing factor to statisticians largely being ignorant of the methods. Given the close relation of FMs to generalised linear models (GLMs) in statistics, this is rather unfortunate. In fact, if GLMs have been used in an application area, then FMs could also be applied. However, FMs became popular due to their successes in solving prediction problems emanating from the Internet of Things (IoT) and using large data sets in the process. Examples of prediction problems include, but are not limited to, recommender system applications. Recommender systems address various problems such as recommending to a user, based on the user’s profile, the best product to purchase, the most appropriate music track to listen to, a movie to watch, what news to read and content recommenders for social media platforms. The number of predictor variables or features that can be used to make predictions are vast, and often most of the variables or features are categorical (often nominal) having many levels. Coding these variables typically leads to vast, complex, and sparse multi-dimension input feature spaces. It is in these situations, where the number of variables or features far outnumber the available observations, that FMs excel and shine because of their ability to pick up associations between features with limited data.

Given this background, we will now discuss some of the applications of FMs. Not surprisingly, the many different applications raised different challenges leading to research that led to extensions of FMs.

As stated previously, the most successes of FMs were recorded in the field of **recommender systems**. A recommender system, in broad terms, is about suggesting to a user which items are most relevant or suitable. The suggestions could refer to various decision-making processes, such as suggesting to a user the best product purchase, the most appropriate music track to listen to, a movie that most closely relates to the user's preference, and news items to read. These systems are particularly useful in assisting a user to select, based on the user's profile, the most appropriate item from many items that a service may offer. Recommender systems are closely related to collaborative filtering approaches to build models based on a user's past behaviour or on similar decisions made by other users. Collaborative filtering is discussed in more detail in the next paragraph. It is claimed that recommender systems can complement search algorithms by assisting users in discovering items they probably would not have found otherwise. Of note, recommender systems are often implemented using search engines that index non-traditional data. Recommender systems based on FMs have been built to solve several problems. For example, for mobile app recommendation (see Liang et al. (2020), Cao et al. (2019) and Cao et al. (2020)), hotels recommendation (see Chen and Yao (2018)), for next track music recommendation (see Gabriel (2020)), collaborative recommendation (see Liu et al. (2018)), top-n feature-based recommendation (see Yuan et al. (2017)) and sequential recommendation (see Wen and Zhang (2020)). Although most of the FM recommender systems have been built using second-order interactions, higher-order FMs have been developed by Knoll (2016) and Knoll et al. (2017). The latter study (Knoll et al., 2017) includes a comparison of state-of-the-art recommender algorithms. Due to the challenges posed by the many different applications several new variants of FMs have been introduced such as field-matrixed FMs (Sun et al., 2021), interaction-aware FMs (Hong et al., 2019), and input-aware FMs (Yu et al., 2019).

Collaborative filtering assumes that people who agreed in the past will agree in the future, or that they will like similar items as they liked in the past. Its objective is to automatically predict the preference of a user, by matching the user to similar obtained by filtering through a database containing the preferences of many different users on various items or issues. For example, if person A has the same opinion as person B on an issue, A is more likely to have B's opinion on a different issue than that of a randomly chosen person. Collaborative filtering models have been applied to many kinds of data including sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors; electronic commerce and web applications where the focus is on user data; or financial data, such as financial service institutions that integrate

many financial sources, etc. Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity. Cold start implies that for a new user or item there is not enough data to make accurate recommendations. Scalability is an issue because often significant computational power is necessary to process the large number of preferences of millions of users for millions of items. Sparsity is caused by the availability of a very small subset of all possible ratings (of users on items). FMs have proven themselves as worthy collaborative filtering models for predicting user interests and ratings. See for example Liu et al. (2018) and Loni et al. (2014). Hong et al. (2013) introduced an FM variant, called a Co-FMs and illustrated its performance in predicting individual decisions on Twitter.

User behaviour analysis for classification and prediction is a popular topic in data science. This type of analysis is performed for several purposes, such as finding users' interests towards an event (elections, championships, etc.) or about a product (for marketing, e-commerce, etc). It will be beneficial for marketers to target specific customers with the marketing actions likely to have the most desirable impact because it will lead to more successful marketing campaigns and retention action outcomes. The return on investment of upsell, cross-sell and retention campaigns will be greater. Many authors (see e.g. Geuens (2015), Wang et al. (2016) and Chen et al. (2016)) have used FMs successfully in various applications in this field.

Click-through rate prediction is the ability to predict if a user clicks or not in an advertisement. In online advertising, click-through rate (CTR) is a very important metric for evaluating the effectiveness of advertisements. As a result, click prediction systems are essential and widely used for sponsored search and real-time bidding. So, CTR prediction plays an important role in computational advertising. Often huge data sets containing many categorical features must be analysed. Therefore, it is not surprising that FMs are often used for CTR prediction problems. For a recent literature review on the topic, the interested reader is referred to Yang and Zhai (2022). To cater for the issues that come with CTR prediction problems, various adaptations were made to standard FM models. For example, sometimes these data sets are so large and sparse that even FMs have difficulty fitting these data sets. To improve FMs, Juan et al. (2016) introduced so-called Field Aware FMs which are particularly adept to handle large, sparse datasets with many categorical features. One of the key ideas was incorporating a scoring function which maps feature vectors from different fields to a scalar score. Other adaptations include Field-weighted FMs (Pan et al., 2018), hybrid systems incorporating

FMs and gradient-boosting decision trees (Wang et al., 2019), FMs and neural networks (see Toğuç and Kuzu (2020) and Guo et al. (2017)) and FMs and deep learning (Li et al., 2019).

FMs have been applied successfully in the field of **context-aware computing**. This is a style of computing in which situational and environmental information about people, places and things is used to anticipate immediate needs and then proactively offer enriched, situation-aware, and usable content, functions, or experiences. An example is to predict user behaviours such as a click or a purchase of products. When characterising user behaviour data-rich context information is available, such as user demographics, item attributes, time and location of the transaction, historical records, and information on last transactions. To learn from such context-rich data a universal solution is to first convert it into high-dimensional generic feature vectors (e.g. by using one hot coding) and then build predictive models on the featured inputs. This poses difficulty for support vector machines and deep neural networks since they are typically not tailored for learning from sparse data. The key ingredient to designing effective models is to cater for the interactions among features. In this context, Xin et al. (2019) developed a convolutional FM for context-aware recommendation, that significantly outperforms existing methods. This method incorporates higher-order interactions and is not limited to two-way interactions only. Other applications of context-aware prediction, amongst others, are discussed by Petroni et al. (2015) and Rendle et al. (2011).

As stated previously, big data comprise of many features of which many are redundant. The process of determining a subset of informative or non-redundant features is called **feature selection** and is related to dimensionality reduction. The process is also referred to as **feature engineering**. Standard FMs give the same weight to all feature interactions, and because of the presence of redundant features, the prediction performance of FMs can be negatively affected. For example, the interactions with redundant features may introduce noise degrading the overall performance of the model. Xiao et al. (2017) proposed an Attentional FM (AFM), which learns the importance of each feature interaction from data via a neural attention network. It is shown that AFM outperforms deep learning methods having a much simpler structure and fewer model parameters. Related papers on FMs as a feature engineering tool include Chen et al. (2019) and Atarashi et al. (2021). Other authors have used AFMs for landslide susceptibility mapping (see Liu et al. (2021)) and for driving safety risk prediction (see Zhang et al. (2021)).

Other applications of FMs include knowledge tracing, sentiment classification, health, finance, and statistics. **Knowledge tracing** is the task of modelling student knowledge over time to accurately predict how students will perform in future interactions. A bibliometric analysis of this field is provided by Liu (2022). Vie (2018), Vie and Kashima (2019) and Zhang and Li (2021) have applied FMs to knowledge tracing. FMs have been used for **sentiment classification** (see e.g. Wang et al. (2018)), which entails identifying opinions in text and labelling them as positive, negative, or neutral, based on the emotions customers express within them. Health-related applications include, amongst others, FMs for modelling the efficacy of anti-cancer drug combinations (Julkunen, 2019) and anti-cancer drug recommendations (Su et al., 2022). **Financial applications** include risk management (see Angulo et al. (2018)), trading and market prediction (see Stübinger et al. (2017)) and driving safety risk (see Zhang et al. (2021)). In statistics, **outlier detection and robust estimation** are well-established fields. Punjabi and Bhatt (2018) and Ni et al. (2018) used robust FMs for robust user response prediction and credit default prediction, respectively, while Zhu et al. (2017) used FMs for outlier detection in sparse data sets.

FMs have benefitted from other fields in the machine learning literature such as deep learning, neural networks, and reinforcement learning. Deep learning and neural networks are well-known and part of the modelling toolkit of machine learners, and have been used by several authors to enhance the capabilities of FMs (see Zhang et al. (2016) and He and Chua (2017)). **Reinforcement** learning is a machine learning training method, closely related to Markov decision processes, and based on rewarding desired behaviours and/or punishing undesired ones. In general, a reinforcement learning agent is able to perceive and interpret its environment, take actions and learn through trial and error. While reinforcement learning has been a topic of much interest in the field of machine learning, its real-world adoption and application remain limited. In a recent paper, Yu and Yin (2022) combined deep learning concepts and reinforcement learning to create an extension of FMs termed ‘deep reinforcement FMs’. Recently, Buet-Golfouse and Utyagulov (2021) combined the kernel trick, a method of using a linear classifier to solve a non-linear problem and published a paper entitled ‘kernel FMs’.

In summary, clearly, the research and applications regarding FMs are increasing at a fast pace. The focus currently is largely on prediction problems fuelled by the Internet of Things using a multitude of features contained in large sparse data sets. Despite the many successes of FMs, most statisticians are oblivious to these methods which can be applied in any application where GLMs are used. Banks

and insurance companies often use regression, logistic regression, classification trees, random forests, and sometimes neural networks, but FMs are seldom considered despite their successes in the machine learning literature. A contributing factor for this is possibly the fact that FMs are not included in the statistical learning and machine learning courses presented by local universities so that data scientists in those companies are ignorant about the technique and its potential applications. Another factor is that commercial software for fitting a wide class of FMs is not readily available. Although banks and insurance companies make use of open-source software, their production systems are usually coded in commercial software like SAS. Further issues are the lack of publications about FMs in statistical journals and the confusion caused by papers using terminology unfamiliar to statisticians. This thesis is an attempt to fill this void.

Therefore, the **objective of this thesis** is to introduce FMs to the statistical community and to make available a set of fitting routines in the widely used SAS software. In the process, we compare FMs to well-known statistical models (GLMs) and evaluate their performance using simulation studies as well as real-world data sets. This will include demonstrating the advantages and disadvantages of FMs over GLMs. As far as we know, such a comprehensive study has not been undertaken yet.

Since this is an article-based thesis, the three papers are reproduced in Chapter 2, Chapter 3 and Chapter 4, respectively. The first and second papers have been published in the *South African Statistical Journal* and the third paper has been submitted to *Risks* for consideration. **The first paper (Chapter 2)** is entitled '*Advantages of using factorisation machines as a statistical modelling technique*'. As stated in the abstract the paper motivates the use of FMs, discusses the fundamentals of FMs, and provides examples of some applications and possible gains by using FMs. Data sets and existing software packages are used to illustrate how FMs may be fitted and in what context it is worth being used. The paper concludes by urging statisticians to include this powerful technique in their model-building toolkit. **The second paper (Chapter 3)** is entitled '*Algorithms for estimating the parameters of factorisation machines*'. As stated in the abstract, the paper implements algorithms for fitting FMs in SAS. These algorithms minimise two loss functions, namely the weighted sum of squared errors and the weighted sum of absolute deviations using coordinate descent and nonlinear programming procedures. The latter algorithm provides a new robust version of a FM having the ability to pick up associations between users and items when outliers are present. A simulation study is used to test the routines in terms of accuracy and efficiency and the predictive power is illustrated by analysing two data sets. While the second paper focuses more on regression models and related

applications, the third paper focuses on logistic regression-type applications. **The third paper (Chapter 4)** is entitled '*Some insights about the applicability of logistic factorisation machines in banking*'. The paper investigates the applicability of FMs to some binary classification problems in banking. Fitting routines, based on logit loss and maximum likelihood, is implemented on commercially available software that is widely used by banks and other large financial services companies. Logit loss is usually used by the machine learning fraternity while maximum likelihood is popular in statistics. Depending on the coding of the target variable, we show that these methods yield identical parameter estimates. Often banks are confronted with predicting events that occur with low probability. To deal with this phenomenon we introduce weights in the above-mentioned loss functions. The accuracy of our fitting algorithms is then studied employing a simulation study and compared with logistic regression. The separation and prediction performance of the methods are then investigated using three case studies covering a recommender system, credit card fraud and credit scoring application. We conclude that logistic FMs are worthy competitors of logistic regression in most applications, but with clear advantages in applications where the number of predictors outnumbers the number of observations.

Chapter 5 concludes the thesis with some summary remarks and recommendations for future research. Finally, we hope that the thesis will contribute to the wide application of FMs in industry and that it will stimulate research in these methods.

REFERENCES

- ANGULO, P., GALLEGRO, V., GÓMEZ-ULLATE, D. & SUÁREZ-GARCÍA, P. 2018. Bayesian Factorization Machines for Risk Management and Robust Decision Making. *Mathematical and Statistical Methods for Actuarial Sciences and Finance*. Springer.
- ATARASHI, K., OYAMA, S. & KURIHARA, M. 2021. Factorization Machines with Regularization for Sparse Feature Interactions. *J. Mach. Learn. Res.*, 22, 153:1-153:50.
- BUET-GOLFOUSE, F. & UTYAGULOV, I. Kernel Factorisation Machines. 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), 2021. IEEE, 1755-1760.
- CAO, B., CHEN, J., LIU, J. & WEN, Y. 2020. A topic attention mechanism and factorization machines based mobile application recommendation method. *Mobile Networks and Applications*, 25, 1208-1219.
- CAO, B., LI, B., LIU, J., TANG, M., LIU, Y. & LI, Y. 2019. Mobile service recommendation via combining enhanced hierarchical Dirichlet process and factorization machines. *Mobile Information Systems*, 2019.
- CHEN, C., HOU, C., XIAO, J. & YUAN, X. 2016. Purchase behavior prediction in e-commerce with factorization machines. *IEICE TRANSACTIONS on Information and Systems*, 99, 270-274.
- CHEN, Y., REN, P., WANG, Y. & DE RIJKE, M. Bayesian personalized feature interaction selection for factorization machines. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019. 665-674.
- CHEN, Y. & YAO, D. 2018. HOTELS RECOMMENDATION METHODS BASED ON PATTERN RECOGNITION AND FACTORIZATION MACHINES. *Latin American Applied Research-An international journal*, 48, 169-174.
- GABRIEL, M. 2020. *Distributed factorization machines for next-track music recommendation*. Wien.
- GEUENS, S. Factorization machines for hybrid recommendation systems based on behavioral, product, and customer data. Proceedings of the 9th ACM conference on recommender systems, 2015. 379-382.
- GUO, H., TANG, R., YE, Y., LI, Z. & HE, X. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247*.
- HE, X. & CHUA, T.-S. Neural factorization machines for sparse predictive analytics. Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval, 2017. 355-364.

- HONG, F., HUANG, D. & CHEN, G. Interaction-aware factorization machines for recommender systems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019. 3804-3811.
- HONG, L., DOUMITH, A. S. & DAVISON, B. D. Co-factorization machines: modeling user interests and predicting individual decisions in Twitter. *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013. 557-566.
- JUAN, Y., ZHUANG, Y., CHIN, W.-S. & LIN, C.-J. Field-aware factorization machines for CTR prediction. *Proceedings of the 10th ACM conference on recommender systems*, 2016. 43-50.
- JULKUNEN, H. 2019. Predictive modeling of anticancer efficacy of drug combinations using factorization machines.
- KNOLL, J. Recommending with higher-order factorization machines. *International Conference on Innovative Techniques and Applications of Artificial Intelligence*, 2016. Springer, 103-116.
- KNOLL, J., KÖCKRITZ, D. & GROß, R. Markov random walk vs. higher-order factorization machines: a comparison of state-of-the-art recommender algorithms. *International Conference on Innovations for Community Services*, 2017. Springer, 87-103.
- LI, Y., WANG, Y., CHEN, C. & HUANG, J. CTR Prediction with User Behavior: An Augmented Method of Deep Factorization Machines. *2019 IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, 2019. IEEE, 822-826.
- LIANG, T., ZHENG, L., CHEN, L., WAN, Y., PHILIP, S. Y. & WU, J. 2020. Multi-view factorization machines for mobile app recommendation based on hierarchical attention. *Knowledge-Based Systems*, 187, 104821.
- LIU, L.-L., YANG, C., HUANG, F.-M. & WANG, X.-M. 2021. Landslide susceptibility mapping by attentional factorization machines considering feature interactions. *Geomatics, Natural Hazards and Risk*, 12, 1837-1861.
- LIU, T. 2022. Knowledge tracing: A bibliometric analysis. *Computers and Education: Artificial Intelligence*, 100090.
- LIU, Y., GUO, W., ZANG, D. & LI, Z. A hybrid neural network model with non-linear factorization machines for collaborative recommendation. *China Conference on Information Retrieval*, 2018. Springer, 213-224.
- LONI, B., SHI, Y., LARSON, M. & HANJALIC, A. Cross-domain collaborative filtering with factorization machines. *European conference on information retrieval*, 2014. Springer, 656-661.

- NI, W., LIU, T., ZENG, Q., ZHANG, X., DUAN, H. & XIE, N. Robust factorization machines for credit default prediction. Pacific Rim International Conference on Artificial Intelligence, 2018. Springer, 941-953.
- PAN, J., XU, J., RUIZ, A. L., ZHAO, W., PAN, S., SUN, Y. & LU, Q. Field-weighted factorization machines for click-through rate prediction in display advertising. Proceedings of the 2018 World Wide Web Conference, 2018. 1349-1357.
- PETRONI, F., DEL CORRO, L. & GEMULLA, R. Core: Context-aware open relation extraction with factorization machines. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015. 1763-1773.
- PUNJABI, S. & BHATT, P. Robust factorization machines for user response prediction. Proceedings of the 2018 World Wide Web Conference, 2018. 669-678.
- RENDLE, S., GANTNER, Z., FREUDENTHALER, C. & SCHMIDT-THIEME, L. Fast context-aware recommendations with factorization machines. Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, 2011. 635-644.
- STÜBINGER, J., WALTER, D. & KNOLL, J. 2017. Financial market predictions with Factorization Machines: Trading the opening hour based on overnight social media data. FAU Discussion Papers in Economics.
- SU, R., HUANG, Y., ZHANG, D.-G., XIAO, G. & WEI, L. 2022. SRDFM: Siamese Response Deep Factorization Machine to improve anti-cancer drug recommendation. *Briefings in Bioinformatics*, 23, bbab534.
- SUN, Y., PAN, J., ZHANG, A. & FLORES, A. FM2: field-matrixed factorization machines for recommender systems. Proceedings of the Web Conference 2021, 2021. 2828-2837.
- TOĞUÇ, H. & KUZU, R. S. Hybrid models of factorization machines with neural networks and their ensembles for click-through rate prediction. 2020 5th International Conference on Computer Science and Engineering (UBMK), 2020. IEEE, 31-36.
- VIE, J.-J. 2018. Deep factorization machines for knowledge tracing. *arXiv preprint arXiv:1805.00356*.
- VIE, J.-J. & KASHIMA, H. Knowledge tracing machines: Factorization machines for knowledge tracing. Proceedings of the AAAI Conference on Artificial Intelligence, 2019. 750-757.
- WANG, S., ZHOU, M., FEI, G., CHANG, Y. & LIU, B. 2018. Contextual and position-aware factorization machines for sentiment classification. *arXiv preprint arXiv:1801.06172*.
- WANG, X., HU, G., LIN, H. & SUN, J. A novel ensemble approach for click-through rate prediction based on factorization machines and gradient boosting decision trees. Asia-Pacific Web

- (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data, 2019. Springer, 152-162.
- WANG, Y., SHANG, W. & LI, Z. The application of factorization machines in user behavior prediction. 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016. IEEE, 1-4.
- WEN, N. & ZHANG, F. 2020. Extended Factorization Machines for Sequential Recommendation. *IEEE Access*, 8, 41342-41350.
- XIAO, J., YE, H., HE, X., ZHANG, H., WU, F. & CHUA, T.-S. 2017. Attentional factorization machines: Learning the weight of feature interactions via attention networks. *arXiv preprint arXiv:1708.04617*.
- XIN, X., CHEN, B., HE, X., WANG, D., DING, Y. & JOSE, J. M. CFM: Convolutional Factorization Machines for Context-Aware Recommendation. *IJCAI*, 2019. 3926-3932.
- YANG, Y. & ZHAI, P. 2022. Click-through rate prediction in online advertising: A literature review. *Information Processing & Management*, 59, 102853.
- YU, H. & YIN, J. 2022. Deep Reinforcement Factorization Machines: A Deep Reinforcement Learning Model with Random Exploration Strategy and High Deployment Efficiency. *Applied Sciences*, 12, 5314.
- YU, Y., WANG, Z. & YUAN, B. An Input-aware Factorization Machine for Sparse Prediction. *IJCAI*, 2019. 1466-1472.
- YUAN, F., GUO, G., JOSE, J. M., CHEN, L., YU, H. & ZHANG, W. Boostfm: Boosted factorization machines for top-n feature-based recommendation. *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, 2017. 45-54.
- ZHANG, J., WU, Z., LI, F., LI, W., REN, T., LI, W. & CHEN, J. Deep Attentional Factorization Machines Learning Approach for Driving Safety Risk Prediction. *Journal of Physics: Conference Series*, 2021. IOP Publishing, 012007.
- ZHANG, W., DU, T. & WANG, J. Deep learning over multi-field categorical data. *European conference on information retrieval*, 2016. Springer, 45-57.
- ZHANG, X. & LI, L. Attentional Neural Factorization Machines for Knowledge Tracing. *International Conference on Knowledge Science, Engineering and Management*, 2021. Springer, 319-330.
- ZHU, M., AGGARWAL, C. C., MA, S., ZHANG, H. & HUAI, J. Outlier detection in sparse data with factorization machines. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017. 817-826.

Chapter 2

**Article title: Advantages of using factorisation machines
as a statistical modelling technique**

Authors:

E. Slabber, T.Verster and R. de Jongh

The article was published in the South African Statistical Journal 2021, Vol. 55, No. 2,
125-144.

ADVANTAGES OF USING FACTORISATION MACHINES AS A STATISTICAL MODELLING TECHNIQUE

Erika Slabber¹

Centre for BMI, North-West University, Potchefstroom, South Africa
e-mail: *erika.slabber@nwu.ac.za*

Tanja Verster

Centre for BMI, North-West University, Potchefstroom, South Africa
e-mail: *tanja.verster@nwu.ac.za*

Riaan de Jongh

Centre for BMI, North-West University, Potchefstroom, South Africa
e-mail: *riaan.dejongh@nwu.ac.za*

Factorisation machines originated from the field of machine learning literature and have gained popularity because of the high accuracy obtained in several prediction problems, in particular in the area of recommender systems. This article will provide a motivation for the use of factorisation machines, discuss the fundamentals of factorisation machines, and provide examples of some applications and the possible gains by using factorisation machines as part of the statistician's model-building toolkit. Data sets and existing software packages will be used to illustrate how factorisation machines may be fitted and in what context it is worth being used.

Keywords: Factorisation machines, Linear regression, Machine learning, Recommender systems.

1. Introduction

Factorisation machines (FMs) are extensions of linear models and can model all the higher-order interactions between predictor variables in a very efficient way. It can be applied to a variety of prediction and classification tasks and present an alternative method for well-known statistical modelling techniques such as regression, logistic regression and regularised versions thereof. FMs were introduced by Rendle (2010) and most of the subsequent research appeared in the machine learning literature. A review of the literature reveals some benefits and drawbacks of FMs. Benefits include simpler models (less parameters are used), fast computation (caused by approximating interaction terms in a linear fashion), automatic feature engineering (since all variables and interactions are considered) and the estimation of parameters in a very sparse and high-dimensional data setting. A drawback is that it may underfit dense data since the parameter space is constrained. The interested reader is referred to Rendle (2010) and McClory (2018) for more information on the benefits and drawbacks. In the machine learning literature, FMs have gained popularity because of spectacular

¹ Corresponding author.

MSC2020 subject classifications. 62H99.

successes obtained in the area of recommender systems. Recommender systems usually deal with the problem of predicting a categorical or interval response variable with categorical predictor variables. Often these variables are nominal, with many levels that lead to a high-dimensional sparse data setting, an area in which FMs thrive.

In this paper we will investigate the above claims and, compared to the above-mentioned statistical techniques, illustrate the possible gain of using FMs. We will do this by the use of examples in selected application areas. In order to fit FMs, optimisation algorithms and software implementations thereof are required. The practical implementation of these software routines is not straightforward and can be complex when categorical variables are present. In the various examples studied in this paper these difficulties will be highlighted and solutions provided. Lastly, since the statistical literature is relatively silent on FMs as a modelling technique, we hope that this paper will be able to convince practising statisticians to include this technique in their modelling toolkit.

The layout of this paper is as follows. In the next section, a short background covering the origination of FMs and the successes obtained in the area of recommender systems are provided by means of a literature review. In Section 3, FMs are defined and explained using familiar regression notation. Some guidelines for the practical implementation of these models are included in this section. Section 4 contains a discussion of the algorithms and software routines for fitting these models to data. In Section 5, the practical application of the software is illustrated empirically. A well-known recommender system data set, as well as simulated data sets from regression and logistic regression applications will be used to illustrate the advantage of using FMs in a large data set context. Since the application of these algorithms is not straightforward, software code will be provided, and the performance of the algorithms compared. Some concluding remarks and suggestions for future research are given in the last section.

2. Background on factorisation machines

When Rendle (2010) introduced the machine learning community to FMs, he motivated the method as one that combines the advantages of support-vector machines with those of factorisation models. Support-vector machines and factorisation models, such as matrix factorisation (Srebro and Jaakkola, 2003), tensor factorisation (Tucker decomposition; Tucker, 1966), parallel factor analysis (Harshman, 1970) and pairwise tensor factorisation (Rendle and Schmidt-Thieme, 2010) are frequently used for prediction tasks in machine learning and intelligent information systems. One well-known example of matrix factorisation is singular value decomposition (SVD) where SVD is interpreted as a method to compute the low-rank approximation of a matrix by minimising the squared error loss. As will become clear later, matrix factorisation is a key concept in FMs, which have now surpassed the successes of factorisation models as recommender systems. Similar to support-vector machines (Steinwart and Christmann, 2008), FMs are general predictors working with any real-valued feature vector, but in contrast to support-vector machines, they model all interactions between variables using factorised parameters. Furthermore, FMs are able to estimate interactions even in problems with huge sparsity (such as recommender systems) where support-vector machines fail, as stated by Rendle (2010). Therefore, Rendle (2010) claimed that FMs combine the power of factorisation models with that of feature engineering, which is desirable in a model-building context and especially in recommender systems where the data are typically sparse and the number of predictor variables

frequently more than the number of observations.

In a very general way, recommender systems are concerned with suggesting relevant products to users. For example, Netflix uses recommender systems to advise their clients on the next best movie to watch (based on their viewing preferences); Amazon uses these systems to recommend to clients which products to consider buying next (based on their buying behaviour); and other technology companies such as Google and YouTube use these systems to advise their clients in similar ways. Of course, the end goal for all companies is to become more profitable by selling more products and by growing their client base. Using recommender systems that can accurately advise clients and convince them to buy more products can therefore be a highly effective marketing tool for companies. Today, because of the wide availability of customer information, recommender systems have become one of the most rapidly growing branches of artificial intelligence. The growth of available customer information is attributable to the wide accessibility of recorded data on customer behaviour as well as an exponential growth in the number of characteristics or variables that can aid in increasing the prediction accuracy of recommender systems. Often the number of variables is so large that they overshadow the number of data points recorded. Building recommender systems for problems where the number of variables outnumbers the data points (also referred to as sparsity) gave rise to the introduction of FMs by Rendle (2010). Because factorisation machines can potentially model the high-dimensional interactions between the predictor variables and incorporate these into the prediction model, it is claimed that FMs cater for feature engineering as well as modelling.

FMs achieved numerous rewards in a number of data science challenges (see Parsons, 2017), for example, in the KDD Cup, an annual data mining and knowledge discovery competition organised by ACM Special Interest Group, FMs achieved first place in 2010 in the Grockit Challenge for predicting the next exam question, first place in 2011 for rating prediction on the MovieLens data set, and third place in 2012 for click prediction. FMs also achieved first place in the Kaggle ‘Blue Book for Bulldozers’ challenge for predicting the auction sale price. For more information on recommender systems, see Bhatnagar (2016).

3. Factorisation machines in a regression context

One of the earliest and best known model-building tools in statistics is regression. The technique has been used both in a descriptive and prediction context with a well-established statistical theory of inference. Interestingly, in some machine learning literature, regression is often considered as part of the ‘machine learning’ model-building toolkit, but mostly in the context of minimising the mean squared error (not maximum likelihood) (Rendle, 2010). Machine learning is all about the model and how to fit the model to data in a prediction context. So the loss function (usually mean squared error), and the optimisation method to minimise the loss function are of cardinal importance. In most machine learning models (FMs, neural networks, etc.) inference (e.g. the significance tests about model parameters) is largely absent. Although regression, specifically linear regression, has served the statistical community well, it has difficulty dealing with modelling all higher-order interactions between predictor variables. In fact, when the number of predictor variables is large, the current methods even struggle with modelling only the two-factor interactions, see Yurochkin et al. (2017), and are unable to fit a model where the number of predictor variables is more than the number of observations. Modelling higher-order interactions results in a combinatorial explosion that renders

the fitting of regression models inefficient, especially in a sparsity context. In the remainder of this section, it is shown how FMs become a very useful alternative.

As in the regression case, FMs are concerned with predicting a response Y from some input data X assuming a relationship of the form $Y = f(X)$ with a parametrically specified f for which the parameter values are to be estimated. The prediction is $\hat{Y} = \hat{f}(X)$. As stated previously, regression and support-vector machines encounter difficulties when the input data X are very sparse, a condition that is often encountered when the input consists mainly of large-sized categorical variables, as in recommender systems. Refer to Hastie et al. (2015) for more information. Rendle (2010) introduced FMs, which posit a very specific relationship between the explanatory data X and the response variable of interest, Y . The claim is that the FM generalises the usual models and can also handle very sparse data more successfully. Below a rather simplistic explanation of the rationale underlying the FM model is given. The linear regression prediction model is

$$Y = \beta_0 + \sum_{k=1}^K \beta_k X_k,$$

where Y is the response variable, β_0 the intercept, β_k the coefficient for the k th predictor variable, and X_k the k th predictor variable (where $k = 1, \dots, K$). In total, $K + 1$ parameters have to be estimated in the linear regression model. Typically, interactions between feature variables (X_k) have to be taken into account in order to get predictions of acceptable quality. If the linear model structure is to be preserved, the extension of the linear regression model is as follows:

$$Y = \beta_0 + \sum_{k=1}^K \beta_k X_k + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \beta_{kj} X_k X_j, \quad (1)$$

which includes all possible two-way interactions. The model (1) will be referred to as the regression model with two-way interactions. Note that an additional $K(K - 1)/2$ parameters β_{kj} have to be estimated. Thus the total number of parameters (P) is given by $P = K + 1 + K(K - 1)/2$. Clearly this could lead to a “small N (with N the number of observations), large P ” situation where ordinary least squares regression becomes unworkable when the X -matrix is singular and its inverse does not exist. This happens when the number of parameters to be estimated, P , is more than the number of observations. One approach to avoid this is to use a nonlinear model:

$$\begin{aligned} Y &= \beta_0 + \sum_{k=1}^K \beta_k X_k + \left(\sum_{k=1}^K \beta_k X_k \right)^2 \\ &= \beta_0 + \sum_{k=1}^K \beta_k X_k + \sum_{k=1}^K \beta_k^2 X_k^2 + 2 \sum_{k=1}^{K-1} \sum_{j=k+1}^K \beta_k \beta_j X_k X_j. \end{aligned}$$

By removing the “purely quadratic” terms and the factor 2, this leads to

$$Y = \beta_0 + \sum_{k=1}^K \beta_k X_k + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \beta_k \beta_j X_k X_j.$$

This model contains no extra parameters — the β_{kj} have been factorised into simple products $\beta_k \beta_j$ of the original linear model parameters. This would generally be too restrictive a model. The

interaction β_{kj} can be better approximated by a sum of simple products consisting of G factors:

$$\beta_{kj} = \sum_{g=1}^G v_{kg} v_{jg}. \quad (2)$$

Accordingly, Rendle (2010) puts forward the following extension:

$$Y = \beta_0 + \sum_{k=1}^K \beta_k X_k + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \sum_{g=1}^G v_{kg} v_{jg} X_k X_j, \quad (3)$$

with $(K \times G)$ “interaction parameter matrix” $\{v_{kg}\}$. Typically, G is chosen to be small, say 2 or 3, in which case the number of additional interaction parameters, $(K \times G)$, may be substantially less than $K(K-1)/2$, the number required in a regression model including all two-way interactions. As described by Freudenthaler et al. (2009), the formulation (3) replaces the two-way interaction effects by their factorised analogues as given by (2). Note that it is well known that for any $(K \times K)$ positive definite matrix $\beta = \{\beta_{kj}\}$ there exists a $(K \times G)$ matrix $V = \{v_{kg}\}$ such that $\beta = VV^T$, provided that G is sufficiently large. Refer to Srebro et al. (2005). This shows that an FM can express any interaction matrix $\{\beta_{kj}\}$ if G is large enough. In general, (2) implies that the data for one interaction assist in the estimation of parameters for related interactions, making the model-fitting process more time efficient. In high-dimensional sparse-data settings, like the classic movie recommender system example (see Section 4.1), it is challenging, if not impossible, to estimate interactions between variables independently. FMs break this independence by the factorisation of higher-order interactions which enable efficient coefficient estimation (Rendle, 2010). FMs can handle sparse data quite successfully due to the fact that interactions appear as products $v_{kg} v_{jg}$, hence allowing the estimation of each component to “borrow strength” from other components. Although we will not consider higher order interactions in this paper, it is sufficient to say that FMs can model d -way variable interactions, where d is the polynomial order. Note that according to Harris (2015), in all the current implementations the order has been fixed at two. This is still the case for the implementations used in this paper (LibFM and PROC FACTMAC - see Section 4). The formulation in (3) easily generalises to a model for d -way interactions (see Blondel et al. (2016) for more information).

Returning to the FM formulation in (3) it should be clear that the choice of the number of factors is of practical concern. Rendle (2010) recommended that a small G (2 or 3) should be used when working with sparse data since there are usually not enough data to estimate complex interactions. If the number of factors is restricted, it leads to better prediction and improved interaction matrices under sparsity. As mentioned before, when K variables are used it will result in $(K)(K-1)/2$ two-way interaction terms. Should a value of G be chosen that is too high, the FM model will not provide a gain in terms of modelling since we will end up with the same number of parameters as when a traditional linear regression model with interactions is fitted. G should thus be chosen such that $K(K-1)/2 > KG$. For example in (3), when K is set to 500, it will result in 124 750 interaction terms that have to be estimated, while a choice of $G = 10$ will result in estimating only 5 000 interaction parameters (instead of 124 750). At this stage, no mechanism has been developed to assist with the choice of G , so it remains rather arbitrary and practitioners will have to evaluate this based on their particular applications. However, Rendle (2010) and others recommend the use

Table 1. Number of interaction parameters to estimate at given choices of K and G .

K	$K(K-1)/2$	KG		
		$G = 2$	$G = 3$	$G = 4$
3	3	6	9	12
4	6	8	12	16
5	10	10	15	20
6	15	12	18	24
7	21	14	21	28
8	28	16	24	32
9	36	18	27	36
10	45	20	30	40
11	55	22	33	44
12	66	24	36	48
13	78	26	39	52
14	91	28	42	56
15	105	30	45	60

of a smaller G since it leads to better generalisation, especially under sparsity conditions. Table 1 illustrates the choices available to the modeller when $K = 3, 4, \dots, 15$ and $G = 2, 3, 4$. Clearly, when $K = 3, 4, 5$ it makes no sense to use FM (4) instead of the regression model (2) since there is no gain in factorising the interaction terms. However, when $K = 15$ and $G = 2$, the benefit of using the FM is clear since far fewer parameters need to be estimated (30 under FM (4) instead of 105 under regression model (2)). The shaded area in the table indicates where the FM comes into its own.

As stated before, high-dimensional problems and sparsity are the feeding areas for FMs that excel in this environment as they not only deal with a large number of variables but also with their higher-order interactions. Because of this, Rendle claimed that FMs have the ability to combine automatic feature engineering with the benefits of factorisation models. A key aspect of the modelling process is to do proper encoding of the categorical variables (e.g. binary, nominal and ordinal) and to formulate the prediction problem in such a way that it can logically be translated into a predictor variable matrix (referred to as the X -matrix in this paper) that is encoded in an appropriate way. Many papers about proper encoding exist in the machine learning literature, as do encoding techniques such as one-hot coding, ordinal coding, label coding, hashing, leave-one-out coding, sum coding, Helmert coding, polynomial coding, backward difference coding and binary coding (see, for example, Potdar et al., 2017; Hancock and Khoshgoftaar, 2020). Often the encoding method requires a proliferation in the number of predictor variables. For example, one nominal variable with 10 levels using dummy variables will result in an extra 8 predictor variables in the X -matrix.

4. Fitting factorisation machines

Assume we have a training data set $(Y_n, X_{n,1}, \dots, X_{n,K})$, $n = 1, \dots, N$. The parameters in model (3) are estimated by minimising some distance measure between Y_n and the predicted value $\hat{Y}_n$ generated

from model (3). Let $\ell(Y, \hat{Y})$ denote a typical loss function or distance measure. We then have to minimise

$$\frac{1}{N} \sum_{n=1}^N \ell(Y_n, \hat{Y}_n).$$

For continuously distributed responses, squared error loss is typically used, where

$$\ell_s(Y, \hat{Y}) = (Y - \hat{Y})^2.$$

Using this loss function in the above expression gives the well-known mean squared error (MSE) distance measure. Note that the root mean squared error (RMSE) is sometimes used instead of MSE, where $RMSE = \sqrt{MSE}$.

For binary responses $Y_n = +1$ or -1 , we can either use hinge loss,

$$\ell_H(Y, \hat{Y}) = \max\{0, 1 - Y\hat{Y}\},$$

or logit loss,

$$\ell_L(Y, \hat{Y}) = \frac{\log(1 + \exp(-Y\hat{Y}))}{\log 2}.$$

See Rosasco et al. (2004) for more detail.

Several FM software implementations currently exist, for example the software package LibFM developed by Rendle (2012), PROC FACTMAC in SAS® Visual Data Mining and Machine Learning (SAS Viya), libFMexe in R (a wrapper of the LibFM package that calls the LibFM executable) and fastFM in Python. In this paper, only LibFM and PROC FACTMAC will be used to demonstrate how FMs may be fitted to a data set. LibFM is the most flexible of the two algorithms since it can handle both regression and classification problems. More specifically, LibFM implements both MSE and logit loss and potentially caters for categorical and continuous predictor and response variables. On the other hand, PROC FACTMAC only implements RMSE and only caters for classification problems for an interval-scale response variable and nominal categorical predictor variables. This currently puts a restriction on the type of problems that PROC FACTMAC can handle. A limitation of LibFM is that the output provided is restricted to the predicted values of the response variable, while PROC FACTMAC provides the predicted response values as well as the estimated coefficients of the predicted variables and associated factors. In conclusion, while both algorithms can be used to fit FM models to typical recommender system type data sets, LibFM can also be used in a regression and logistic regression context.

As far as algorithms for the minimisation of the loss function are concerned, both LibFM and PROC FACTMAC implement stochastic gradient descent (SGD) optimisation, while only LibFM gives the user the option to use three further algorithms, namely adaptive stochastic gradient descent (SGDA), alternating least squares (ALS), and Markov Chain Monte Carlo (MCMC). The algorithms implemented in LibFM are described in detail in Rendle (2012) and the algorithm used in PROC FACTMAC in SAS Institute Inc. (2019b). Note that both LibFM and PROC FACTMAC have adapted the above-mentioned loss functions by including regularisation of the model parameters. The reason for this is that if the number of factors is large, many parameters have to be estimated, which makes FMs prone to overfitting. PROC FACTMAC implements a max-norm regularisation of the factor parameters, while LibFM implements an L2 regularisation of all model parameters. Furthermore,

LibFM provides the user with the option to group the model parameters and specify individual regularisation parameters for different parts of the model. All the above-mentioned algorithms require initial values for the so-called hyper parameters that are inputs to the optimisation algorithms. Although default values are supplied, the modeller has to be very careful of the algorithms getting stuck in local optima. This is why experimentation with these inputs is advised. PROC FACTMAC circumvents this problem by choosing the input values based on a grid search combined with a genetic algorithm.

Rendle (2012) claims that “even though factorisation models have a high prediction quality in many applications, it is non-trivial to work with them.” For each problem that cannot be described with categorical variables (as in PROC FACTMAC), a new specialisation model has to be derived and a learning algorithm has to be developed and implemented. This is very time-consuming, error-prone and only applicable for experts in FM models. Although LibFM goes a long way to alleviate this problem, the previous remark by Rendle is certainly true and it took the authors of this paper considerable time to learn how to use the software packages when fitting FM models to various data sets. In his paper, Rendle recommends that the inexperienced user should rather choose the MCMC algorithm, since it determines the regularisation values automatically, while initial estimates for this have to be provided for the other LibFM algorithms. Rendle (2012) gives the following practical advice for applying LibFM to prediction problems:

- For inexperienced users, use MCMC since it is most simple to work with and only requires one input value, i.e. the standard deviation. The standard deviation refers to the standard deviation of the normal distribution that is used in the MCMC algorithm to generate the initial values of the factor parameter matrix V ; see (3).
- When a predictive model for a new data set is being built, start with a low factorisation dimensionality and first determine the standard deviation for initialisation, because intelligently chosen values will speed up the MCMC sampling. Refer to Rendle (2012) for more information.
- Several values for the standard deviation should be tested (e.g. 0.1, 0.5, 0.9). The success can be seen quickly on the first few iterations by monitoring the training error.
- After determining an appropriate initial value for the standard deviation, MCMC can be run with a large number of iterations and larger factorisation dimensionality. The output and convergence can be monitored by the output of LibFM.

Remark. It should be noted that SAS® is still in the process of developing all the different features of PROC FACTMAC. As stated above, one of the limitations of PROC FACTMAC is that only nominal predictor variables and interval response variables are allowed as input (SAS Institute Inc., 2017). A new update of PROC FACTMAC was recently released that can now handle interval predictor variables as well, but for some reason at least two nominal predictor variables have to be included (SAS Institute Inc., 2019a). So, at this stage the application domain of PROC FACTMAC is quite restricted when compared to LibFM. However, an advantage of PROC FACTMAC over LibFM is that it automatically encodes nominal variables.

5. Empirical examples

The objective of this section is to give an overview of how to use the LibFM and PROC FACTMAC packages to fit FMs to data sets in different settings including a recommender system, a regression setting and a logistic regression setting. In the first example, the well-known MovieLens data set (developed by the GroupLens project at the University of Minnesota and freely available on the internet — see Harper and Konstan, 2015) will be used to demonstrate how both packages may be used to fit an FM as a recommender system where predictor variables are typically nominal or categorical. The results from these two implementations are then compared to ordinary least squares (OLS) regression and the LASSO. Then in the next example, a simulated regression data set is used to demonstrate how LibFM may be used to fit an FM to a regression data set that typically consists of continuous predictor variables. Note that, at the moment, PROC FACTMAC cannot be applied on a data set containing continuous predictor variables. In the last example, logistic regression data was simulated and the performance of PROC LOGISTIC compared to that of LibFM. Some performance comparisons are made to compare the various algorithms.

5.1 Recommender system

The well-known and well-analysed MovieLens data set is used in this example to demonstrate a relatively simple application of a recommender system. In particular, we will show how a seemingly straightforward and simple data set can become a tricky prediction problem due to the nature of the data. The data set was collected from companies that provide movies for online viewing — see Silva and Wright (2017). It consists of ratings that users (viewers) give for items (movies) that they have viewed. Suppose there are three users and four items. The rating matrix could then look like the one in Table 2.

The objective is to build a recommender system that will be able to predict the unknown ratings (indicated by ‘?’) so that companies would know which movie should be recommended to which user next. Although users ($U1, U2, U3$) and items ($I1, I2, I3, I4$) are indexed here, users will typically be uniquely identified by a user number or ID, and movies by a specific name. Clearly the users and items are nominal variables and the rating an interval-scale variable on a scale of 1 to 5, where 5 indicates the highest preference. Therefore the natural way to represent this data set for input into an FM is to follow so-called one-hot encoding (i.e. 0/1 indicator variables) as indicated in Table 3.

Since the values of the predictor variables $U1, U2, U3, I1, I2, I3$ and $I4$ are indicator variables,

Table 2. Rating matrix.

	$I1$	$I2$	$I3$	$I4$
$U1$	2	3	1	?
$U2$	?	5	?	1
$U3$	?	?	4	1

Table 3. Transformed X -matrix using one-hot encoding.

Obs	$U1$	$U2$	$U3$	$I1$	$I2$	$I3$	$I4$	Y
1	1	0	0	1	0	0	0	2
2	1	0	0	0	1	0	0	3
3	1	0	0	0	0	1	0	1
4	0	1	0	0	1	0	0	5
5	0	1	0	0	0	0	1	1
6	0	0	1	0	0	1	0	4
7	0	0	1	0	0	0	1	1

the algorithms will estimate the model parameters for each level of the variables (see the discussion of the PROC FACTMAC output later in this section). The MovieLens data set is given in several sizes and for the purposes of this illustration we will use the 100K data set. For a more detailed discussion of this data set refer to Candillier et al. (2007) and Mondal (2018).

The MovieLens 100K data set comprise of 100 000 ratings completed by 943 users for 1 682 movies. The input X -matrix is clearly very sparse since it contains only 100 000 ratings out of a possible 1 586 126 ($943 \times 1\,682$) and, using one-hot encoding, a number of 2 625 ($943 + 1\,682$) ‘new’ predictors. The extent of the sparsity is due to the fact that not all users rate all the movies, and the considerable increase in predictors due to the coding of the nominal variables.

In the next subsections it is shown how the LibFM and PROC FACTMAC packages may be used to achieve this and in the last subsection a comparison is made of the performance of the algorithms.

LibFM

To illustrate the format of files required in LibFM, an illustrative example, as given in the LibFM user manual, is explained before moving to the MovieLens data. The LibFM implementation supports two file formats for input data: a text format and a binary format. Since the text format is easier and recommended for new LibFM users, it was the format used in this example. The first few observations of an example data set are given in Table 4.

When text format is used, the user needs to ensure that each row states first the value Y and then the nonzero values of X as given in Exhibit 1. The response of each of the three cases is thus 4 for the first, 2 for the second and -1 for the third. After the response, each line contains the nonzero elements of X , where an entry like $0:1.5$ reads $X_0 = 1.5$ and thus $X_3 = -7.9$, etc. The left side of INDEX:VALUE states the index within X whereas the right side states the value of $X_{\text{INDEX}} = \text{VALUE}$. Once the data are in this format, an FM model can be developed in LibFM.

Additionally, in recommender systems a file format like *userid*, *itemid* and *rating* are typically used. When such models are developed in LibFM, a two-step process is followed. Start by using a Perl script to convert such data sets to the LibFM file format (this is included in the script’s directory when LibFM is installed). It is important to note that Perl needs to be installed before the conversion script can be executed in LibFM. An extract for the sorted 100K MovieLens data used is given in Table 5.

The code as given in Exhibit 2 is used to convert a recommender system file to LibFM format. Note that this code specifies that the response values are found in column 2 (numbering starts at 0) and that the TimeStamp variable be dropped.

Exhibit 3 gives an extract of the resulting LibFM file ready to be used in model development. This shows that user 1 gave a rating of 5 for movie 1, 3 for movie 2 and 4 for movie 3. The last entry for

Table 4. Extract of sample data set.

Y	X_0	X_1	X_2	X_3	X_4	X_5	X_6
4	1.5	0	0	-7.9	0	0	0
2	0	10^{-5}	0	2	0	0	0
-1	0	0	0	0	0	0	1

Table 5. Extract of sorted 100K MovieLens data.

<i>UserID</i>	<i>ItemID</i>	<i>Rating</i>	<i>TimeStamp</i>
1	1	5	874965758
1	2	3	876893171
1	3	4	878542960

Exhibit 1: LibFM data format (text)

```
4 0:1.5 3:-7.9
2 1:1e-5 3:2
-1 6:1
...
```

Exhibit 2: LibFM conversion code

```
./triple_format_to_libfm.pl -in ratings.dat -target 2 -delete_column 3
-separator "\t"
```

Exhibit 3: LibFM converted recommender system file

```
5 0:1 1:1
3 0:1 2:1
4 0:1 3:1
...
3 0:1 272:1
4 273:1 1:1
2 273:1 10:1
...
```

Exhibit 4: Code used to create FM model in LibFM

```
SGD:
libFM -task r -train Ratings.dat -test Ratings.dat -dim '1,1,10'
-iter 1000 -method sgd -learn_rate 0.01 -init_stdev 0.5
-out Predicted.csv
ALS:
libFM -task r -train Ratings.dat -test Ratings.dat -dim '1,1,10'
-iter 1000 -method als -init_stdev 0.5 -out Predicted.csv
MCMC:
libFM -task r -train Ratings.dat -test Ratings.dat -dim '1,1,10'
-iter 1000 -method mcmc -init_stdev 0.5 -out Predicted.csv
SGDA:
libFM -task r -train Ratings.dat -test Ratings.dat -dim '1,1,10'
-iter 1000 -method sgda -learn_rate 0.01 -init_stdev 0.5
-validation Ratings.dat -out Predicted.csv
```

Exhibit 5: Code used to create FM model in PROC FACTMAC

```
cas mysession;
libname mycas cas;
libname MyData "/home/user3/sasuser.viya";

/* Import movielens into SAS */

proc factmac data= mycas.movielens nfactors=10 learnstep=0.01 maxiter=20
outmodel= mycas.factors;
input User Item /level=nominal;
target Rating /level=interval;
output out= mycas.out copyvars=(User Item Rating);
run;
```

user 1 is 3 0:1 272:1. This means that user 1 rated 272 movies, with a rating of 3 for movie 272. The first entry for user 2 is thus 4 273:1 1:1. Note that user 2 is now numbered as 273 (the next available number after the 272 movies rated by user 1). User 2 also watched movie 1 and gave a rating of 4. The second entry for user 2 is 2 273:1 10:1. This means that user 2 didn't rate movies 2 – 9, but gave a rating of 2 to movie 10. Once the file is in the correct format, the LibFM algorithms can be performed by making use of the code given in Exhibit 4. Refer to the LibFM manual for more information on the selection of parameter values used in the code.

SAS PROC FACTMAC

PROC FACTMAC is one of the machine learning procedures that is available in SAS Visual Data Mining and Machine Learning. It is specifically developed to take advantage of the distributed environment that the SAS Viya platform provides. In order to develop an FM model on the MovieLens data in PROC FACTMAC, the code as given in Exhibit 5 can be used. Note that here we can specify whether the variables used are interval- or nominal-scaled variables and has the advantage that it automatically handles the dummy coding for nominal variables.

Performance comparison

This section compares the performance of the LibFM algorithms (MCMC, ALS and SGD) and SAS's PROC FACTMAC on the MovieLens data. Since SAS provides algorithms for fitting OLS and LASSO regression with all two-way interactions, we included these methods in our comparison. SAS PROC GLM was used to calculate the OLS fits. As stated previously, OLS regression becomes unworkable when the X matrix becomes singular. This happens when $P > N$ and/or when very high correlations between the predictor variables are present (the well-known problem of multicollinearity). Naturally, there are also regularised regression methodologies that cater for the low- N high- P problem (like the Ridge and LASSO regression) so it was decided to include the LASSO for comparison purposes. Refer to Hastie et al. (2015) for more information on these models, and for statistical learning with sparsity in general. Note that the LASSO is given as an option in PROC

GLMSELECT together with various regularisation options. In our case, the Schwartz–Bayesian Criterion (SBC) was used throughout.

In this set, two categorical variables, UserID and ItemID, are considered with 943 and 1 682 levels respectively. PROC GLM was performed but an error message occurred stating that GLM cannot handle this number of dummy variables. When the LASSO was performed using PROC GLM SELECT, a solution could not be obtained due to “insufficient memory”.

In Table 6 the MSE of the errors obtained for LibFM and SAS algorithms are given where the maximum number of iterations are set at 20, 1 000 and 10 000 respectively. The number of factors, G , is set to 10 for all implementations. Note that this choice is rather arbitrary, but the value seemed good when the number of new predictors created due to the nominal variables is taken into account, i.e. the selection of G is still small (when compared to the number of variables) as recommended by Rendle (2010). From Table 6, as far as improvement in MSE is concerned, we can conclude that 1 000 iterations is a reasonable setting since not much fitting accuracy is gained by increasing the maximum number of iterations beyond 1 000. The MSE values obtained by all these algorithms are good, with SGD and ALS the better algorithms in this case. However, to get a clear answer to the best algorithm to use, a comprehensive simulation study is required. The run time of the algorithms dramatically increased when the maximum iterations are set at 10 000. This was especially the case with the LibFM algorithms that took 13 minutes to complete, while PROC FACTMAC finished in less than 3 minutes. For the lower settings (20 and 1 000) the run time of all the algorithms were less than 2 minutes.

Box-and-whisker plots of the error distribution of the fits obtained for the algorithms (with the iterations set to 1 000) appear in Figure 1. These plots are graphical representations of some summary statistics. The length of the box represents the interquartile range (the distance between the 25th and 75th percentiles). The symbol in the box interior represents the group mean and the horizontal line represents the group median. The vertical lines issuing from the box extend to the group minimum and maximum values not deemed outlier values. The circles above or below these extended lines are outlier values. Again, the error distributions are similar, with MCMC and PROC FACTMAC slightly wider when comparing the box (observations between the 25th and 75th percentiles).

In conclusion, both the LibFM and FACTMAC packages fit the MovieLens data set well (where OLS and LASSO could not obtain a solution). This clearly indicates that FMs are very successful in the estimation of parameters in a very sparse or high-dimensional setting. Next, we consider a similar study on a regression data set.

Table 6. Comparison of the MSE achieved by the algorithms.

Iterations	LibFM			SAS		
	MCMC	ALS	SGD	FACTMAC	OLS	LASSO
20	0.580	0.463	0.610	0.700	No solution	Insufficient memory
1 000	0.519	0.428	0.425	0.472	No solution	Insufficient memory
10 000	0.520	0.427	0.421	0.467	No solution	Insufficient memory

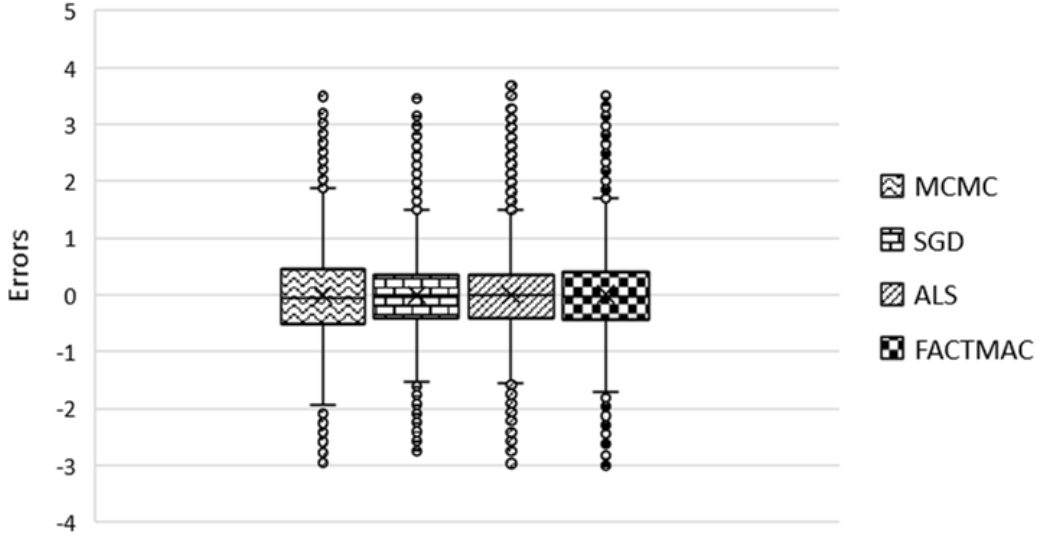


Figure 1. Box-and-whisker plots of the errors obtained by the algorithms.

5.2 Regression example

In order to study the performance of FMs in a regression context, the regression model with two-way interactions as given in (1) was used. Note that all analyses were performed on a laptop computer (i7 processor, 1.8 GHz CPU) unless otherwise specified. Different sized data sets were generated where the X -matrix consisted of predictors that were generated as $N(0, 1)$ variables. The response variable Y was then obtained by assuming that all regression parameters have a value of 1 and by adding an error term from the $N(0, 0.25)$ distribution. Data sets were generated for values of $K = 10, 20, 50, 100, 200$ and $N = 1\,000, 5\,000, 10\,000$ in order to compare the LibFM algorithms with the performance of ordinary least squares (OLS) regression with two-way interactions and LASSO regression. One would expect a good model to obtain (at best) an MSE around 0.25. A value lower than 0.25 indicates that the model tends to overfit and a value higher than 0.25 that the model tends to underfit. SAS PROC GLM was used to calculate the regression fits and PROC GLMSELECT for the LASSO fits. In order to illustrate the performance of the different methods, the results obtained for the $K = 50$ case are given in Table 7, with the run time in seconds given in brackets within the table. The number in brackets after the type of model used indicates the number of parameters.

When $K = 50$ a total of 1 276 ($1 + 50 + 50 \times 49/2$) parameters have to be estimated by GLM and GLMSELECT for both the OLS and LASSO regressions, respectively. For FMs, G was taken as $G = 2, 5, 10$ and 20, resulting in 151 ($G = 2$), 301 ($G = 5$), 551 ($G = 10$) and 1 051 ($G = 20$) coefficients to be estimated by LibFM algorithms SGD, ALS and MCMC. From Table 7 the following conclusions can be drawn:

- Regression with all two-way interactions and the LASSO tend to overfit the data for all sample sizes studied. The extent of overfitting reduces with increasing sample size.
- The ALS and MCMC algorithms produce good fits (close to 0.25) for $G = 2$ at all the sample

Table 7. MSE results and run time of regression data set with 50 predictor variables.

Modelling technique used		Number of observations		
		1 000	5 000	10 000
Regression with 2-way interactions (1 276)		no answer (10.43)*	0.183 (6.62)	0.215 (23.16)
LASSO (1 276)		$2.97E^{-07}$ (18.17)**	0.185 (30.17)	0.216 (33.15)
LibFM	SGD – 2 factors (151)	0.356 (6.13)	0.408 (11.88)	0.386 (15.98)
	SGD – 5 factors (301)	0.307 (9.43)	0.400 (20.26)	0.379 (30.45)
	SGD – 10 factors (551)	0.108 (12.30)	0.387 (34.87)	0.356 (51.48)
	SGD – 20 factors (1 051)	0.045 (18.95)	0.357 (50.58)	0.332 (70.23)
	ALS – 2 factors (151)	0.246 (6.13)	0.247 (14.73)	0.252 (25.66)
	ALS – 5 factors (301)	0.180 (9.37)	0.232 (26.20)	0.244 (45.22)
	ALS – 10 factors (551)	0.104 (13.86)	0.219 (43.92)	0.235 (84.28)
	ALS – 20 factors (1 051)	0.026 (19.81)	0.204 (76.78)	0.225 (155.99)
	MCMC – 2 factors (151)	0.236 (6.47)	0.263 (9.5)	0.299 (26.75)
	MCMC – 5 factors (301)	0.181 (9.41)	0.241 (16.33)	0.249 (50.31)
	MCMC – 10 factors (551)	0.123 (12.60)	0.230 (29.18)	0.242 (88.20)
	MCMC – 20 factors (1 051)	0.052 (19.80)	0.210 (53.78)	0.231 (157.04)

* Note: The $X'X$ matrix was found to be singular and a generalised inverse was used to solve the normal equations.

** The adaptive weights for the LASSO method are not uniquely determined because the full least squares model is singular.

sizes. However, when the number of factors increases, both methods seem to overfit the data, especially at smaller sample sizes. This conclusion confirms Rendle’s remark that a small number of factors is generally a better choice for model generalisation.

- With FM regression, as the number of factors increases, the MSEs improve but, as expected, the run time is longer. At small choices of G ($G \leq 5$) the run time of the FM algorithms are mostly better than the LASSO.
- Overall in this example we can conclude that the FM algorithms (ALS and MCMC) outperform the other modelling techniques when $G = 2$ since they produce good fits (no overfitting), the run times are similar or less and the number of parameters used in these models is greatly reduced.

In summary, when the number of observations and predictor variables are ‘small’ and $P < N$, OLS and LASSO regression with interaction overfit the data. In this example and for the cases considered here, overfitting was achieved for $K \leq 100$ and $N \leq 10\,000$. However, for the case $K = 200$ and $N = 100\,000$, the OLS regression did not obtain a solution after seven days while the MCMC algorithm with $G = 10$ obtained a solution in about three hours (performance results not shown). The OLS regression was then performed on a server (128GB DDR RAM @ 800MHz

and speed of hard drive is 10k RPM) and obtained a solution within 24 hours. However, the MSE obtained by MCMC was much higher than that obtained by OLS regression, although the latter tends to overfit. Although this cannot be regarded as a conclusive study, the results indicate that FMs do not have a substantial advantage over traditional statistical methods in a dense classical regression context (except with a reduction of the number of parameters required), however it might lead to better generalisation when a small number of factors is chosen. This will have to be determined by conducting a simulation study in a prediction setting. Harris (2015) stated that FMs perform extremely well on sparse data, including data of very high sparsity. But as a trade-off, they do not perform well on dense data so other algorithms are more suited to this class of data, as seen in our analyses with results given in Table 6 and Table 7.

To conclude, in all cases considered here, SAS PROC GLM and GLMSELECT with OLS and LASSO (with all two-way interactions) outperformed the LibFM in terms of accuracy. The LibFM algorithms for small choices of G sometimes have significant speed advantage (fewer parameters to estimate) and may lead to better generalisation. It should be noted that in a big data context the conclusions might be different, since the data sets generated here are very much ‘small-scale data sets’, so the potential benefit of FMs might arise when studying large-scale data sets. This is left for future research.

5.3 Logistic regression example

In finance, logistic regression is frequently used to build credit scorecards for assessing the credit-worthiness of clients. Typically large data sets containing millions of records are used, and the data are split into a training and test set in order to evaluate the predictive power of the scorecards. One issue is that these data sets are typically classified. In order to investigate the performance of FMs in this context, differently sized data sets with $N = 100, 1\,000, 10\,000$ and $100\,000$ were generated by assuming a logistic regression model and by using simulation.

The linear part of the model contained ten predictor variables and all their two-way interactions plus an error term. The predictor variables were independently generated from a standard normal distribution $N(0, 1)$ and the error term from an $N(0, 0.25)$ distribution. The coefficients of the predictor variables and their interactions were assumed to be 1. The binary target variable (Y) is then obtained by applying the logistic link function to the linear term and by comparing the result to a uniformly generated variable. The interested reader is referred to de Jongh et al. (2015) for more information.

The simulated training and test data sets were then augmented by including another five variables (again generated from $N(0, 1)$) with their two-way interactions and the FMs and logistic regression were fitted on these augmented data sets in order to determine how well the true model is estimated in the presence of noise.

The results, by using the Gini as performance measure, are given in Table 8. The Gini statistic measures the discriminatory power of a model with values varying between 0 and 1. Large values correspond to stronger associations between the predicted and observed values. Refer to Allison (2003) for more information.

Note that, because the data sets are generated using a logistic regression model, one would expect that the traditional model will perform better than FMs since the latter use much less model parameters. However, the performance of the FM-based logistic regression is quite surprising:

Table 8. Gini results of logistic regression data sets with ten predictor variables.

	Gini statistic							
	SAS: Proc Logistic		LibFM: MCMC		LibFM: MCMC		LibFM: MCMC	
Number of parameters	55		30		40		50	
Number of factors	N/A		2		3		4	
Data set	Train	Test	Train	Test	Train	Test	Train	Test
$N = 100$	1.0000*	0.1680	1.0000	0.8368	1.0000	0.8333	1.0000	0.7804
$N = 1\,000$	0.9690	0.8643	0.9431	0.9123	0.9468	0.9101	0.9502	0.9079
$N = 10\,000$	0.9405	0.9287	0.9390	0.9312	0.9394	0.9308	0.9396	0.9305
$N = 100\,000$	0.9357	0.9340	0.9355	0.9342	0.9355	0.9342	0.9355	0.9342

* Warning: There is complete separation of data points. The maximum-likelihood estimate does not exist. The LOGISTIC procedure continues in spite of the above warning. Results shown are based on the last maximum-likelihood iteration. Validity of the model fit is questionable.

- As far as the performance on the training sets is considered, traditional logistic regression does slightly better than FM logistic regression. Note that the extent of the outperformance is, as expected, less as the number of FM factors and sample size increase.
- When performance on the test data sets is studied, FM -based logistic regression does better than traditional logistic regression and the extent of the outperformance is more pronounced when the number of factors ($G = 2$) and sample sizes ($N \leq 10\,000$) are smaller.

To conclude, from the three examples studied here, the potential gains of using FMs as modelling technique have been illustrated. These include:

- The efficient estimation of model parameters in a high-dimensional and sparse-data setting (MovieLens example).
- Comparable performance to traditional methods, but much faster computationally due to the fact that fewer parameters and thus simpler models are used.
- An indication that FM-based regression and logistic regression with a small number of factors seem to generalise better, i.e. their prediction performance is better than traditional methods especially at smaller sample sizes.

6. Conclusion and direction for further research

FMs have obtained huge successes in the area of recommender systems. Recommender system data sets are characterised by high dimensionality and sparsity. Thus, problems where the number of predictor variables typically outnumbers the number of observations and when all possible values in the predictor space are considered, many of the observations are ‘missing’ (not observed). As illustrated in this paper, the above comment is clear from our analysis of the MovieLens data (sparse or high-dimensional data). Our analysis shows that both the LibFM and PROC FACTMAC packages fit the MovieLens data set well (where OLS and LASSO could not obtain a solution).

The performance benefits of using FMs in a classical regression setting are less obvious. Our analysis showed that, in a classical regression context, OLS and LASSO outperformed FMs (LibFM) in terms of accuracy but they tend to overfit the data set studied here. The LibFM algorithms for small choices of G sometimes have significant speed advantage (fewer parameters to be estimated) and may lead to better generalisation. When the number of parameters is considered, we can argue the advantage of the use of FM models (LASSO used 1 276 parameters versus FMs from as little as 151 parameters).

When logistic regression models are considered, and the performance of logistic regression models is compared to FM models, we can conclude that FMs clearly outperformed logistic regression models when smaller samples are considered from our analysis (in terms of Gini values obtained on the test sets). As the sample size increases, FMs still slightly outperformed logistic regression models by using fewer parameters.

It should be clear from the list of references given in this paper that the overwhelming majority are in conference proceedings, and a few are in computer science and engineering journals. It is interesting to note that neural networks, a well-known machine learning technique that originated in the 1940s (see e.g. McCulloch and Pitts (1943) and Rosenblatt (1958)), also took some time before it was being included in practising statisticians' model-building toolkit. The first paper on neural networks appeared almost fifty years later in the local statistical journal (see de Jongh and de Wet (1993)). Nowadays, practising statisticians frequently make use of neural networks and we hope that this paper will contribute to the incorporation of FMs in the model-building toolkit of statisticians.

From the nonstandardised use of statistical terminology and the lack of rigorous statistical theory in the literature, it is evident that FM research is still at the beginning of the statistical learning curve. Therefore, future research on FMs presents many opportunities. Apart from the standard statistical questions that can be researched, many open problems exist. For instance, what is the optimum choice for the number of factors for a particular problem? According to Rendle (2010), in sparse settings, typically the number of factors used should be small because there is not enough data to estimate complex interactions. He claims that when the number of factors is restricted it leads to better generalisation and thus improved interaction matrices under sparsity. See Rosasco et al. (2004). These claims should be researched by, among others, comprehensive simulation studies. In the field of model development, computationally efficient, time efficient and robust learning methods on big data and complex interactions are necessary. More advanced modelling aspects such as the distribution of the errors and multi-collinearity issues currently play no role. So research into statistical inference is needed, such as model selection, over- and underfitting, and the effects of outliers.

In conclusion, the successes of FMs warrant the attention of statisticians both for practical application and as a fruitful research area. We hope that statisticians will take up this challenge.

Acknowledgements. The authors would like to thank Prof. Hennie Venter and Prof. Freek Lombard for valuable contributions to the paper and would also like to thank Prof Tertius de Wet for careful reading of the final manuscript. This work is based on research supported in part by the Department of Science and Innovation (DSI) of South Africa. The grant holder acknowledges that opinions, findings and conclusions or recommendations expressed in any publication generated by DSI-supported research are those of the authors and that the DSI accepts no liability whatsoever in this regard.

References

- ALLISON, P. (2003). *Logistic Regression: Using the SAS System*. SAS Institute and Wiley & Sons.
- BHATNAGAR, V. (2016). *Collaborative Filtering Using Data Mining and Analysis*. IGI Global, Hershey, PA.
- BLONDEL, M., FUJINO, A., UEDA, N., AND ISHIHATA, M. (2016). Higher-order factorization machines. *In Advances in Neural Information Processing Systems 29*. Barcelona, Spain.
- CANDILLIER, L., MEYER, F., AND BOULLÉ, M. (2007). Comparing state-of-the-art collaborative filtering systems. *In Machine Learning and Data Mining in Pattern Recognition. MLDM 2007. Lecture Notes in Computer Science*, volume 4571. Springer, Berlin, 548–562.
- DE JONGH, P., DE JONGH, E., PIENAAR, M., GORDON-GRANT, H., OBERHOLZER, M., AND SANTANA, L. (2015). The impact of pre-selected variance inflation factor thresholds on the stability and predictive power of logistic regression models in credit scoring. *ORiON*, **31**, 17–37.
- DE JONGH, P. AND DE WET, T. (1993). An introduction to neural networks. *South African Statistical Journal*, **27**, 103–128.
- FREUDENTHALER, C., SCHMIDT-THIEME, L., AND RENDLE, S. (2009). Factorization machines: Factorized polynomial regression models.
URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.364.8661>
- HANCOCK, J. T. AND KHOSHGOFTAAR, T. M. (2020). Survey on categorical data for neural networks. *Journal of Big Data*, **7**, 1–41.
- HARPER, F. M. AND KONSTAN, J. A. (2015). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, **5**, 1–19.
- HARRIS, B. (2015). Factorization machines: A new way of looking at machine learning.
URL: <https://securityintelligence.com/factorization-machines-a-new-way-of-looking-at-machine-learning/>
- HARSHMAN, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multimodal factor analysis. *UCLA Working Papers in Phonetics*, **16**, 1–84.
- HASTIE, T., TIBSHIRANI, R., AND WAINWRIGHT, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, Boca Raton, FL.
- McCLORY, P. (2018). Factorization machines for machine learning.
URL: <https://medium.com/@pmdev/factorization-machines-for-machine-learning-63b549c86111>
- McCULLOCH, W. S. AND PITTS, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, **5**, 115–133.
- MONDAL, A. (2018). Factorization machines for movie recommendations.
URL: <http://www.stokastik.in/factorization-machines-for-movie-recommendations/>
- PARSONS, N. (2017). Factorization machines for recommendation systems.
URL: <https://getstream.io/blog/factorization-recommendation-systems/>
- POTDAR, K., PARDAWALA, T. S., AND PAI, C. D. (2017). A comparative study of categorical variable encoding techniques for neural network classifiers. *International Journal of Computer Applications*, **175**, 7–9.

- RENDLE, S. (2010). Factorization machines. *In 2010 IEEE International Conference on Data Mining*. Sydney, Australia, 995–1000.
- RENDLE, S. (2012). Factorization machines with LibFM. *ACM Transactions on Intelligent Systems and Technology*, **3**, 1–22.
- RENDLE, S. AND SCHMIDT-THIEME, L. (2010). Pairwise interaction tensor factorization for personalized tag recommendation. *In Proceedings of the Third ACM International Conference on Web Search and Data Mining*. New York, USA, 81–90.
- ROSASCO, L., VITO, E. D., CAPONNETTO, A., PIANA, M., AND VERRI, A. (2004). Are loss functions all the same? *Neural Computation*, **16**, 1063–1076.
- ROSENBLATT, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, **65**, 386.
- SAS INSTITUTE INC. (2017). SAS® Visual Data Mining and Machine Learning 8.2 Procedures. URL: <https://documentation.sas.com/api/docsets/casml/8.2/content/casml.pdf?locale=en>
- SAS INSTITUTE INC. (2019a). An Introduction to SAS® Viya® 3.4 Programming. URL: <https://documentation.sas.com/api/docsets/pgmdiff/3.4/content/pgmdiff.pdf?locale=en>
- SAS INSTITUTE INC. (2019b). SAS® Visual Data Mining and Machine Learning 8.5: Procedures. URL: https://documentation.sas.com/?docsetId=casml&docsetTarget=casml_factmac_syntax07.htm&docsetVersion=8.5&locale=ja
- SILVA, J. AND WRIGHT, R. (2017). Factorization machines: A new tool for sparse data. *In SAS Global Forum*. Orlando, FL.
- SREBRO, N. AND JAAKKOLA, T. (2003). Weighted low-rank approximations. *In Proceedings of the 20th International Conference on Machine Learning*. Washington DC, 720–727.
- SREBRO, N., RENNIE, J. D. M., AND JAAKKOLA, T. S. (2005). Maximum-margin matrix factorization. *In Advances in Neural Information Processing Systems 17*. MIT Press, Cambridge, MA, 1329–1336.
- STEINWART, I. AND CHRISTMANN, A. (2008). *Support Vector Machines*. Springer, New York, NY.
- TUCKER, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika*, **31**, 279–311.
- YUROCHKIN, M., NGUYEN, X., AND VASILOGLOU, N. (2017). Multi-way interacting regression via factorization machines. *In Proceedings of the 31st Conference on Neural Information Processing Systems*. Long Beach, CA, 2595–2603.

Chapter 3

**Article title: Algorithms for estimating the parameters of
factorisation machines**

Authors:

E. Slabber, T.Verster and R. de Jongh

The article was published in the South African Statistical Journal 2022, Vol. 56, No. 2,
69-89.

Algorithms for estimating the parameters of factorisation machines

E. Slabber, T. Verster and P. J. de Jongh

Centre for Business Mathematics and Informatics, North-West University, South Africa

Since the introduction of factorisation machines in 2010, it became a popular prediction technique amongst machine learners who applied the method with success in several data science challenges such as Kaggle or KDD Cup. Despite these successes, factorisation machines are not often considered as a modelling technique in business, partly because large companies prefer tried and tested software for model implementation. Popular modelling techniques for prediction problems, such as generalised linear models, neural networks, and classification and regression trees, have been implemented in commercial software such as SAS which is widely used by banks, insurance, pharmaceutical and telecommunication companies. To popularise the use of factorisation machines in business, we implement algorithms for fitting factorisation machines in SAS. These algorithms minimise two loss functions, namely the weighted sum of squared errors and the weighted sum of absolute deviations using coordinate descent and nonlinear programming procedures. Using a simulation study, the above-mentioned routines are tested in terms of accuracy and efficiency. The prediction power of factorisation machines is then illustrated by analysing two data sets.

Keywords: Factorisation machines, Fitting algorithms, Parameter estimation.

1. Introduction

When Rendle (2010) introduced factorisation machines (FMs), he described it as a new model class that combines the advantages of support-vector machines (SVMs) with those of factorisation models. According to Rendle, FMs model all interactions between predictor variables using factorised parameters that enable FMs to estimate interactions even in problems with huge sparsity where SVMs fail. Although higher order factorisation machines are defined in Rendle (2010), the models include many terms and fitting thereof is problematic (see Rendle, 2012). Therefore, in this paper, we will concentrate on fitting second order factorisation machines only. Yurochkin et al. (2017) describe a second order factorisation machine (FM) as a linear regression model which include all the predictors as well as the approximations of all the second order interactions between the predictors. Application areas of FMs, amongst others, include recommender systems (see e.g. Parsons, 2017), click-through rate prediction (Juan et al., 2016), marketing prediction problems (Juan et al., 2017), and social network prediction problems (Hong et al., 2013). We will now define factorisation machines.

Corresponding author: E. Slabber (erika.slabber@outlook.com)
MSC2020 subject classifications: 62H99.

Suppose we have observations $\{Y_n, X_{n1}, \dots, X_{nK}\}$, $n = 1, \dots, N$, on a target or response variable Y and K predictor variables $X_1, \dots, X_K$. For the moment we will assume that the variable Y is continuous, while the predictors can be of any type, i.e. continuous, discrete, nominal or ordinal. The model equation is given by

$$Y_n = \beta_0 + \sum_{k=1}^K \beta_k X_{nk} + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \langle \boldsymbol{\varphi}_k, \boldsymbol{\varphi}_j \rangle X_{nk} X_{nj} + e_n, \quad (1)$$

where β_0 is the intercept, $\beta_1, \dots, \beta_K$ the regression coefficients, $\boldsymbol{\varphi}_k$, $k = 1, \dots, K$ a G dimensional vector of factor loadings for each variable, and e_n the error term. Here $\langle \boldsymbol{\varphi}_k, \boldsymbol{\varphi}_j \rangle = \sum_{g=1}^G \varphi_{kg} \varphi_{jg}$ denotes the inner product of the vectors $\boldsymbol{\varphi}_k$ and $\boldsymbol{\varphi}_j$. Note that both $\boldsymbol{\beta}$ and $\boldsymbol{\varphi}$ must be estimated and that G is an input parameter for the fitting procedure.

Second order factorisation machines relate to regression with two-way interactions in the following way. The linear regression model with all possible two-way interactions is

$$Y_n = \beta_0 + \sum_{k=1}^K \beta_k X_{nk} + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \beta_{kj} X_{nk} X_{nj} + e_n, \quad (2)$$

where, as before, the β_k are the regression coefficients and the β_{kj} the coefficients of the interactions that need to be estimated. Careful examination of (1) and (2) indicates that β_{kj} in (2) is replaced by $\sum_{g=1}^G \varphi_{kg} \varphi_{jg}$ in (1). Rendle's key idea emanated from the well-known result that for any $(K \times K)$ positive definite matrix $\{\beta_{kj}\}$, there exists a $(K \times G)$ matrix $\{\varphi_{kg}\}$ such that $\{\beta_{kj}\} = \{\varphi_{kg}\} \{\varphi_{jg}\}$, provided that G is sufficiently large. The interaction β_{kj} is therefore approximated by a sum of simple products consisting of G factors, i.e.

$$\beta_{kj} \approx \sum_{g=1}^G \varphi_{kg} \varphi_{jg}. \quad (3)$$

Clearly, by choosing $\beta_{kj} = \sum_{g=1}^G \varphi_{kg} \varphi_{jg}$, (1) is a special case of (2). Note that the FM model in (1) has $1 + K + KG$ parameters (or coefficients) while the regression model (2) has $1 + K + K(K-1)/2$ parameters. Therefore, when $G < (K-1)/2$, an FM requires less parameters to be estimated. When the number of model parameters exceed the number of observations, two-way interaction multiple regression models are impractical to fit (see e.g. Hastie et al., 2015 or Kutner et al., 2005). This is frequently the case when dealing with recommender system problems, which are characterised by many categorical predictor variables, each having many levels (see e.g. Rendle, 2010, and Parsons, 2017). When these variables are encoded as dummy variables, the predictor variable space have many zeros leading to the problem of high sparsity. Here FMs provide an advantage through the careful selection of the number of factors G which is independent of the number of predictors. To summarise, the important difference to regression is that the effect of interaction is not modelled by an independent parameter β_{kj} , but with a factorised parameterisation $\sum_{g=1}^G \varphi_{kg} \varphi_{jg}$, which implies that the pairwise interactions have low rank. This allows FMs to estimate reliable parameters even in highly sparse data where standard models fail.

The linear model (2) may be extended to a generalised linear model (GLM), which allows the modelling of nominal or ordinal target variables by postulating a linear relationship between the

target and predictors, even though their underlying relationship is not linear. This is made possible by using a link function, which links the target variable to a model that is linear in its parameters. Unlike linear regression, the error distribution of the target variable need not be normally distributed but are assumed to follow an exponential family of distributions such as normal, binomial, Poisson or gamma distributions. For example, if the target variable is binary the logistic link function may be used to link the target variable to a linear model including the predictors and their two-way interactions. Such models are easy to fit using PROC GLM in SAS through maximisation of the appropriate log likelihood function. Algorithms for fitting FMs with different loss functions are discussed in Rendle (2012) and incorporated in his LibFM package. Rendle (2012) estimates the model parameters by defining a loss function and then minimising the sum of losses over the observed data. He considered squared error and logit loss, as well as L2 regularisation. LibFM, written in C, uses logit loss that may be used to fit logistic regression FM analogues of GLMs (see Pijnenburg and Kowalczyk, 2017). Note that when the underlying model is not intrinsically linear, care should be taken when interpreting the interaction effects (see Norton et al., 2004). FM extensions of GLM models, such as logistic FMs, will not be considered in this paper, but in a future paper in which we investigate alternative credit scoring modelling techniques.

Apart from LibFM, a number of FM fitting routines are available, such as PROC FACTMAC in SAS® Visual Data Mining and Machine Learning (SAS Viya), *fastFM* in Python (see Bayer, 2016) and *rsparse* an R package for statistical learning on sparse matrices (see Selivanov, 2021). In a previous paper (Slabber et al., 2021), we illustrated the use of *LibFM* and PROC FACTMAC and highlighted some of the shortcomings of these routines. For example, LibFM only outputs the predicted values of the model but not the estimated parameters, while PROC FACTMAC outputs the predicted values and the estimated parameters but has the limitation that it can only handle nominal predictor variables and an interval scaled target variable (see SAS Institute Inc., 2017, 2019). LibFM does not have these limitations and can handle any type of predictor or target variable, while PROC FACTMAC is restricted to recommender system type applications.

To make the application of FMs more accessible to business, we implement our FM fitting routines in SAS, which is widely used, especially by large corporates such as banks, insurance and telecommunication companies. We consider two algorithms for fitting FMs namely coordinate descent (CD) and nonlinear programming (NLP). The CD algorithm is implemented in PROC IML and the NLP algorithm in PROC OPTMODEL of SAS. Two loss functions are considered, namely the weighted sum of squared errors (WSSE) and the weighted sum of absolute deviations (WSAD). While WSSE is the most common loss function used to fit FMs, WSAD provides a robust alternative, since it is less sensitive to outliers. For more information on robust estimation and outlier detection, see for example Rousseeuw and Leroy (2005).

The layout of the paper is as follows. In the next section, for both loss functions, we show how the CD method may be implemented to fit FMs. The NLP implementation of these loss functions in PROC OPT MODEL is straightforward and will be mentioned briefly. In Section 3, the accuracy and efficiency of these routines are tested by means of a simulation study, and then the routines are applied to an artificially generated and real data set. Section 4 contains some concluding remarks and ideas for future research.

2. Fitting factorisation machines

Given estimators $b_0, b_1, \dots, b_K$ and $f_{kg}, k = 1, \dots, K, g = 1, \dots, G$, of the parameters, we denote the corresponding estimator of Y_n by

$$\hat{Y}_n = b_0 + \sum_{k=1}^K b_k X_{nk} + \frac{1}{2} \sum_{g=1}^G \left[\left(\sum_{k=1}^K f_{kg} X_{nk} \right)^2 - \sum_{k=1}^K f_{kg}^2 X_{nk}^2 \right], \quad (4)$$

which is equivalent (see Rendle 2010, 2012 or Slabber et al. 2021) to

$$\hat{Y}_n = b_0 + \sum_{k=1}^K b_k X_{nk} + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \sum_{g=1}^G f_{kg} f_{jg} X_{nk} X_{nj}.$$

We chose to use (4), because it simplified the formulation of the CD algorithm for the loss functions in (5) and (6) below. In the case of least squares, the estimators of Y_n are obtained by minimising the weighted sum of squared errors (WSSE) given by

$$WSSE = \sum_{n=1}^N w_n \left[Y_n - \hat{Y}_n \right]^2 \quad (5)$$

with respect to $b_0, b_1, \dots, b_K$ and $f_{kg}, k = 1, \dots, K, g = 1, \dots, G$. Here $\{Y_n\}$ denotes the data on the target variable Y and $w_1, \dots, w_N$ are non-negative weights that sum to 1. Since it is well-known that least squares estimators are sensitive to outliers (e.g. see Maronna et al., 2019), we also consider a least absolute deviation (LAD) estimator, which minimises the weighted sum of absolute deviations (WSAD). This estimator is obtained by minimising

$$WSAD = \sum_{n=1}^N w_n \left| Y_n - \hat{Y}_n \right| \quad (6)$$

with respect to $b_0, b_1, \dots, b_K$ and $f_{kg}, k = 1, \dots, K, g = 1, \dots, G$. Again $w_1, \dots, w_N$ are non-negative weights which sum to 1.

Remarks

- Note that the loss functions in (5) and (6) are both convex.
- Minimisation of (5) can be done using standard minimisation routines. However, the objective function (6) is non-differentiable making the minimisation problem difficult.
- In the equally weighted case ($w_i = N^{-1}$ for all $i = 1, \dots, N$), WSSE becomes mean squared error (MSE) and WSAD becomes mean absolute deviation (MAD).
- The weights w_i could be defined to produce bounded-influence type estimators by using, for example, Mallows weights (see de Jongh et al., 1988).
- The idea of robust factorisation machines is not new, and some researchers have explored the idea (see e.g. Punjabi and Bhatt, 2018 and Liu et al., 2019), but we are not aware of an implementation of the LAD estimator considered here.

2.1 CD algorithm for minimising WSSE

Suppose we have current estimates $b_0, b_1, \dots, b_K$ and $f_{kg}, k = 1, \dots, K, g = 1, \dots, G$, which we wish to update via the CD algorithm. We begin with the intercept or bias term. Write

$$\widehat{Y}_n(0) = \sum_{k=1}^K b_k X_{nk} + \frac{1}{2} \sum_{g=1}^G \left[\left(\sum_{k=1}^K f_{kg} X_{nk} \right)^2 - \sum_{k=1}^K f_{kg}^2 X_{nk}^2 \right] = \widehat{Y}_n - b_0, \quad (7)$$

that is, $\widehat{Y}_n(0)$ is (4) without the intercept term. The updated version of the intercept, say $\widetilde{b}_0$, is obtained by minimising

$$WSSE = \sum_{n=1}^N w_n \left[Y_n - \widehat{Y}_n(0) - \widetilde{b}_0 \right]^2 \quad (8)$$

w.r.t. $\widetilde{b}_0$. The choice of $\widetilde{b}_0$ that minimises this expression is

$$\widetilde{b}_0 = \sum_{n=1}^N w_n [Y_n - \widehat{Y}_n(0)] = \sum_{n=1}^N w_n [Y_n - \widehat{Y}_n + b_0] = b_0 + \sum_{n=1}^N w_n [Y_n - \widehat{Y}_n]. \quad (9)$$

Next consider b_j and write

$$\widehat{Y}_n(j) = b_0 + \sum_{k \neq j}^K b_k X_{nk} + \frac{1}{2} \sum_{g=1}^G \left[\left(\sum_{k=1}^K f_{kg} X_{nk} \right)^2 - \sum_{k=1}^K f_{kg}^2 X_{nk}^2 \right] = \widehat{Y}_n - b_j X_{nj}, \quad (10)$$

so $\widehat{Y}_n(j)$ is (4) without $b_j X_{nj}$.

Minimising

$$WSSE = \sum_{n=1}^N w_n \left[Y_n - \widehat{Y}_n(j) - \widetilde{b}_j X_{nj} \right]^2 \quad (11)$$

w.r.t. $\widetilde{b}_j$, we obtain the updated version of the j -th regression coefficient $\widetilde{b}_j$. To minimise this with respect to $\widetilde{b}_j$, note that we have a regression problem in which the X_{nj} are regressed on the residuals $Y_n - \widehat{Y}_n(j) = Y_n - \widehat{Y}_n + b_j X_{nj}$ and the minimising choice of $\widetilde{b}_j$ is given by

$$\widetilde{b}_j = b_j + \frac{\sum_{n=1}^N w_n [Y_n - \widehat{Y}_n] X_{nj}}{\sum_{n=1}^N w_n X_{nj}^2}. \quad (12)$$

Next, we consider f_{jh} . To isolate the contribution of f_{jh} to (5) is somewhat more complicated but manipulation leads to

$$\begin{aligned} \widehat{Y}_n = b_0 + \sum_{k=1}^K b_k X_{nk} + \frac{1}{2} \sum_{g \neq h}^G \left[\left(\sum_{k=1}^K f_{kg} X_{nk} \right)^2 - \sum_{k=1}^K f_{kh}^2 X_{nk}^2 \right] \\ + \frac{1}{2} \left[\left(\sum_{k \neq j}^K f_{kh} X_{nk} \right)^2 - \sum_{k \neq j}^K f_{kh}^2 X_{nk}^2 \right] + f_{jh} X_{nj} \sum_{k \neq j}^K f_{kh} X_{nk}. \end{aligned}$$

Denoting the sum of all but the last term by $\widehat{Y}_n(j, h)$ we get

$$\widehat{Y}_n = \widehat{Y}_n(j, h) + f_{jh}Z_n(j, h),$$

with

$$Z_n(j, h) = X_{nj} \sum_{k \neq j}^K f_{kh} X_{nk}. \quad (13)$$

Again $\widehat{Y}_n(j, h)$ is (4) without the term $f_{jh}Z_n(j, h)$. Let $\widetilde{f}_{jh}$ denote the updated version of f_{jh} , then it must minimise

$$WSSE = \sum_{n=1}^N w_n \left[Y_n - \widehat{Y}_n(j, h) - \widetilde{f}_{jh}Z_n(j, h) \right]^2. \quad (14)$$

To minimise this equation with respect to $\widetilde{f}_{jh}$, note that we again have a regression problem in which $Z_n(j, h)$ is regressed on the residuals $Y_n - \widehat{Y}_n(j, h) = Y_n - \widehat{Y}_n + f_{jh}Z_n(j, h)$ and the minimising choice of $\widetilde{f}_{jh}$ is given by

$$\widetilde{f}_{jh} = f_{jh} + \frac{\sum_{n=1}^N w_n [Y_n - \widehat{Y}_n] Z_n(j, h)}{\sum_{n=1}^N w_n Z_n^2(j, h)}. \quad (15)$$

Thus, the coordinate descend algorithm for minimising WSSE proceeds as follows:

- (a) Start with some initial choice of the estimates (more below).
- (b) Keeping the b_k and the f_{kg} at their values, compute a new value for b_0 from (9).
- (c) Then, keeping all else fixed cycle through $j = 1, \dots, K$, computing new values for b_j from (12).
- (d) Again, keeping all else fixed, cycle through $j = 1, \dots, K$, $h = 1, \dots, G$, computing new values for f_{jh} from (15).
- (e) Repeat steps (b)–(d) until convergence, giving the final CD estimates.

Regarding the choice of initial estimates, we can take $b_k = 0$ for $k = 1, \dots, K$, or as the regression coefficient estimates resulting from a multiple regression fit to the data. We cannot set the f_{jh} equal to 0 since we then have $Z_n(j, h) = 0$ by (13), and (15) fails to produce an update. Hence, something else is needed and in our experiments we followed Rendle (2012) by choosing the initial f_{jh} randomly from a standard uniform distribution denoted by $U(0,1)$ or a standard normal distribution denoted by $N(0,1)$.

2.2 CD algorithm for minimising WSAD

In this subsection, the algorithm for WSAD is described. It follows the same steps as for WSSE except that the squared deviations in (8), (11) and (14) are replaced by absolute deviations.

Taking $\widehat{Y}_n(0)$, $\widehat{Y}_n(j)$, $\widehat{Y}_n(j, h)$ and $Z_n(j, h)$ as defined before, we have current estimates b_0 , $b_1, \dots, b_K$ and f_{kg} , $k = 1, \dots, K$, $g = 1, \dots, G$ which we wish to update via the CD algorithm.

If $\tilde{b}_0$ denotes the updated version of the intercept, then

$$WSAD = \sum_{n=1}^N w_n \left| Y_n - \hat{Y}_n(0) - \tilde{b}_0 \right|$$

must be minimised w.r.t. $\tilde{b}_0$. The choice of $\tilde{b}_0$ that minimises this expression is difficult to obtain, and we will use linear programming software to solve for $\tilde{b}_0$.

If $\tilde{b}_j$ denotes the updated version of the j th regression coefficient, then $\tilde{b}_j$ must minimise

$$WSAD = \sum_{n=1}^N w_n \left| Y_n - \hat{Y}_n(j) - \tilde{b}_j X_{nj} \right|.$$

To minimise this with respect to $\tilde{b}_j$, note that we just have a regression problem in which the X_{nj} are regressed on the residuals $Y_n - \hat{Y}_n(j) = Y_n - \hat{Y}_n + b_j X_{nj}$ and the minimising choice of $\tilde{b}_j$ is again not straightforward and will be solved by using linear programming.

If $\tilde{f}_{jh}$ denotes the updated version of f_{jh} , it must be chosen to minimise

$$WSAD = \sum_{n=1}^N w_n \left| Y_n - \hat{Y}_n(j, h) - \tilde{f}_{jh} Z_n(j, h) \right|.$$

To minimise this with respect to $\tilde{f}_{jh}$, note that we again have a regression problem in which $Z_n(j, h)$ is regressed on the residuals $Y_n - \hat{Y}_n(j, h) = Y_n - \hat{Y}_n + f_{jh} Z_n(j, h)$ and the minimising choice of $\tilde{f}_{jh}$ will again be solved by linear programming.

The coordinate descend algorithm for minimising WSAD follows the same steps as the WSSE routine, but by substituting the WSSE parameter updating equations (9), (12) and (15) with the linear programming equivalent. The initial estimates may be obtained as before, however one might consider substituting the multiple regression coefficient estimates with the estimates obtained from a LAD regression.

2.3 Implementation of the algorithms

Both the above-mentioned algorithms were programmed in SAS using PROC IML and we used PROC LPSOLVE for solving the linear programming problems. See SAS Institute Inc. (2017) for more information on PROC IML and PROC LPSOLVE. As convergence criterion we used the mean squared differences of the change in adjusted coefficients at each subsequent iteration and as a measure of the prediction performance we used the mean squared error. Although CD does not guarantee a global minimum, the loss functions considered here are both convex having a unique minimum, and therefore concern regarding the convergence of the algorithms is alleviated. We will refer to the WSSE CD implementation as the WSSE CD routine and to the WSAD CD implementation as the WSAD CD routine.

Both loss functions were also implemented using the general nonlinear programming (NLP) solver implemented in SAS PROC OPTMODEL. Implementation is straight forward since it only requires the specification of the FM model and the loss function to be minimised. PROC OPTMODEL is a versatile solution giving the programmer the benefit of powerful tried and tested built-in optimisation routines as well as intelligent selection of the starting values for the parameters (see SAS Institute

Inc., 2014). We will refer to these OPTMODEL implementations as the WSSE NLP and WSAD NLP routines. Note that a drawback of these routines is that it requires the SAS/OR license at a substantial cost. Because of this high price tag, the CD algorithms offer a useful alternative since it can be implemented in any programming language. Using both the CD and NLP implementations allow us to validate the numerical accuracy of the two optimisation implementations.

Note that all analyses were performed on a laptop computer (i7 processor, CPU = 1.8 GHz).

3. Examples

In this section, we firstly test the accuracy of our algorithms by means of a simulation study, and secondly fit FMs to an artificially constructed ratings data set and lastly to the well-known Boston house price data set. Note that we used equally weighted loss functions in all calculations performed in this section.

3.1 Testing the accuracy of the fitting routines

Consider the FM model given in (1). We will first fit a 1-factor and then a 2-factor FM on two artificially generated data sets using the WSSE CD routine and secondly, test the accuracy of all the above-mentioned routines by means of a simulation study.

Data set 1

For the first dataset we took $K = 5$ predictors and $G = 1$ factor. The true values chosen for the parameters are shown in Table 1. $N = 1000$ observations were generated with all 5 predictors taken to be independent and identically distributed according to the standard normal distribution $N(0,1)$ and the error term was taken to be $N(0,0.25)$ distributed. Using equally weighted WSSE, the resulting estimates are shown in Table 1.

Except for the intercept, the estimates of the regression coefficients are quite close to their true values. This is also true for the estimates of the factor loadings provided that we change their signs. From (1) we note that replacing all the φ_k by $-\varphi_k$ yields the exact same model so that changing the signs of the estimates is valid. The MSE of the fitted model is 0.3225, which is nearly 30% larger than the true error variance of 0.25. The discrepancies noted in the estimates of the intercept and MSE could be due to sampling error, and this will be investigated in more detail in the simulation study.

Table 1. WSSE CD fit on data set 1.

Parm	True		Parm	True	
β	value	Estimate	φ	value	Estimate
β_0	0.5	0.29			
β_1	0.8	0.79	φ_{11}	0.5	-0.52
β_2	0.3	0.27	φ_{21}	-0.6	0.54
β_3	0.2	0.17	φ_{31}	0.2	-0.18
β_4	0.0	0.00	φ_{41}	0.0	0.00
β_5	0.0	0.03	φ_{51}	0.0	-0.03

Data set 2

For the second data set we took $K = 8$ predictors and $G = 2$ factors. The true values chosen for the parameters are shown in Table 2 below. $N = 2500$ observations were generated with all 8 predictors taken to be standard normal variates and the error term as before. The estimates are also shown in the table. Again, except for the intercept, the estimates of the regression coefficients are very close to their true values. The MSE of the fitted model is 0.3072 and about 22% larger than the true error variance of 0.25. Again these discrepancies will be studied in the simulation study below.

Note that the estimates of the factor loadings in this example may give the impression that the estimated factor loadings are always close to the true values up to a sign change. When fitting FMs to other data sets we found that the estimated factor loadings are not always close to the true values, but that the inner product of the estimated factor loadings is always close to the inner product of the true factor loadings.

Simulation study

The objective of this study is to investigate the accuracy and efficiency of the routines implemented. Assuming the model parameters as given in Table 2, we simulate 100 data sets of 2500 observations each, by drawing the errors in model (1) from a contaminated normal distribution. Here the errors were drawn with probability 0.9 from an $N(0, 0.25)$ distribution and, in order to simulate the presence of outliers, with probability 0.1 from an $N(0, 25)$ distribution. This was done to check whether the WSAD routines outperform the WSSE routines when outliers are present. Since the estimates of the factor loadings sometimes vary in terms of sign and closeness to the true factor loadings, we decided to report the accuracy of the factor estimates in terms of the deviance of the true inner product of the factor loadings with that of the associated estimated inner product. For each combination of predicted variables, the true inner product is given in Table 3. Note from (3) that these values can also be viewed as the true implied interaction coefficients.

Starting values for the parameters of the NLP routines were not specified and we left that for PROC OPTMODEL to decide. For the CD routines, the starting values for the coefficients of the predictor variables and intercept, were obtained by fitting a least squares (LS) regression for WSSE and a LAD

Table 2. WSSE CD fit on data set 2.

Parm	True		Parm	True		Parm	True	
β	value	Estimate	φ	value	Estimate	φ	value	Estimate
β_0	0.5	0.26						
β_1	0.8	0.80	φ_{11}	0.5	0.45	φ_{12}	1.0	1.03
β_2	0.3	0.30	φ_{21}	-0.6	-0.62	φ_{22}	0.0	-0.01
β_3	0.2	0.20	φ_{31}	0.2	0.18	φ_{32}	0.0	0.02
β_4	0.0	0.00	φ_{41}	0.0	-0.03	φ_{42}	0.0	-0.01
β_5	0.0	0.01	φ_{51}	0.0	0.00	φ_{52}	0.5	0.49
β_6	0.1	0.10	φ_{61}	0.0	-0.01	φ_{62}	0.5	0.50
β_7	0.2	0.20	φ_{71}	0.0	0.03	φ_{72}	-0.5	-0.50
β_8	0.9	0.90	φ_{81}	1.0	0.95	φ_{82}	0.5	0.53

Table 3. True inner product of factor loadings.

	X_2	X_3	X_4	X_5	X_6	X_7	X_8
X_1	-0.30	0.10	0.00	0.50	0.50	-0.50	1.00
X_2		-0.12	0.00	0.00	0.00	0.00	-0.60
X_3			0.00	0.00	0.00	0.00	0.20
X_4				0.00	0.00	0.00	0.00
X_5					0.25	-0.25	0.25
X_6						-0.25	0.25
X_7							-0.25

regression for WSAD. In both cases the starting values for the factor loadings were generated from a standard uniform distribution.

The results of the simulation study are given in Tables 4, 5, 6 and 7. The performance of the routines in terms of the number of iterations until convergence and time taken to converge is given in Table 4. Note that the convergence criteria were calculated for each data set and the table contains the summary statistics (average, standard deviation, minimum and maximum) obtained over 100 repetitions.

As expected, when compared to the more intricate WSAD routines, the number of iterations needed until convergence are less for the WSSE routines. Also, the time taken to converge, measured in hours (h), minutes(m) and seconds(s), clearly shows the superiority of the NLP implementations in terms of computational efficiency. Interestingly, compared with the WSAD NLP routine, the WSAD CD routine needs fewer iterations to converge, but takes much longer.

We calculated the MSE for each generated data set and then averaged the MSEs over simulation runs. As seen in Table 5, the average MSEs obtained by the four routines are within 2% of the theoretical variance of 2.725. This alleviates the concern we had in the previous section about the MSE estimate not being close to the true value.

Note that the two WSSE routines give the same answers, as do the two WSAD routines. Closer

Table 4. Summary statistics of performance criteria.

		Average	Standard deviation	Minimum	Maximum
Number of iterations	WSSE CD	32.03	9.4	14	62
	WSSE NLP	9.32	1.18	7	13
	WSAD CD	98.15	36.54	19	240
	WSAD NLP	191.27	115.89	105	732
Time to converge	WSSE CD	7.95s	2.70s	3.36s	16.92s
	WSSE NLP	0.15s	0.03s	0.11s	0.28s
	WSAD CD	17m53.84s	1h12m16.15s	2m44.07s	12h15m40.64s
	WSAD NLP	6.54s	4.20s	3.22s	31.78s

Table 5. Summary statistics of mean squared prediction error.

	Average	Standard deviation	Minimum	Maximum
WSSE CD	2.69	0.23	2.20	3.28
WSSE NLP	2.69	0.23	2.20	3.28
WSAD CD	2.71	0.24	2.21	3.30
WSAD NLP	2.71	0.24	2.21	3.30

Table 6. Summary statistics of the coefficient estimates of the predictor variables.

	True values	Average of estimates				Standard deviation of estimates			
		WSSE CD	WSSE NLP	WSAD CD	WSAD NLP	WSSE CD	WSSE NLP	WSAD CD	WSAD NLP
β_0	0.5	0.502	0.502	0.501	0.501	0.031	0.031	0.015	0.016
β_1	0.8	0.803	0.803	0.800	0.801	0.032	0.032	0.013	0.013
β_2	0.3	0.300	0.300	0.300	0.301	0.034	0.034	0.013	0.014
β_3	0.2	0.195	0.195	0.203	0.203	0.031	0.031	0.012	0.013
β_4	0.0	0.002	0.002	-0.001	0.000	0.029	0.029	0.014	0.014
β_5	0.0	0.000	0.000	0.001	0.001	0.030	0.030	0.014	0.015
β_6	0.1	0.098	0.098	0.101	0.100	0.033	0.033	0.014	0.014
β_7	0.2	0.205	0.205	0.201	0.202	0.032	0.032	0.014	0.014
β_8	0.9	0.899	0.899	0.898	0.898	0.030	0.030	0.013	0.013

Table 7. MSE of true and estimated inner products of factor loadings (WSSE NLP fit).

	X_2	X_3	X_4	X_5	X_6	X_7	X_8
X_1	0.0011	0.0005	0.0005	0.0007	0.0009	0.0010	0.0011
X_2		0.0005	0.0004	0.0007	0.0006	0.0005	0.0011
X_3			0.0000	0.0002	0.0003	0.0003	0.0009
X_4				0.0002	0.0002	0.0002	0.0007
X_5					0.0006	0.0007	0.0007
X_6						0.0007	0.0007
X_7							0.0012

inspection revealed that when the MSE values for each generated data set were compared, the two WSSE routines gave the same answers up to seven decimals and the WSAD routines the same answers up to two decimals. Of course, this should be the case as the loss functions are convex and therefore both CD and NLP converge to the same solution. Any difference observed relates to the stopping criteria, which can be adjusted to improve the similarity of the results obtained.

For each of the four routines, Table 6 contains the average and standard deviation of the coefficient estimates of the predictor variables. Again, the results obtained for both WSSE routines and for both WSAD routines are the same up to two decimal places. Closer inspection revealed that the coefficient estimates obtained for each data set were almost exactly the same, i.e. up to five decimals for the WSSE routines and up to two decimal places for the WSAD routines. Note that the accuracy with which the intercept term is estimated is close to that of the other regression coefficients, again alleviating our concern expressed when we analysed data sets 1 and 2.

The average of the coefficient estimates over simulation runs are close to the true values and no significant bias is observed. As expected, the standard deviation of the coefficient estimates of the robust WSAD fits are lower than the standard deviations obtained by the WSSE fits, indicating that robust FMs are worthwhile considering when outliers are present in a data set. Note that the MSE of the difference between the true and estimated coefficients can be calculated by adding the squared bias and squared standard deviation. In all cases this number is very small, i.e. about 0.001 or less. Table 7 contains the MSE of the difference between the estimated and true inner products of the factor loadings. We only provide the results for the WSSE NLP routine since the MSEs for the WSSE CD routine was identical and that of the WSAD routines even smaller. The above-mentioned results confirm that the four routines are numerically accurate, and that the NLP routines converge much faster than the CD routines.

Remarks

- We investigated the sensitivity of the results of the CD routines to different starting values by setting the starting values for the coefficients of the predictor variables to zero, and by generating the starting values for the factor loadings from a standard normal distribution. The results were virtually identical and led us to believe that the routines are not affected by different starting values, at least not in this study.
- We conducted a similar simulation study by generating the errors from a $N(0, 0.25)$ distribution. As expected, the WSSE routines outperformed the WSAD routines in terms of having smaller standard deviations.
- Assuming model (2), we conducted a similar simulation study, but in this case specified the values of the interaction coefficients and not that of the factor loadings. Note that now 37 ($1 + 8 + 28$) parameters have to be estimated whereas a 2-factor FM ($G = 2$) only requires the estimation of 25 ($1 + 8 + 16$) parameters. Again, the results obtained were very similar to what we presented here. Therefore, the FM fitting routines did a remarkable job of estimating the coefficients of the predictor variables and factor loadings using the approximation in (3).
- Compared to the CD routines, the NLP routines are computationally much faster and give very similar results. Given space considerations, we decided to present only the results of the NLP routines for the two examples below.

3.2 Recommender system example

To study the behaviour of FMs in a recommender system context, we decided to construct a ratings data set exhibiting a structured pattern, and then distort the pattern to see whether FMs can pick up the original structure. The ratings, on a 5-point scale, are assumed to be provided by 20 users (denoted by U_1 to U_{20}) on 20 items (denoted by I_1 to I_{20}). For example, the items could be movies that were rated by viewers (users), where a low rating by a viewer indicates a poor movie and high rating an excellent one. The ratings are given in Table 8, and it should be clear that identical ratings by users have been organised in five blocks containing four users each. The ratings in this table will be referred to as the original data set since it contains the original ratings.

We then randomly removed 20% of the ratings, indicated by the grey shaded cells, and introduced 10 errors (or outliers) by changing the ratings in the black shaded cells. In each of these cells a rating of 1 was replaced with a 5 and a rating of 5 with a 1.

The data set that remains after the removal of the 20% ratings will be referred to as the incomplete data set (without outliers), and the incomplete data set containing the outliers as the incomplete data set (with outliers). Note that both the incomplete data sets do not contain identical users and, should the labels of users and items be randomised, the block structure will not be clearly visible. Our objective is now to determine whether FMs can correctly predict the missing ratings with and without the presence of outliers. Since we want to predict an ordinal rating using two nominal predictors, the relevance of model (1) may be questioned. However, nothing prevents us from applying the WSSE or WSAD FM fitting routines to this data set; in fact, in his first paper Rendle (2010) used MSE loss to fit the MovieLens data set which is a ratings data set like the one studied here. See also the last remark at the end of this section.

Before we can fit an FM to the incomplete data sets, we have to input it into an appropriate format understood by the algorithms. The data set was encoded using one hot or dummy coding (see Slabber et al., 2021 or Rendle, 2012). Using this method, the two predictor variables (users and items) escalate to 40 nominal predictors (the 20 levels of users plus the 20 levels of items). Suppose we want to fit an FM with 10 factors (FM10), then this will require the estimation of 441 ($1 + K + KG = 1 + 40 + 40(10) = 441$) parameters.

Note that the number of parameters are more than the number of observations (320, i.e. 400 minus the 20% missing values), which is typical when fitting recommender systems. However, compared to fitting a full two-way interaction regression model (821 parameters), the FM10 model results in a significant reduction in the number of parameters that need to be estimated. When G is equal to 4, the reduction in the number of parameters is a further 240, since instead of 441, only 201 parameters need to be estimated.

Once the FM model has been fitted, we obtain 320 estimated ratings $\hat{y}_1, \dots, \hat{y}_{320}$ and 80 predicted ratings $\hat{y}_{321}, \dots, \hat{y}_{400}$. These are all numeric values and do not correspond exactly to a particular rating category. In order to achieve this, we transform the numeric estimates and predictions to $\hat{y}_1^*, \dots, \hat{y}_{400}^*$, where for $i = 1, \dots, 400$,

$$\hat{y}_i^* = \begin{cases} \lfloor 0.5 + \hat{y}_i \rfloor & \text{for } 0.5 \leq \hat{y}_i < 5.5, \\ \hat{y}_i & \text{otherwise,} \end{cases}$$

with $\lfloor x \rfloor$ the largest integer contained in x . For example, if $\hat{y}_1 = 2.4$ then $\hat{y}_1^* = 2$ or if $\hat{y}_2 = 2.6$ then

Table 8. Original ratings data set.

	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	I_{16}	I_{17}	I_{18}	I_{19}	I_{20}
U_1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5
U_2	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5
U_3	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5
U_4	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5
U_5	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4
U_6	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4
U_7	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4
U_8	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4
U_9	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3
U_{10}	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3
U_{11}	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3
U_{12}	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2	3	3	3	3
U_{13}	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2
U_{14}	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2
U_{15}	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2
U_{16}	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1	2	2	2	2
U_{17}	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1
U_{18}	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1
U_{19}	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1
U_{20}	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	1	1	1	1

$\hat{y}_2^* = 3$. Let $y_i, i = 1, \dots, 400$ denote the original (unchanged) ratings, then the fitted MSE is defined as $\frac{1}{320} \sum_{i=1}^{320} (\hat{y}_i^* - y_i)^2$ and the predicted MSE as $\frac{1}{80} \sum_{i=321}^{400} (\hat{y}_i^* - y_i)^2$.

Firstly, we fit the algorithms to the encoded incomplete data set (without outliers) for various choices of G . The results of the NLP WSSE and WSAD FM fits are given in Table 9 for different choices of factors $G = 2, 3, 4, 6, 8$ and 10 . In addition to the fitted MSE and predicted MSE, we also counted the number of estimated and predicted transformed ratings that were unequal to the value of the original ratings and present the results under the headings ‘Incorrectly fitted’ and ‘Incorrectly predicted’ respectively. This, together with the run time of the algorithms are given in Table 9 for the NLP implementation.

The results show that, when using less than 4 factors, both the WSSE and WSAD algorithms struggle to fit the data set. Also, the run time of the WSAD routines is long, especially for smaller choices of G . As far as model selection is concerned, one would always prefer a model that generalises well and does not over-fit the data set. An indication of overfitting, and therefore poor generalisation, would be a very small fitted MSE (overfitting) and a large predicted MSE (poor generalisation). For an in-depth discussion of these topics the interested reader is referred to the excellent texts by Hastie et al. (2009), Hastie et al. (2015) and Efron and Hastie (2016). When 4 or more factors are specified, both the WSSE and WSAD algorithms obtain perfect fits, with the 4-factor models being the parsimonious choice. It is interesting to note that the untransformed estimated and predicted ratings, for the perfect fitting models, were all within one hundredth of the corresponding original ratings.

An obvious question would be to determine how many missing values will be needed before the model fails to pick up the structure in the data. Although this can be researched further, it remains remarkable how well the model is able to predict. Part of the success may be attributed to the fact that the interactions appear as inner products in (1), hence allowing the estimation of each component to “borrow strength” from other components (see Rendle, 2010, 2012; Slabber et al., 2021).

Table 9. WSSE and WSAD FM fits on the incomplete data set (without outliers).

	G	Fitted MSE	Predicted MSE	Incorrectly fitted	Incorrectly predicted	Run time
WSSE	2	0.672	1.013	206	60	0.58s
	3	0.328	0.675	105	45	0.01s
	4	0.000	0.000	0	0	1.36s
	6	0.000	0.000	0	0	0.01s
	8	0.000	0.000	0	0	7.17s
	10	0.000	0.000	0	0	11.43s
WSAD	2	Did not converge within 10 hours				
	3	0.700	0.942	18	6	1h18m17.82s
	4	0.000	0.000	0	0	1h25m1.43s
	6	0.000	0.000	0	0	2m15.30s
	8	0.000	0.000	0	0	1m50.22s
	10	0.000	0.000	0	0	14m3.06s

Secondly, for $G = 4$ and $G = 8$, we fit the algorithms to the encoded incomplete data set with outliers. In Table 10 we present the fitted and predicted MSE of the WSSE and WSAD fits. Again, we present the number of incorrectly fitted and predicted ratings and the run time of the routines. The last column contains the number of outliers that are corrected, meaning that the estimated transformed rating is equal to the original rating and not the value of the outlier.

The results in Table 10 show that the best model is WSAD FM4, since it corrected 9 outliers and gave only one incorrect prediction and 3 incorrect estimates. When observing the estimated and predicted ratings of the WSAD FM4 fit (see the first remark at the end of this section), it turns out that for user U_9 the estimated transformed ratings for I_5 , I_6 , and I_7 as well as the predicted rating for I_8 are all equal to 2 (untransformed rating estimate 1.92), instead of the original ratings of 5 (refer to Table 8). So, WSAD FM4 only failed in the above-mentioned 4 cases, therefore estimating and predicting the original ratings in all other cells correctly. When we examined the coefficient estimates of U_9 (see the first remark at the end of this section), we noticed that the vector of coefficient estimates of U_9 , which should ideally be the same as that of U_{10} , U_{11} and U_{12} , does not fit the general block structure as if the algorithm had difficulty deciding whether U_9 is closer to U_8 or U_{10} . Interestingly, the coefficient estimates for items I_5 , I_6 , I_7 , and I_8 are almost exactly the same. Consider Table 8 and observe the location of the outlier (U_9 's rating on I_7 , which is now 1 instead of 5) and the missing value (U_9 's rating on I_8). The two cells find themselves on the edge of a block originally rated as 5s and adjacent to three other blocks originally rated as 1s (above), 2s (above and to the right), and 1s (to the right). The FM clearly has difficulty deciding to which of the four blocks the four circled cells belong, and therefore the transformed estimated ratings of 2 is understandable.

When comparing the 8-factor WSSE FM fit with that of the 4-factor WSSE FM fit, the much higher predicted MSE of the former might seem strange given the much lower number of incorrect predictions. Closer inspection revealed that a few very large prediction errors caused the high predicted MSE. The 8-factor fits of both WSSE and WSAD seem to overfit since they predict poorly, but fit the data well. This was confirmed when we inspected the estimated ratings for the non-corrected outliers. In all cases the transformed estimated ratings were equal to the value of the outliers.

Remarks

- Due to space considerations, we do not provide the results of the various FM fits. These results include the coefficient and factor loading estimates as well as the estimated and predicted ratings. These results are available from the first author upon request.

Table 10. WSSE and WSAD FM fits on the incomplete data set (with outliers).

	G	Fitted MSE	Predicted MSE	Incorrectly fitted	Incorrectly predicted	Run time	Outliers corrected
WSSE	4	1492.867	18 588.896	269	68	1.21s	0
	8	0.500	224 181.994	10	13	1m59.64s	0
WSAD	4	0.084	0.113	3	1	40m21.64s	9
	8	0.325	1.720	7	7	3h36m31.67s	4

- It should be clear from the discussion here that one should consider various choices of G , before selecting the final model.
- When we applied PROC FACTMAC to the ratings data set, the results were similar to the WSSE routines, which is expected because PROC FACTMAC also employs the MSE loss function.
- As stated previously, the applicability of the FM model in (1) to this prediction problem might be questioned. Because the target variable is ordinal, it might be more appropriate to consider an FM model formulated and fitted analogous to an ordinal logistic regression. However, it should be clear that the FM models fitted in this section perform remarkably well.

3.3 Boston housing data set

To test FMs on a real-world data set, the well-known publicly available Boston housing data set is used. This data set was first used by Harrison Jr and Rubinfeld (1978) in a study regarding the impact of air pollution (as measured by the square of nitrogen oxide concentration) and 12 other explanatory variables on the price of owner-occupied homes. The data set is a sample of 506 observations on census tracts in the Boston Standard Metropolitan Statistical Area (SMSA) in 1970. For a detailed discussion see Belsley et al. (1980) and the more recent paper by Zhang (2008).

The set, which was recently updated and corrected, is freely available on various websites (see e.g. Perera, 2018). The data set consists of the price variable relevant to the analysis (lmv the logarithm of the median value of owner-occupied homes) and 13 explanatory variables ($crim$, zn , $indus$, $chas$, nox , rm , age , dis , rad , tax , $ptratio$, $black$, $lstat$). Here $crim$ is the per capita crime rate by town, zn the proportion of a town's residential land zoned for lots greater than 25000 square feet, $indus$ the proportion of non-retail business acres per town, $chas$ Charles River dummy variable with value 1 if tract bounds on the Charles River, nox the nitric oxide concentration (parts per hundred million) squared, rm the average number of rooms per dwelling, age the proportion of owner-occupied units built prior to 1940, dis the logarithm of the weighted distances to five employment centres in the Boston region, rad the logarithm of the index of accessibility to radial highways, tax the full-value property tax rate per \$10 000, $ptratio$ the pupil-teacher ratio by town, $black$ is $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks in the population and $lstat$ the logarithm of the proportion of the population that is of lower status.

Because of the scale differences among the predictor variables, and because we are interested in identifying interaction effects amongst the predictor variables, we standardised all the predictor variables. See for example Frost (2019) on the importance of standardisation in regression analysis when the model contains polynomial and/or interaction terms. The importance of standardisation was not mentioned previously, because the predictors in the ratings data set are binary (dummy variables) and the predictors in the simulation study are standard normal variates, therefore not requiring standardisation.

To test the performance of FMs on this data set, we split the set in a training and test data set. As is done frequently in data competitions (see e.g. Mandav, 2021), we selected an 80% random sample from the full data set as the training set and the remaining observations as the test set. We then fitted several models on the training data set and tested the prediction performance on the test set. The models considered were a multiple regression, a multiple regression including all two-way

Table 11. Boston data: Model comparison on training and test data set.

Model	Parameters	Fitted RMSE	Predicted RMSE
Multiple regression	13	4.794	4.324
Regression (incl. two-way interactions)	92	2.467	4.008
WSSE FM2	40	3.102	3.445
WSAD FM2	40	3.624	3.381
WSSE FM4	66	2.742	5.024
WSAD FM4	66	3.133	4.927

interactions, and 2- and 4-factor (WSSE and WSAD) FMs. SAS PROC GLM was used to fit the regressions and we present the results of the NLP FM fits in Table 11. Again, the results of the CD routines were similar to that of the NLP routines, but again it took a much longer time to converge. Since the data competitions usually measure performance using root mean square error (RMSE) we report the fitted and predicted RMSE and not the MSE as was done previously.

Note that the models in Table 11 are nested within each other, therefore the smaller model will always have the larger fitted RMSE. For example, as seen in Table 11, the fitted RMSE obtained by the two-way interaction regression (the most complex model having 92 parameters) is smallest; while the fitted RMSE obtained by the multiple regression (the simplest model having 13 parameters), is the largest. As far as prediction is concerned, WSAD FM2 has the lowest predicted RMSE, closely followed by WSSE FM2. The good performance of the 2-factor FMs suggests the presence of interaction effects between the predictor variables, and the superior performance of WSAD FM2 suggests the presence of outliers. The latter has been confirmed by several studies (see e.g. Belsley et al., 1980).

To investigate the presence of interaction effects, we studied the significant interaction effects as obtained from the two-way interaction regression model. The model identified significant interaction effects between variables *crim* and *chas*, *crim* and *rm*, *rm* and *lstat*. The fact that significant interaction effects are present explains the better performance of the FM2 model. The predicted RMSE of 3.381 obtained by WSAD FM2 and 3.445 obtained by WSSE FM2, compare well with the best performers on the Kaggle leader boards; see Kaggle (2021). The leader boards list two scores based on RMSE, one based on 70% of the test data (top scores 3.72, 3.93 and 3.94) and the other one on 30% of the test data (top scores 3.69, 3.89 and 3.94). Although our scores are not directly comparable it has lowest predicted RMSE.

4. Conclusion

Assuming a factorisation machine model formulation, we minimise two well-known loss functions by using coordinate descent and non-linear programming. The coordinate descent routines were implemented in SAS PROC IML and the nonlinear programming routines in SAS PROC OPTMODEL. We demonstrated that these routines are adequate for fitting factorisation machine models and that these routines may provide a useful addition to the statistician's model building toolkit. Furthermore, the implementation in SAS might stimulate the application of FMs to many prediction problems in business. We are currently working on extending our routines to include regularisation and are testing

the performance of the routines on large data sets. Also, we are doing research on the applicability of FMs in a credit scoring context, using an FM formulation analogous to logistic regression.

Acknowledgments. The authors acknowledge that this research benefited from valuable contributions by Prof. Hennie Venter and Prof. Tertius de Wet. A detailed and extremely valuable report by a referee assisted us to substantially improve the presentation of the paper. This work is based on research supported in part by the Department of Science and Innovation (DSI) of South Africa. The grant holder acknowledges that opinions, findings and conclusions or recommendations expressed in any publication generated by DSI-supported research are those of the authors and that the DSI accepts no liability whatsoever in this regard.

References

- BAYER, I. (2016). fastfm: A library for factorization machines. *Journal of Machine Learning Research*, **17**, 6393–6397.
- BELSLEY, D. A., KUH, E., AND WELSCH, R. E. (1980). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. Wiley, New York, NY.
- DE JONGH, P. J., DE WET, T., AND WELSH, A. H. (1988). Mallows-type bounded-influence-regression trimmed means. *Journal of the American Statistical Association*, **83**, 805–810.
- EFRON, B. AND HASTIE, T. (2016). *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*. Cambridge University Press, Cambridge.
- FROST, J. (2019). *Introduction to Statistics: An Intuitive Guide for Analyzing Data and Unlocking Discoveries*. Statistics by Jim Publishing, State College, PA.
- HARRISON JR, D. AND RUBINFELD, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, **5**, 81–102.
- HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. H. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, NY.
- HASTIE, T., TIBSHIRANI, R., AND WAINWRIGHT, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, Boca Raton, FL.
- HONG, L., DOUMITH, A. S., AND DAVISON, B. D. (2013). Co-factorization machines: Modeling user interests and predicting individual decisions in Twitter. *In Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*. 557–566.
- JUAN, Y., LEFORTIER, D., AND CHAPELLE, O. (2017). Field-aware factorization machines in a real-world online advertising system. *In Proceedings of the 26th International Conference on World Wide Web Companion*. 680–688.
- JUAN, Y., ZHUANG, Y., CHIN, W.-S., AND LIN, C.-J. (2016). Field-aware factorization machines for CTR prediction. *In Proceedings of the 10th ACM Conference on Recommender Systems*. 43–50.
- KAGGLE (2021). House price prediction with Boston housing dataset.
URL: <https://www.kaggle.com/c/house-price-prediction-with-boston-housing-dataset>
- KUTNER, M. H., NACHTSHEIM, C. J., NETER, J., AND LI, W. (2005). *Applied Linear Statistical Models*. McGraw-Hill Irwin, New York, NY.
- LIU, C., ZHANG, T., LI, J., YIN, J., ZHAO, P., SUN, J., AND HOI, S. C. H. (2019). Robust factorization

- machine: A doubly capped norms minimization. *In Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 738–746.
- MANDAV, H. (2021). Train and test data split for ML models.
URL: <https://medium.com/mllearning-ai/train-and-test-data-split-152bad0afbb2>
- MARONNA, R. A., MARTIN, R. D., YOHAI, V. J., AND SALIBIÁN-BARRERA, M. (2019). *Robust Statistics: Theory and Methods (with R)*. Second edition. Wiley, Hoboken, NJ.
- NORTON, E. C., WANG, H., AND AI, C. (2004). Computing interaction effects and standard errors in logit and probit models. *The Stata Journal*, **4**, 154–167.
- PARSONS, N. (2017). Factorization machines for recommendation systems.
URL: <https://getstream.io/blog/factorization-recommendation-systems>
- PERERA, P. (2018). The Boston housing dataset.
URL: <https://www.kaggle.com/prasadperera/the-boston-housing-dataset/data>
- PIJNENBURG, M. AND KOWALCZYK, W. (2017). Extending logistic regression models with factorization machines. *In International Symposium on Methodologies for Intelligent Systems*. Springer, 323–332.
- PUNJABI, S. AND BHATT, P. (2018). Robust factorization machines for user response prediction. *In Proceedings of the 2018 World Wide Web Conference*. 669–678.
- RENDLE, S. (2010). Factorization machines. *In 2010 IEEE International Conference on Data Mining*. 995–1000.
- RENDLE, S. (2012). Factorization machines with LibFM. *ACM Transactions on Intelligent Systems and Technology*, **3**, 1–22.
- ROUSSEEUW, P. J. AND LEROY, A. M. (2005). *Robust Regression and Outlier Detection*. Wiley, New York, NY.
- SAS INSTITUTE INC. (2014). *SAS/OR® 13.2 User's Guide: Mathematical Programming*. SAS Institute Inc., Cary, NC.
URL: <https://support.sas.com/documentation/onlinedoc/or/132/optmodel.pdf>
- SAS INSTITUTE INC. (2017). *SAS® Visual Data Mining and Machine Learning 8.2: Procedures*. SAS Institute Inc., Cary, NC.
URL: <https://documentation.sas.com/api/docsets/casml/8.2/content/casml.pdf>
- SAS INSTITUTE INC. (2019). *SAS® Visual Data Mining and Machine Learning 8.5: Procedures*. SAS Institute Inc., Cary, NC.
URL: <https://documentation.sas.com/api/docsets/casml/8.5/content/casml.pdf>
- SELIVANOV, D. (2021). *rsparse: Statistical Learning on Sparse Matrices*. R package version 0.5.0.
URL: <https://CRAN.R-project.org/package=rsparse>
- SLABBER, E., VERSTER, T., AND DE JONGH, R. (2021). Advantages of using factorisation machines as a statistical modelling technique. *South African Statistical Journal*, **55**, 125–144.
- YUROCHKIN, M., NGUYEN, X., AND VASILOGLOU, N. (2017). Multi-way interacting regression via factorization machines. *In Proceedings of the 31st Conference on Neural Information Processing Systems*. 2595–2603.
- ZHANG, T. (2008). Adaptive forward-backward greedy algorithm for sparse learning with linear

models. *Advances in Neural Information Processing Systems*, **21**, 1921–1928.

Chapter 4

**Article title: Some insights about the applicability of
logistic factorisation machines in banking**

Authors:

E. Slabber, T.Verster and R. de Jongh

The article was submitted to Risks.

Article

Some Insights about the Applicability of Logistic Factorisation Machines in Banking

Erika Slabber, Tanja Verster and Riaan de Jongh *

Centre for BMI, North-West University, Potchefstroom, 2520, South Africa; erika.slabber@outlook.com (E. S.); tanja.verster@nwu.ac.za (T.V.)

* Correspondence: erika.slabber@outlook.com; Tel.: +27-826651127

Abstract: Logistic regression is a very popular binary classification technique in many industries, particularly in the financial service industry. It has been used to build credit scorecards, estimate the probability of default or churn, identify the next best product in marketing, and many more applications. The machine learning literature has recently introduced several alternative techniques, such as deep learning neural networks, random forests, and factorisation machines. While neural networks and random forests form part of the practitioner's model-building toolkit, factorisation machines are seldom used. In this paper, we investigate the applicability of factorisation machines to some binary classification problems in banking. To stimulate the practical application of factorisation machines, we implement the fitting routines, based on logit loss and maximum likelihood, on commercially available software that is widely used by banks and other large financial services companies. Logit loss is usually used by the machine learning fraternity while maximum likelihood is popular in statistics. Depending on the coding of the target variable, we will show that these methods yield identical parameter estimates. Often, banks are confronted with predicting events that occur with low probability. To deal with this phenomenon, we introduce weights in the above-mentioned loss functions. The accuracy of our fitting algorithms is then studied by means of a simulation study and compared with logistic regression. The separation and prediction performance of factorisation machines are then compared to logistic regression and random forests by means of three case studies covering a recommender system, credit card fraud, and a credit scoring application. We conclude that logistic factorisation machines are worthy competitors of logistic regression in most applications, but with clear advantages in recommender systems applications where the number of predictors typically outnumbers the number of observations.

Keywords: logistic regression; factorisation machines; random forests; machine learning; recommender system, credit scoring; logit loss, maximum likelihood estimation

Citation: Slabber, Erika, Tanja Verster, and Riaan de Jongh. 2023. Some Insights about the Applicability of Logistic Factorisation Machines in Banking. *Risks* 11: x. <https://doi.org/10.3390/xxxxx>

Academic Editor(s): Mogens Steffensen and Elisa Luciano

Received: 24 November 2022

Revised: 30 January 2023

Accepted: 16 February 2023

Published: date



Copyright: © 2023 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The logistic regression model is one of the most commonly used statistical techniques for solving binary classification problems (Siddiqi 2017; Hand and Henley 1997; Baesens et al. 2016). Banks are heavily reliant on logistic regression when building retail credit scorecards. The latter methodology allows for a straightforward transformation of the estimated default probabilities into scores and facilitates the interpretation of the predictor variables in terms of their importance and effect on default through the odds ratio. Not only are these models applied to assess the creditworthiness of clients but also to other areas, such as debt collection and marketing. In marketing analytics, it has frequently been used, amongst others, to build customer attrition models and campaign response models. There are many other popular classifiers, such as neural networks, XGBoost, and random forests (see, e.g., Lessmann et al. 2015), but these are mostly used for building challenger models and not often implemented as production models. One of the reasons for this is

the lack of interpretability of some of these models, which hampers their implementation and acceptance by regulators. Yet, as discussed by Gilpin et al. (2018), there are methods to overcome this limitation, e.g., Shapley values, Local Interpretable Model-agnostic Explanations (LIME), and Accumulated Local Effects (ALE) plots, but these have not yet been adopted widely. In this paper, our focus is on factorisation machines, a modelling technique that is popular among machine learners, but seldom used by statisticians and practitioners. A reason for the lack of interest among statisticians is that they are probably unaware of the many papers on factorisation machines that are mostly published in machine learning conference proceedings. Practitioners, on the other hand, use commercial software for building production systems and fitting algorithms for factorisation machines are restricted or not readily available. Therefore, the main aim of this paper is to make practitioners and statisticians aware of the power of this modelling tool as a binary classifier and to provide fitting algorithms for use in commercial software (SAS). Using the logistic regression model definition, we adapt the standard definition of factorisation machines to include logistic factorisation machines and develop fitting algorithms based on maximum likelihood and logit loss. Logit loss is usually used by the machine learning fraternity, while maximum likelihood is popular in statistics. Depending on the coding of the target variable, we will show that these methods yield identical parameter estimates. Often, banks are confronted with predicting events that occur with low probability. To deal with this phenomenon, we introduce weights in the above-mentioned loss functions. The accuracy of our fitting algorithms is then studied by means of a simulation study and compared with logistic regression. The separation and prediction performance of the methods are then investigated by means of three case studies covering a recommender system, credit card fraud, and credit scoring applications. Like factorisation machines, tree-based learning methods also address the interactive features between variables, and, therefore, we include random forests (Breiman 2001) when we investigate the predictive ability of factorisation machines. In the end, we will show that factorisation machines are worthy competitors of logistic regression and random forests and should be included in the modelling toolkit of statisticians and practitioners.

The paper is organised as follows. In the next section, we introduce logistic factorisation machines, and then, in Section 3, we discuss the algorithms for fitting logistic factorisation machines. In particular, we show that subject to the binary coding of the target variable, identical parameter estimates are obtained when minimising the sum of logit loss or by maximising the logistic log-likelihood. In Section 4, we provide an overview of metrics that are typically considered for evaluating the performance of binary classifiers. In order to test the accuracy of our routines, we conduct a simulation study of which the results are presented and discussed in Section 5. Then, in Section 6, the performance of the techniques is further evaluated in a number of case studies that involve the prediction of customer preferences (a recommender system application), prediction of credit card fraud (a fraud application), and prediction of the probability of default (a scorecard application). The first data set is an artificially constructed one that clearly demonstrates the ability of factorisation machines in a recommender system setting. The second data set is publicly available from the Kaggle website, and the last data set was sourced from one of our client projects. The latter data set is proprietary to the client and cannot be disclosed. Section 7 concludes the paper.

2. Logistic Factorisation Machines

For binary response models, the response Y of an individual or an experimental unit can take on one of two possible values, denoted for convenience by 0 and 1 (for example, $Y = 1$ if a certain event occurred, otherwise, $Y = 0$). The probability of an event occurring, given a set of predictor variables $\mathbf{X}$, can be written as $p(\mathbf{X}) = P(Y = 1 | \mathbf{X})$. Then, $p(\mathbf{X})$ may be modelled by the logistic function $p(\mathbf{X}) = \frac{1}{1 + \exp(-\boldsymbol{\beta}'\mathbf{X})}$, where $\boldsymbol{\beta}$ is a vector of parameters and p is the linkage function in a generalised linear model. Let $\mathbf{x}$ denote a

particular realisation of $\mathbf{X}$, then the odds of a positive event are $\frac{p(\mathbf{x})}{1-p(\mathbf{x})}$. Notice that since $p(\mathbf{x}) \in [0,1]$, the logarithm of the odds ranges in the set of real numbers, thus constituting an unrestricted continuous quantity to which it is possible to fit a linear regression model as

$$\text{logit}(p(\mathbf{x})) = \ln\left(\frac{p(\mathbf{x})}{1-p(\mathbf{x})}\right) = \boldsymbol{\beta}'\mathbf{x} . \quad (1)$$

In order to simplify our notation somewhat, we will drop the conditional dependence notation $p(\mathbf{X})$ and use p for the probability of an event occurring. Given an observed data set $(Y_n, X_{n1}, \dots, X_{nK}), n = 1, \dots, N$, the binary logistic regression model (LR) can now be written as

$$\text{logit}(p_n^{LR}) = \ln\left(\frac{p_n^{LR}}{1-p_n^{LR}}\right) = Z_n^{LR} = \beta_0 + \sum_{k=1}^K \beta_k X_{nk} \quad \text{for } n = 1, \dots, N . \quad (2)$$

Here, Z_n^{LR} is the linear regression predictor function, X_{nk} is the n -th observation of the k -th predictor variable, β_k is the parameter or coefficient of the k -th variable, β_0 is an intercept (or bias) term, N is the number of observations, and K is the number of predictor variables. Note from (2) that $p_n^{LR} = \exp(Z_n^{LR}) / (1 + \exp(Z_n^{LR}))$ and $1 - p_n^{LR} = 1 / (1 + \exp(Z_n^{LR}))$.

Similarly, the logistic regression model with all two-way interactions (LRI) can be defined as

$$\text{logit}(p_n^{LRI}) = \ln\left(\frac{p_n^{LRI}}{1-p_n^{LRI}}\right) = Z_n^{LRI} = \beta_0 + \sum_{k=1}^K \beta_k X_{nk} + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \beta_{kj} X_{nk} X_{nj} \quad (3)$$

for $n = 1, \dots, N$, where Z_n^{LRI} is the linear regression predictor function that incorporates all two-way interactions between the predictor variables. For a detailed exposition of logistic regression, see Kleinbaum et al. (2005) and Hilbe (2009). In this paper, we will not be interested in the interpretation of the interaction effects between two variables but rather in whether the presence of interaction terms in the model improves prediction. Although the interpretation of the coefficient of the interaction between two variables is straightforward in linear models, Ai and Norton (2003) showed that this is not the case when fitting non-linear models. In the latter case, they showed for logit and probit models that the interaction effect could not be evaluated simply by looking at the sign, magnitude, or statistical significance of the coefficient of the interaction term. Instead, the interaction effect requires the computation of the cross derivative, and like the marginal effect of a single variable, the magnitude of the interaction effect depends on all the covariates or predictors in the model. In addition, it can have different signs for different observations, making simple summary measures of the interaction effect difficult. In particular, they showed that the sign of the interaction coefficient does not necessarily indicate the sign of the interaction effect between the two variables and that the interaction effect could be non-zero even if the coefficient of the interaction term is zero. We will not pursue this further but felt that it is important to note when interpreting interaction effects.

Rendle (2010) defined factorisation machines as a general-purpose supervised learning algorithm that can be used for both classification and regression tasks. It is an extension of a linear model that is defined to capture interactions or associations between predictor variables even within high dimensional sparse data sets. Combining the definition of factorisation machines in Rendle (2012) with the logistic model with interactions as given in (3), we can define second-order logistic factorisation machines (LFM) as follows

$$\begin{aligned} \text{logit}(p_n^{LFM}) &= \ln\left(\frac{p_n^{LFM}}{1-p_n^{LFM}}\right) = Z_n^{LFM} \\ &= \beta_0 + \sum_{k=1}^K \beta_k X_{nk} + \sum_{k=1}^{K-1} \sum_{j=k+1}^K \sum_{g=1}^G \phi_{kg} \phi_{jg} X_{nk} X_{nj} \quad \text{for } n = 1, \dots, N , \end{aligned} \quad (4)$$

where G is the number of factors, and Z_n^{LFM} is the second-order factorisation machine predictor function. Note that as stated by Slabber et al. (2022),

$$\beta_{kj} \approx \sum_{g=1}^G \phi_{kg} \phi_{jg} \quad (5)$$

Again, β_0 is the intercept, $\beta_1, \dots, \beta_K$ are the regression coefficients, and $\phi_k, k = 1, \dots, K$ is a G dimensional vector of factor loadings for each variable. Here, $\langle \phi_k, \phi_j \rangle = \sum_{g=1}^G \phi_{kg} \phi_{jg}$ denotes the inner product of the vectors ϕ_k and ϕ_j . Note that both the β 's ($1+K$ parameters) and ϕ 's (KG factor loadings) have to be estimated and that G is an input parameter for the fitting procedure. Therefore, the LFM model in Equation (4) has $1 + K + KG$ parameters (or coefficients), while the regression model in Equation (3) has $1 + K + K(K-1)/2$ parameters. When $G < (K-1)/2$, LFM requires fewer parameters to be estimated. The advantages and disadvantages of FMs compared to regression models with interaction are discussed in Slabber et al. (2021) and Slabber et al. (2022). As mentioned in the latter papers, when the number of predictors is large, ordinary two-way interaction regression models suffer from a combinatorial explosion of parameters which make them impractical to fit (see, e.g., James et al. (202021)). In such cases, LFM provides an attractive alternative since the number of parameters increases in a linear rather than in a quadratic way.

3. Fitting Logistic Factorization Machines

As stated previously, we have a data set $(Y_n, X_{n1}, \dots, X_{nK}), n = 1, \dots, N$, and we want to estimate the parameters of models (2), (3), and (4). In the machine learning literature, for binary responses $Y_n = +1$ or -1 , logit loss (LL) is often used (see, e.g., Rendle (2012)), while in statistics, for binary responses $Y_n = 1$ or 0 , maximum likelihood (ML) is the method of choice (see, e.g., Kleinbaum et al. (2005)). Let Z_n denote any of the above-mentioned predictor functions, and let $l(Y_n, Z_n)$ denote some loss function, then to estimate the parameters contained in Z_n , we can minimise $\sum_{n=1}^N l(Y_n, Z_n)$. We will now show identical parameter estimates resulting from minimising the sum of logit loss and maximising the logistic log-likelihood.

Denote the binary encoding $+1/-1$ by Y'_n and the binary encoding $1/0$ by Y_n . Then, for binary responses Y'_n , and the omission of a constant factor (see Rendle 2010), the logit loss is defined by

$$\begin{aligned} l_{LL}(Y'_n, Z_n) &= \ln(1 + \exp(-Y'_n Z_n)) \\ &= I(Y'_n = -1) \ln(1 + \exp(Z_n)) + I(Y'_n = 1) \ln(1 + \exp(-Z_n)) \end{aligned}$$

where $I(\cdot)$ is the indicator function.

Then, since $Y'_n = 2Y_n - 1$, the above equation can immediately be written as

$$l_{LL}(Y'_n, Z_n) = I(Y_n = 0) \ln(1 + \exp(Z_n)) + I(Y_n = 1) \ln(1 + \exp(-Z_n)), \quad (6)$$

We have $p_n = \exp(Z_n) / (1 + \exp(Z_n))$ and $1 - p_n = 1 / (1 + \exp(Z_n))$, and, therefore, $\ln(1 - p_n) = -\ln((1 + \exp(Z_n)))$ and $\ln(p_n) = -\ln((1 + \exp(-Z_n)))$. Substituting this into (6) yields

$$\begin{aligned} l_{LL}(Y'_n, Z_n) &= -I(Y_n = 0) \ln(1 - p_n) - I(Y_n = 1) \ln(p_n) \\ &= -\{(1 - Y_n) \ln(1 - p_n) + Y_n \ln(p_n)\} = -l_{ML}(Y_n, p_n) \end{aligned} \quad (7)$$

with l_{ML} is the logistic log-likelihood.

Therefore, except for a constant factor, minimising logit loss is the same as minimising the negative of the logistic log-likelihood. So minimising logit loss should provide the same parameter estimates as maximising the logistic log-likelihood.

In the imbalanced case, where the number of non-events (zeros) far outnumber the number of events (ones), the logistic log-likelihood is usually weighted in the following way

$$\sum_{n=1}^N w_n \{Y_n \ln(p_n) + (1 - Y_n) \ln(1 - p_n)\}, \quad (8)$$

with $w_n = \frac{1}{2N_1}$ if $Y_n = 1$, $w_n = \frac{1}{2N_0}$ if $Y_n = 0$, $N_1 = \sum_{n=1}^N I(Y_n = 1)$, and $N_0 = \sum_{n=1}^N I(Y_n = 0)$ and $N_1 + N_0 = N$ (see, e.g., Venter and de Jongh (2023)). Then, the totals for the events and non-events are $\frac{1}{2}$ each, and the overall total of all the weights is 1. Again, maximising (8) will yield the same parameter estimates as minimising

$$\sum_{n=1}^N w_n \{\ln(1 + \exp(-Y'_n Z_n))\}, \quad (9)$$

with $w_n = \frac{1}{2N_1}$ if $Y'_n = +1$ and $w_n = \frac{1}{2N_0}$ if $Y'_n = -1$ and $N_1 = \sum_{n=1}^N I(Y'_n = +1)$, $N_0 = \sum_{n=1}^N I(Y'_n = -1)$ and $N_1 + N_0 = N$

To fit the logistic regression models given in (2) and (3), we will use SAS PROC GLM for maximising (8) with respect to the parameters contained in Z_n^{LR} and Z_n^{LRI} . The LFM's can then be obtained by using the NLP algorithm in SAS PROC OPTMODEL to minimise (8) or (9) with respect to the parameters contained in Z_n^{FM} . Note that SAS software is widely used in the financial services industry and is often the preferred choice for implementing production systems. Note that SAS Viya incorporates a procedure for fitting factorisation machines (PROC FACTMAC), but this routine has limited application (see Slabber et al. (2022)) and cannot be applied to binary classification problems. It requires a SAS Viya licence that is more expensive than the licence required to run our routines. Note that the NLP algorithm intelligently selects starting values for the algorithm and does not require the specification thereof, although it is provided as an option. To fit the random forests, we used SAS PROC HPFOREST. Unless stated otherwise, we used the default settings when fitting these models and denote them by RF. For the implementation of imbalanced data sets, we used balanced random forests (BFR).

Once a model has been fitted, an estimate $\hat{p}_n$ for p_n becomes available. To predict or classify a binary value, an 'optimal' threshold (t) or cut-off must be determined. This will be discussed in more detail in the next section.

4. Performance Measures

As stated previously, our main objective is to evaluate the performance of LFM's in comparison with LR and LRI using typical data sets found in banking. We will discuss the measures that are mostly referenced in the literature. See, for example, Siddiqi (2017), Baesens et al. (2016), Prorokowski (2019), Engelmann and Rauhmeier (2006), and James et al. (2021). Performance measures are commonly broken into discrimination (or separability measures) and calibration (or accuracy or goodness-of-fit measures). Discrimination measures quantify the degree of separability between the two classes assessed, and calibration measures quantify how close the estimated value deviates from the observed value. See Baesens et al. (2016) for a detailed discussion. Before giving our overview of the typical performance measures, note the difference between goodness-of-fit and predictive accuracy. Goodness-of-fit is how well a model can predict data points that were used to estimate its parameters, whereas predictive accuracy is how well a model can predict new data points. To assess the latter, a data set is usually split into a so-called training set and a test or validation set. We have divided the discussion of the metrics or measures into two subsections, namely popular measures that are, in our opinion, frequently used in practice and other measures that seem to be less popular.

4.1. Popular Measures

The confusion matrix, or error matrix, and metrics derived from it are widely used in practice to assess the prediction performance of binary classifiers. The matrix is frequently cited on the internet and in reliable sources such as James et al. (2021) and Baesens et al. (2016). Consider the confusion matrix depicted in Table 1. Since we are working with binary classification, positive and negative corresponds to the target value (an event occurred or not), the columns represent the actual values, and rows represent the predicted values. Then, TP is the true positive where a positive predicted value matches the actual

one, TN is the true negative where a negative predicted value matches the actual one, FP is a false positive (Type 1 error) where the predicted value is positive but actually negative, and FN is a false negative (Type 2 error) where the predicted value is negative but actually positive.

Table 1. Confusion or error matrix as a metric of prediction performance.

Predicted	Actual	
	Positive	Negative
Positive	True positives (TP)	False positives (FP)
Negative	False negatives (FN)	True negatives (TN)

Note that for the models considered here, some cut-off values or thresholds have to be determined to obtain the above matrix once the model has been fitted. From this matrix, a number of other metrics may be derived, namely

Accuracy (A) is defined as $A = \frac{TP+TN}{TP+FP+TN+FN}$ and is the proportion of the binary values that were predicted correctly.

Precision (P) is defined as $P = \frac{TP}{TP+FP}$ and gives the proportion of predicted positives that were truly positive. This metric is preferred if we want to be very sure of our prediction; for example, we do not want to predict a customer defaulting on a loan incorrectly.

Recall (R) or the true positive rate is defined as $R = \frac{TP}{TP+FN}$ and is the proportion of actual positives that were predicted correctly. This metric is usually used when we want to capture as many positives as possible; for example, when predicting the presence of cancer, we want to capture the disease even if we are not very sure. Note that in the statistical literature, recall is sometimes referred to as sensitivity (see James et al. 2021).

Specificity (S) is defined as $S = \frac{TN}{FP+TN}$ and is the proportion of actual negatives that were predicted correctly. Note that $1 - S = \frac{FP}{FP+TN}$ is the false positive rate which is the proportion of negative cases that were incorrectly predicted as positives.

Probability of correct classification (PCC) is defined as $PCC = \left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP} \right) / 2$ and measures the performance of the classifier detecting either of the classes. Again, this measure can also be used to select a threshold which maximises the PCC .

The F1 score $F1 = 2 \left(\frac{P \times R}{P+R} \right)$ is the harmonic mean of recall and precision and obtains its maximum when recall equals precision. So, F1 tries to find a model with good precision and recall; for example, if you evaluate the credit worthiness of clients applying for a certain product, you want to be sure that the client is creditworthy, and you want to identify as many such clients as possible. The F1 score manages this trade-off. The measure is also frequently used to obtain the best threshold or cut-off value that results in the best F1 score. Since the F1 score gives equal weight to precision and recall, the score is often adjusted by weighting the importance of the two metrics. Many other metrics derived from the confusion matrix are possible, such as variants of the F-score, the Matthews correlation coefficient, Fowlkes-Mallows index, and diagnostic odds ratio (see Prorokowski (2019)).

The Gini coefficient ($Gini$) is one of the most frequently used metrics for evaluating the classification performance of any model used for binary classification tasks. Contrary to the above-mentioned metrics, the Gini does not rely on the choice of a threshold. The fact that this measure is invariant to threshold choice is important since typically financial institutions employ different thresholds in different applications. For example, the threshold used for assigning credit limits or for deciding on the type of credit card to issue is not the same. The Gini coefficient is defined as $Gini = 2(AUC) - 1$, where AUC is the area under the receiver operating characteristic ROC curve. The latter curve is obtained by plotting the true positive rate (also sensitivity or recall) against the false positive rate (also 1-specificity) for a range of thresholds or cut-off values. This provides a direct link between the confusion matrix and the construction of the Gini coefficient as outlined below.

Once the *ROC* curve is determined, the *AUC* tells us how much the model, regardless of the chosen threshold, is able to distinguish between positive and negative cases and therefore is a measure of the degree of separability between the two classes. The *AUC* typically varies between 0.5 (a model with no separation power or a ‘random’ model) and 1 (a model with perfect separation power), and the Gini typically varies between 0 (no separation) and 1 (perfect separation). The Gini coefficient is a way to adjust the *AUC* to simplify interpretation; for example, a perfectly random model has a Gini of 0, a perfect model of a Gini of 1, and a perfectly reversing model of a Gini of -1 . Negative Ginis are indicative of something that has seriously gone wrong in the modelling process, for example, when predicting the wrong class (predicting the non-events instead of the events). Note that *AUC*, and therefore Gini, is both threshold and scale invariant and not affected by over-sampling. For more detail regarding the use of Gini in practice, the interested reader is referred to Engelmann and Rauhmeier (2006), Siddiqi (2012), and (SAS Institute Inc. 2010).

Remark: *The metrics A, P, R, PCC, and F1 should be interpreted carefully when dealing with imbalanced data sets where the event rate is typically small, i.e., when the number of actual positives is much smaller than the actual negatives. Therefore, ‘good’ values for these metrics are dependent on the event rate, which is equal to the number of events (positives) divided by the number of observations. For example, if the event rate is 2% and the accuracy is 90%, then the accuracy is poor.*

4.2. Other Measures

Goodness-of-fit

Maximum likelihood estimation is used to estimate the parameters in the logistic regression model and finds the minimal discrepancy between the observed response and the predicted response. The resulting summary measure of this discrepancy is known as the deviance (see McCullagh and Nelder (1989)). The Akaike information criteria (AIC) and the Schwarz-Bayesian Information Criteria (SBIC) can be useful for comparing the goodness-of-fit of models. Both these information criteria are fairly simple derivations from the deviance, adjusted for sample size and a number of predictors. However, the above-mentioned measures have no standard of magnitude, and there are no statistical tests for these indices and no cut-off for what constitutes a good fit. So, they are occasionally used to compare non-nested models, i.e., models that do not have the same cases and where one model has a subset of predictors from the other model. When models are nested, such as in our case, there is no need for AIC and SBIC, and the likelihood ratio (difference in deviances) can be used as a statistical test for goodness-of-fit. Note that AIC and SBIC give the modeller an indication of how well the model fits for a specific misclassification cost, while the Gini gives you an indication of your model’s separation performance on average across all misclassification costs. Misclassification costs are mostly assumed to be equal, but this is not the case in imbalanced problems where the misclassification of a (rare) event could be worse than the misclassification of a non-event.

Other goodness-of-fit statistics that are often used in practice (see Baesens et al. (2016) and Prorokowski (2019)) include the Hosmer–Lemeshow, Spiegelhalter, Kolmogorov–Smirnov, and Vasicek statistics as well as the Brier score. Note that these quantities do not cater for misclassification cost.

R-square measures

In logistic regression, there is no true R-squared value as there is in ordinary least squares regression; however, the deviance (see McCullagh and Nelder (1989)) is often used as the equivalent of the sum of squared errors in linear regression and many R-square measures have been based on it. Shtatland et al. (2000) analysed a number of deviance-based R-square measures and argued that the measures should be used simultaneously when comparing the quality of the fit of logistic regression models. They motivate

its use for its similarity to the very popular use of the R-square statistic in linear regression and claim that the measures have some advantages compared to popular information criteria such as AIC and SBIC in terms of interpretability. Allison (2014) discusses the shortcomings of these measures in a predictive power context and recommends that the measures used by McFadden and Zarembka (1974) and Tjur (2009) be used instead.

The H measure (H)

The H-measure was proposed by Hand (2009) as an alternative for *AUC*. This measure satisfies a criterion for coherence of performance measures that the *AUC* does not. According to Hand (2009), the *AUC* is a coherent measure of separability between classes since it is based solely on the order of scores of the objects and not on the numerical values of those scores. He argues further that separability is not the same as classification performance since separability appears to ignore the central role of the classification threshold. Implicit in the *AUC* is a hidden assumption that each rank of the objects is equally likely to be chosen as the threshold. This is an unrealistic assumption in most practical applications, and he then proposes the H-measure, which takes into account the context of the application without requiring a rigid value of relative misclassification cost to be set. In a recent paper, Hand and Anagnostopoulos (2022) address queries that users have raised about the measure, including questions about its interpretation, the choice of misclassification cost weighting function, and other practical concerns. Of course, defining a suitable misclassification cost function for a specific problem is not an easy task and is open to criticism. Although this measure has been promoted for some time, it has not gained popularity among practitioners. One of the reasons is that it has not been taken up in regulatory guidelines and that the Gini is often preferred in practice despite the criticism against its use (see Baesens et al. (2016)).

4.3. Summary

Because of the multitude of metrics, we have decided to stick to the widely used Gini coefficient since it provides an indication of a model's ability to separate the classes on average across all classification costs. Although it is not a coherent measure of prediction performance, it is a coherent measure of separability between classes. In our simulation study below, our main objective is to check whether the fitting algorithms converge and estimate the true parameter values accurately. A secondary objective is to compare the binary classifiers in terms of their performance. Since no real application is at the core of the simulation study, misclassification cost is of secondary importance, and we will rely on the Gini as a coherent measure of separability between classes. However, when we analyse practical data sets, we will compare the prediction performance of the models by calculating F1 scores in the following way. Using the training data set, we will fit each model and calculate the F1 score as the maximum F1 score obtained by the particular model over a range of possible cut-off values. Then, the cut-off value corresponding to the maximum F1 score, obtained on the training set, is used to obtain the F1 scores for the particular model on the validation set.

5. Simulation Study

To test the accuracy of our fitting routines in terms of correctly estimating the model parameters, we conducted a simulation study where we generated data according to model (3). We considered sample sizes of $N = 500, 1000, 1500$, and $K = 10$ predictor variables, each generated independently as standard normal variables. The model parameters were set as follows: $\beta_1 = 1$, $\beta_2 = 2$, $\beta_3 = -3$, $\beta_4 = 1$, $\beta_5 = 1$, $\beta_{12} = 1$, $\beta_{13} = 1$, $\beta_{23} = 2$, $\beta_{910} = -2$, and all other β 's as zero. The binary responses Y_n values were then generated as follows:

$$Y_n = \begin{cases} 1 & \text{if } u_n \leq p_n \text{ for } n = 1, \dots, N \\ 0 & \text{otherwise} \end{cases}$$

where the u_n 's are generated as standard uniform variables ($u_n \sim U(0,1)$). Note that when we minimise the sum of logit loss, all the zeros should be replaced by minus ones.

We repeated the process 200 times, and to each of the 200 data sets, we fitted LR, LRI and 2- and 4-factor LFMs. We used maximum likelihood estimation in all cases, but for FMs, in order to numerically check the equivalence of the parameter estimates from (8) and (9), we included minimising the sum of logit loss. It also served as a further check for the accuracy of our algorithmic implementations. The two- and four-factor LFMs fitted with logit loss will be denoted by LLFM2 and LLFM4, respectively, and maximum likelihood fits by MLEFM2 and MLEFM4, respectively.

Tables 2, 3 and 4 contain the average, standard deviation and mean squared error (MSE) of the fitted coefficients of the predictor variables as obtained over the 200 data sets. Since the results for the different sample sizes were similar, we will only present the results for $N = 1000$ and make some remarks about the other sample sizes. Note that the bias may be obtained as the difference between the true parameter value and the average obtained over simulation runs (see Table 2). Since the data sets were generated according to model (3), one would expect that LRI should perform best, in terms of lowest bias and variance, followed by the LFMs and then LR. However, excluding LR, all LFMs obtained the lowest bias and variance and hence the smallest mean squared error, with the two-factor logistic factorisation machine being the best. The LR estimates have the lowest variance of all but are extremely biased, as is expected. When inspecting the coefficient estimates obtained for the individual simulation runs, we were not surprised, due to the proof in (7), to find that up to eight decimals, the logit loss and maximum likelihood fit of both the two- and four-factorisation machines gave the same coefficient estimates for each of the data sets. This holds for the coefficients of the predictor variables, as well as for the inner product approximations (5) of the coefficients of the interaction terms. At a summative level, the results in Tables 2, 3, and 4 confirm this as well. An explanation for the better performance of the factorisation machines is that fewer parameters have to be estimated, causing less variation and hence higher accuracy. Note that LR has the lowest MSE for the predictor coefficients but exhibits the most bias for the non-zero predictor coefficients and is, as expected, clearly not the preferred model.

Table 2. Average of predictor coefficient estimates over 200 repetitions for $N = 1000$ for logistic regression (LR), logistic regression with interaction (LRI), two factor factorisation machine based on logit loss (LLFM2) and maximum likelihood (MLEFM2), and four factorisation machines (LLFM4 and MLEFM4).

Betas	True Value	LR	LRI	LLFM2	MLEFM2	LLFM4	MLEFM4
0	0	0.225	0.008	0.014	0.013	0.012	0.013
1	1	0.576	1.157	1.075	1.075	1.136	1.136
2	2	1.046	2.321	2.155	2.155	2.278	2.278
3	−3	−1.578	−3.486	−3.240	−3.240	−3.419	−3.419
4	1	0.561	1.171	1.092	1.092	1.149	1.149
5	1	0.565	1.172	1.080	1.080	1.149	1.149
6	0	0.002	0.009	0.005	0.005	0.008	0.008
7	0	−0.005	0.002	0.002	0.002	0.000	0.000
8	0	0.003	−0.005	−0.007	−0.007	−0.006	−0.006
9	0	−0.008	0.003	−0.004	−0.004	0.003	0.003
10	0	0.005	0.006	0.011	0.011	0.007	0.007

Table 3. Standard deviation of predictor coefficient estimates over 200 repetitions for $N = 1000$ for logistic regression (LR), logistic regression with interaction (LRI), two factor factorisation machine based on logit loss (LLFM2) and maximum likelihood (MLEFM2), and four factorisation machines (LLFM4 and MLEFM4).

Betas	LR	LRI	LLFM2	MLEFM2	LLFM4	MLEFM4
0	0.082	0.158	0.129	0.129	0.148	0.148
1	0.088	0.154	0.136	0.136	0.149	0.149
2	0.111	0.256	0.225	0.225	0.248	0.248
3	0.134	0.345	0.311	0.311	0.328	0.328
4	0.093	0.169	0.153	0.153	0.164	0.164
5	0.095	0.170	0.147	0.147	0.164	0.164
6	0.078	0.148	0.127	0.127	0.142	0.142
7	0.089	0.147	0.130	0.130	0.143	0.143
8	0.080	0.143	0.127	0.127	0.136	0.136
9	0.095	0.154	0.135	0.135	0.149	0.149
10	0.087	0.154	0.138	0.138	0.149	0.149

Table 4. Mean squared error of predictor coefficient estimates over 200 repetitions for $N = 1000$ for the logistic regression and factorisation machine models.

Betas	LR	LRI	LLFM2	MLEFM2	LLFM4	MLEFM4
0	0.057	0.025	0.017	0.017	0.022	0.022
1	0.188	0.049	0.024	0.024	0.041	0.041
2	0.921	0.169	0.074	0.074	0.139	0.139
3	2.039	0.355	0.154	0.154	0.284	0.284
4	0.201	0.058	0.032	0.032	0.049	0.049
5	0.198	0.058	0.028	0.028	0.049	0.049
6	0.006	0.022	0.016	0.016	0.020	0.020
7	0.008	0.021	0.017	0.017	0.021	0.021
8	0.006	0.020	0.016	0.016	0.018	0.018
9	0.009	0.024	0.018	0.018	0.022	0.022
10	0.008	0.024	0.019	0.019	0.022	0.022

As far as the estimation of the interaction coefficients is considered, we will present the results for LRI and only MLEFM2 below since the other LFMs performed similarly. The MSE for the interaction coefficients are given in Table 5 for LRI and in Table 6 for MLEFM2. Again, the results for LLFM2 and MLEFM2 were the same, and we observed slightly bigger MSE's for the four-factor LFMs (MLEFM4 and LLFM4). When comparing the MSEs of the LRI and MLEFM2 fit, remarkably, the MSE of the MLEFM2 fits is much smaller than that of LRI despite the fact that the latter model was used to generate the data sets. This is testimony to the remarkable ability of factorisation machines to approximate this model with fewer parameters. In this case, LRI had to estimate $1 + K + K(K-1)/2$, i.e., 65 parameters while MLEFM2 only had to estimate $1 + K + KG$, i.e., 31 parameters.

Table 5. Mean squared error of interaction coefficient estimates for LRI ($N = 1000$).

	2	3	4	5	6	7	8	9	10
1	0.06821	0.08044	0.02662	0.02950	0.02522	0.02432	0.02343	0.02842	0.02549
2		0.20197	0.03057	0.03199	0.03260	0.03029	0.02791	0.03498	0.03719
3			0.04024	0.04144	0.04289	0.03534	0.04172	0.04525	0.04125
4				0.03165	0.02307	0.02285	0.02166	0.02837	0.02934
5					0.02131	0.02465	0.02273	0.02747	0.02052
6						0.02056	0.02078	0.02142	0.02436

7	0.01717	0.02657	0.02172
8		0.02304	0.02339
9			0.16936

Table 6. Mean squared error of interaction coefficient estimates for MLEFM2 ($N = 1000$).

	2	3	4	5	6	7	8	9	10
1	0.00510	0.00747	0.00001	0.00003	0.00009	0.00001	0.00001	0.00010	0.00017
2		0.02911	0.00005	0.00008	0.00040	0.00009	0.00005	0.00014	0.00013
3			0.00006	0.00008	0.00031	0.00005	0.00005	0.00034	0.00014
4				0.00000	0.00000	0.00000	0.00000	0.00009	0.00028
5					0.00000	0.00000	0.00000	0.00011	0.00006
6						0.00000	0.00000	0.00009	0.00011
7							0.00000	0.00015	0.00006
8								0.00008	0.00014
9									0.02429

We also recorded the time (in seconds) that the algorithms took to converge, as well as the Gini coefficients obtained for each of the fits. The results appear in Tables 7 and 8. Clearly, the factorisation machines took a longer time to converge than the logistic regression fits. Interestingly, compared to the four-factor factorisation machines, the two-factor factorisation machines took less time on average to converge but seem to struggle more in some cases (see the larger maximum convergence times). As far as the Gini results are concerned, LR is way-off, as expected, while the other techniques give fairly similar results.

Table 7. Time that the estimation algorithms of the models took to converge over the simulation runs.

	Average	Standard Deviation	Minimum	Maximum
LR	0.057	0.015	0.032	0.185
LRI	0.095	0.065	0.062	0.825
LLFM2	5.807	7.158	0.364	28.634
MLEFM2	5.310	6.526	0.336	37.872
LLFM4	8.572	3.820	1.013	21.079
MLEFM4	8.491	3.917	1.101	20.204

Table 8. Gini coefficients obtained by the models over simulation runs.

	Average	Standard Deviation	Minimum	Maximum
LR	0.750	0.023	0.679	0.803
LRI	0.935	0.011	0.904	0.964
LLFM2	0.926	0.014	0.846	0.956
MLEFM2	0.926	0.013	0.844	0.956
LLFM4	0.933	0.011	0.903	0.962
MLEFM4	0.933	0.011	0.902	0.962

To test the predictive ability of the various methods considered here, we repeated the above simulation study by changing the underlying model that was previously used to generate the data set. We also randomly split the generated data sets in a 70% training set and a 30% validation set. When generating the data, we used the same specification as before but included an extra (standard normal) predictor variable with a coefficient of two. We then excluded the variable when fitting the algorithms to the data sets. The algorithms were fitted to the training set only and then used to predict the validation set. We again used Gini as our performance measure. Tables 9 and 10 contains the average fitted

and predicted Ginis obtained as well as the standard deviation, minimum and maximum. In order to evaluate the performance of the models, in this setting, with a popular tree-based method, we include standard random forests (RF) as a competitor. Consider the results for the training set in Table 9. Except for LR and RF, the other algorithms give fairly similar Gini estimates with the logit loss, and the maximum likelihood fits again, providing, almost always, exactly the same Ginis for each data set. Here, LR performs the worst and RF the best. For the validation set (Table 10), the predicted Ginis of LRI and the two-factor factorisation machines are best. Interestingly the four-factor factorisation machines break down, which might be due to overfitting the training data set and then being unable to generalise to the validation set.

Table 9. Fitted Ginis on the training set obtained over 200 simulation runs.

	Average	Standard Deviation	Minimum	Maximum
LR	0.689	0.031	0.605	0.763
LRI	0.877	0.018	0.821	0.926
LLFM2	0.859	0.021	0.791	0.913
MLEFM2	0.859	0.022	0.768	0.913
LLFM4	0.873	0.018	0.816	0.924
MLEFM4	0.873	0.018	0.816	0.924
RF	0.999	0.001	0.999	1.000

Table 10. Predicted Ginis for the validation set obtained over 200 simulation runs.

	Average	Standard Deviation	Minimum	Maximum
LR	0.671	0.045	0.546	0.801
LRI	0.800	0.035	0.658	0.886
LLFM2	0.816	0.036	0.657	0.901
MLEFM2	0.816	0.036	0.676	0.901
LLFM4	0.245	0.235	-0.290	0.789
MLEFM4	0.251	0.222	-0.231	0.788
RF	0.696	0.049	0.537	0.798

The same might be true for RF; however, RF slightly outperforms LR on the validation set but performs poorly relative to LRI and the two-factor FMs. This is expected since the data sets are generated from an underlying model that favours LRI and FMs. The results in Table 10 show that the choice of G , the number of factors in the LFM, is crucial and an inappropriate choice can be catastrophic, as indicated by the negative Ginis in the table. A remark about this will be made in the last section.

Remark: *The results for the other sample sizes were similar, with the larger sample sizes providing better results in terms of less variance and bias observed.*

6. Analysis of Prediction Performance on Some Data Sets

In this section, we investigate and compare the prediction performance of LFMs with LRI and LR by analysing three data sets. Various studies have confirmed that FMs perform well in recommender system applications and in sparse data settings, but little is known about their performance in the typical binary classification problems banks deal with. In this section, we will investigate two such applications where logistic regression has a stronghold, namely fraud identification and credit scoring. However, before we discuss the latter case studies, we decided to first illustrate the behaviour and prediction performance of FMs on an artificially constructed binary data set that is typically found in recommender system applications. This will set the scene for what is to follow in the banking applications. Note that we now drop the notation that distinguished between logistic

loss and maximum likelihood and will subsequently refer to the two-factor LFM as LFM2 and the four-factor LFM as LFM4 and so on.

6.1. Artificial Recommender System Example

In this subsection, an artificially constructed binary data set as given in Table 11 is studied. Theoretically, 20 users (denoted by U_1 to U_{20}) may record their preferences for 20 items (denoted by I_1 to I_{20}). Users could be customers indicating their propensity to buy certain items, recording their preferences for movies, or selecting political candidates they would vote for. Another application could be the identification of the next best product to sell, and Table 11 could then represent the products that a customer owns (indicated by ones) or does not own (indicated by zeros).

The shaded values in Table 11 indicate missing observations, in other words, values that are not available and have to be predicted from the observed values. In our case, the positive responses of the users have been organised in five blocks containing four users each, and assuming no missing values, a clear structure of identical users is visible in Table 11. For example, users U_1 to U_4 all picked items I_1 to I_4 , and users U_{13} to U_{16} all picked items I_{13} to I_{16} . Of course, should the labels of users and items be randomised, the block structure of identical users will not be clearly visible. The shaded values represent 120 of the 400 values that were randomly removed from the data set and that we want to predict. Can an LFM model the underlying associations between the users using the 280 observations and then predict the missing values correctly?

Table 11. The full binary ratings data set where the shaded zeros and ones indicate the removed ratings.

	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	I_{16}	I_{17}	I_{18}	I_{19}	I_{20}
U_1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
U_2	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
U_3	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
U_4	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
U_5	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
U_6	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
U_7	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
U_8	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
U_9	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
U_{10}	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
U_{11}	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
U_{12}	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
U_{13}	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
U_{14}	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
U_{15}	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
U_{16}	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
U_{17}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
U_{18}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
U_{19}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
U_{20}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Before we can fit an LFM to the incomplete data set (without the missing values), we have to input it into an appropriate format understood by the algorithms. The data set is therefore encoded using one hot coding (see Slabber et al. (2021) or Rendle (2012)), which means that the two predictor variables (users and items) will escalate to 40 nominal predictors (20 users plus 20 items). If we want to fit an LFM with two factors (LFM2), this will require the estimation of 121 ($1 + K + KG = 1 + 40 + 40 \times 2 = 121$) parameters. This is a significant reduction compared to fitting a full two-way LRI where $821 \left(1 + K + \frac{K(K-1)}{2} = 1 + 40 + 20(39) = 1 + 40 + 780 = 821\right)$ parameters need to be estimated. Note

that for LRI, the number of parameters to be estimated is more than the number of observations, which will result in failure; however, an unweighted LFM2 model fitted to the 280 observations obtained a near-perfect fit and prediction. For example, the estimated and predicted probabilities ($\hat{p}_n^{LFM2}$) obtained for the actual ones varied between 0.999999981 and 0.999999998 and for the actual zeros between 0.000000000 and 0.000000046. So rounded to the nearest integer, we will estimate and predict Table 11 perfectly. Therefore, if we construct a confusion matrix by using any threshold (say) between 0.0001 and 0.9999, we will obtain 100% for all the confusion matrix-based measures given in Section 4.1. Impressively, with only 70% of the binary values available, LFMs were able to pick up the block structure, and resulting associations between users and items and make a perfect prediction.

To test the power of the method further, we removed 50% of the ratings. When rounding the estimated and predicted probabilities to the nearest integer, we obtained a perfect fit but predicted two of the removed zeros as ones. However, when 70% of the values in the original set were removed, the LFM models fitted and predicted poorly. In all the above-mentioned cases studied, the LR and LRI models performed poorly. LR failed to pick up the associations between users and items because these models do not contain any interaction terms. LRI failed completely due to the input matrix being singular, caused by the fact that the number of predictors outnumbered the number of observations. LFMs perform well because the factor structure enables the individual factors to borrow strength from each other to pick up the associations between users and items (see Rendle (2010) and Slabber et al. (2021)).

Remarks.

- (a) *When inspecting the results of the LFM fits, we noticed that, when all the observations on the edges of a block structure are removed, LFMs struggle to predict those values correctly.*
- (b) *A straightforward implementation of RF on this problem provided poor results. This is expected because it is well-known that RFs struggle with sparse data sets. Of course, research on improving RFs to cater for these types of problems are ongoing (see, e.g., Hariharan et al. 2017 and Wang et al. 2018). However, given the fact that LFM2 provide a perfect fit on both training and validation sets, it is clearly the winner in this case.*

6.2. Credit Card Fraud Example

The contents and context of this Kaggle data set are described in Kaggle (2021). In brief, it contains data on $N = 284,807$ transactions, of which 492 are fraudulent ($Y_n = 1$) and the remainder clean ($Y_n = 0$). There are $K = 30$ regressors, of which 28 are principal component transformations of features that are not detailed due to confidentiality issues; the other two regressors are described as 'Time' and 'Amount'. Here, we refer to the regressors simply as $X_1, X_2, \dots, X_{30}$. This is a large data set, and the illustration below follows the modelling paradigm of dividing the data set into training and validation sets. The rows of the data set were permuted randomly, and then half of them were put into the training set and the remainder into the validation set. Thus, the training data set has $N_{train} = 142,404$ observations with $N_{train,1} = 245$ frauds and $N_{train,0} = 142,159$ cleans. This data set is highly unbalanced since the fraction of frauds is only 0.172% of the total. As described in Section 2, expressions (8) and (9), the assignments of the weights in the fitting criterion are used to address this matter. We take $w_n = 1/2N_{train,1}$ if $Y_n = 1$ and $w_n = 1/2N_{train,0}$ if $Y_n = 0$. Then, the totals of the weights for the frauds and the cleans are $1/2$ each, and the overall total of all the weights is 1. This enables the small number of frauds to play a meaningful role in the model training. The results obtained for this data set are given in Tables 12 and 13. The Ginis obtained by the models for the training and validation data sets are given in Table 12, and the F1 scores are in Table 13. Since we are dealing with a highly imbalanced data set, we fitted a balanced random forest (BFR)

(see Agusta 2019). Consider the results presented in Table 12. Very high Ginis are obtained by all models on the training set; however, on the validation set, there is a more than 15% drop in separation performance of BRF and the LFM. This indicates that the models overfit the training set and fail to generalise to the validation set (see, e.g., Engelmann and Rauhmeier (2006) and James et al (2021). Interestingly, the LRI fit does not exhibit this substantial drop in separation performance and obtains similar results as LR. Standard SAS output provides the 95% confidence interval of the LR Gini on the validation set as (0.8942, 0.9644), and since this model is parsimonious compared to LRI, it will be preferred in practice. However, when the F1 scores in Table 13 are considered, the practitioner could make a different conclusion since the best F1 score on the validation set is obtained by BRF (F1 = 0.860) and the second best by LR (0.803). BRF achieves perfect separation on the training set (F1 = 1.000), while the LR score of 0.815 is close to what LR achieved on the validation set. In the end, the consistency achieved by LR in terms of both Gini and F1 scores, as well as the interpretability advantage, will sway the practitioners' decision in its favour. Note that we did not calculate F1-scores for the LFM4 and LFM6 models since we felt that the parsimonious model (LFM2) is of more interest to the practitioner and gave similar performance to the others as seen in Table 12.

Table 12. Ginis obtained by the various models on the full data set.

Modelling Technique	Train	Validation	Number of Parameters to Estimate
LR	0.987	0.929	31
LRI	0.994	0.925	466
LFM2	0.998	0.824	91
LFM4	0.999	0.835	151
LFM6	0.999	0.741	211
BRF	0.996	0.737	

Table 13. F1 scores obtained by the various models on the full data set.

Modelling Technique	Train	Validation
LR	0.815	0.803
LRI	0.648	0.507
LFM2	0.782	0.752
BRF	1.000	0.860

In retrospect, the better performance of LR is no surprise since 28 of the 30 predictor variables resulted from a principal components analysis (PCA) where each principal component is orthogonal to each other. Therefore, it is highly unlikely that LR will be outperformed by LFM, LRI, and even BRF since the presence of interaction terms is unlikely.

Since LFM has distinguished themselves as performing well in data sets where the number of predictors is more than the number of observations, we randomly selected 200 fraud cases and 200 non-fraud cases from the data set. We then split the data set into a training set containing 100 fraud cases and 100 non-fraud cases and allocated the remaining observation to the validation set. The results are given in Table 14 (Gini coefficients) and Table 15 (F1 scores). Note that LRI could not be fitted because the number of parameters to be estimated (466) is much larger than the number of observations (200) resulting in the input matrix being singular.

Table 14. Ginis obtained by the models on the reduced data set.

Modelling Technique	Train	Validation	Number of Parameters to Estimate
LR	0.995	0.827	31
LRI			
LFM2	0.999	0.816	91
LFM4	0.999	0.857	151
LFM6	0.999	0.853	211
BRF	0.999	0.843	

Table 15. F1 scores obtained by the models on the reduced data set.

Modelling Technique	Train	Validation
LR	0.989	0.900
LRI		
LFM2	1.000	0.881
BRF	1.000	0.916

When we compare the results in Table 12 to that in Table 14, it is interesting to note that LR separation performance drops dramatically. Although all models seem to overfit, the separation obtained is good, with LFM4 having the highest Gini but by a small margin. This supports the finding by Rendle that FMs generally perform very well when the number of predictors outnumbers the number of observations, i.e., when $K > N$. As far as Table 14 is concerned, the F1 scores obtained by the models on the training and validation sets are close, with BRF marginally the best performer.

Remark.

- To run PROC OPTMODEL, we had to adapt the SAS Config file by changing MEMSIZE from 2G to 10G; otherwise, one gets stuck in memory problems, especially when fitting LFM6.*
- We considered various other data sets where the number of observations was reduced by keeping the number of frauds and non-frauds equal. As the sample size gets smaller, the advantage of the FMs over LR, LRI, and BRF becomes more prominent.*

6.3. Credit Scoring Example

In this section, we compare logistic factorisation machines and logistic regression with respect to their ability in separating the defaulters from non-defaulters, and we use a large proprietary data set obtained from a bank. Note that the information on the predictor variables is classified as confidential. The data set was previously analysed by De Jongh et al. (2015), and we will use a reduced data set resulting from their study that contains 38 predictor variables and 335,523 observations, of which 10,267 (3.06%) defaulted. Because of the larger number of zeros, we found that the unweighted fit performed as well as the weighted fit and will only present the results for the unweighted fits. For this data set, we have available both the original and the weights-of-evidence (WoE) transformed variables. The original data set comprised 1,294,811 observations and 802 variables that were reduced through a process of variable selection and elimination of correlated variables. The Gini statistic obtained after fitting an LR model to the WoE transformed variables of the reduced data set was 0.829, as seen in Table 8 of the above-mentioned paper. Below, we obtain similar values on a random 50% split of the reduced data set in a training and validation set.

Before fitting the models, we standardised the original variables, which is recommended by Frost (2019) if one expects the presence of interaction terms. The fits on the original standardised and the WoE transformed data set are given in Tables 16 and 17 below. As mentioned previously, we fitted the model on the training data set while the

performance is evaluated on the holdout portion (validation data set). The Ginis obtained for the data sets are given in Table 16, and the F1 scores are in Table 17. As in the previous example, because we are dealing with an imbalanced data set, we fitted a balanced random forest (BFR).

Table 16. Ginis obtained by the models for the credit scoring data set.

Modelling Technique	Standardised Original Variables		WoE Transformed Variables		Number of Parameters
	Train	Validation	Train	Validation	
LR	0.772	0.783	0.814	0.824	39
LRI	0.805	0.800	0.847	0.825	742
LFM2	0.797	0.802	0.828	0.829	115
LFM4	0.804	0.804	0.832	0.830	191
LFM6	0.809	0.807	0.836	0.830	267
BRF	0.999	0.883	0.999	0.862	

Table 17. F1 scores obtained by the models for the credit scoring data set.

Modelling Technique	Standardised Original Variables		WoE Transformed Variables	
	Train	Validation	Train	Validation
LR	0.383	0.386	0.398	0.399
LRI	0.400	0.386	0.433	0.401
LFM2	0.393	0.391	0.411	0.406
BRF	0.999	0.391	0.977	0.402

Consider the results in Table 16. In both the original standardised and the WoE case, the highest Gini on the validation data set is obtained by BRF, followed by the LFM models and then the LR and LRI models. Except for BRF, where the Gini on the training set is considerably larger than that obtained on the validation set, all models seem to generalise well and exhibit no signs of overfitting. The consistency of the results of the latter models on both the training and validation set indicates the stability of the models (see, e.g., Engelman and Rauhmeier (2006)). Now consider the results in Table 17 where the F1 scores are presented. As in the previous example, we omitted the four- and six-factor LFMs since their performance are similar to the two-factor LFM, (refer to Table 16). As seen in Table 17, BRF clearly achieve the best F1 score on the training data set (for the original and WoE variables) but perform similar to the other models on the validation sets. So, as far as predictive performance is concerned, there is little to choose between these models.

As motivated below, in the case of the WoE transformed data, one would expect that no interactions between variables are present; hence one would expect that the models studied here should perform similarly. However, in this example, this also seems to hold for the standardised original variables. This concurs with a remark made by Crook (2014) that he seldomly found significant interaction terms when building retail credit scorecards. So, in this case, practitioners will most probably select the easily interpretable and simpler LR model for further analysis and implementation.

Remarks on WoE Transformation

Banks almost always use the WoE transformation to build retail credit scorecards using ordinary logistic regression. This so-called balance scorecard methodology (see, e.g., Siddiqi (2017)) relies on fitting logistic regression models where the predictor variables are typically binned in categories, and then the target variable is used to obtain a WoE value for each category of the predictor variables. Here, the WoE value, in a specific category, is obtained as the logarithm of the ratio of the percentage non-events and the

percentage events. There are three main advantages of this transformation in credit scoring: (a) outliers and missing values are grouped in classes and binned separately, (b) no dummy variables are needed since categorical variables are handled by WoE, and (c) WoE eliminates the need for linear transformations of non-linear relationships, such as log and square root. The WoE variables are constructed by means of a transformation applied to the categories resulting from the discretization, typically by means of classification trees, of the original inputs. In our opinion, logistic factorisation machines do not have clear advantages over the above-mentioned methods when the predictor variables are WoE transformed. The main reason is that the WoE should be monotonic for each variable. According to Zeng (2014), one criterion used for a good binning algorithm is that a logistic regression run on a single WoE variable should provide a slope of close to one and an intercept close to $\ln\left(\frac{N_0}{N_1}\right)$. Schaeben (2020) further argued that WoE is a special case of logistic regression when the predictor variables are jointly conditional independent, given the target variable. It is a required modelling assumption of weights-of-evidence to ensure its features and proper estimates of probabilities. Therefore, WoE-transformed predictor variables cannot reasonably include interaction terms, providing little scope for LRI or FMs to provide better prediction performance than LR. However, if the original (untransformed variables) are used, the inclusion of proper interaction terms in logistic regression compensates for the lack of joint conditional independence completely (see Schaeben (2014)).

According to Sharma (2011), WoE transformations usually work well in logistic models not containing interaction terms, and this lack of adaptation with respect to interacting variables is one of the main criticisms that may be made regarding the mentioned methodology. However, recently, Giner-Baixauli et al. (2021) extended the WoE-based methodology to include new WoE variables that capture interaction effects. The authors also show that the extended WoE approach enables the improvement of the predictive capability over the standard WoE-based methodology.

In conclusion, if the WoE transformation is applied, we are of the opinion that there is little scope for the application of LFM in retail credit-scoring contexts. However, in applications where the original untransformed data is used or when the number of predictors is more than the number of observations, its use as a challenger model is recommended.

7. Conclusions

In this paper, we introduced the reader to LFMs and compared their performance with logistic regression and random forests. We have implemented LFM fitting routines in SAS using PROC OPTMODEL, and since SAS is widely used by large companies, we trust that these routines will facilitate the application of LFMs commercially. A simulation study confirmed the accuracy of our routines, and by using two bank-related data sets, we illustrated the strengths and weaknesses of these models. In the process, we hope that we have succeeded in convincing the reader that LFMs are worthy competitors to random forests (BRF) and logistic regression (LR and LRI). The LFMs have clear advantages over the other methods in high-dimensional and sparse data settings and performed competitively on the banking data sets. In comparison to logistic regression (i.e., LR without interaction terms), it comes at the cost of many more parameters, which would be impractical from the model monitoring point of view. However, their inclusion as a challenger model is highly recommended.

As far as future research is concerned, we suggest that a detailed study be designed to compare the predictive performance of classifiers, including discrete choice models that address interactive effects, such as proposed by Ai and Norton (2003) and Jiang (2021). Ai and Norton (2003) studied the estimation and inference problems for interaction terms in logit and probit models. Jiang (2021) proposed a semiparametric-ordered response in which explanatory variables can interactively affect the ordered response dependent

variable of interest. Such a study should include a plethora of metrics for balanced and imbalanced data sets. Additionally, as far as the predictive accuracy of LFM is concerned, the choice of the number of factors G is important. This is worth pursuing further, along with the reduction in computational demands as well as the applicability in high-dimensional sparse data settings.

Author Contributions: Conceptualization, E.S., T.V., and R.d.J.; methodology, E.S., T.V., and R.d.J.; software, E.S.; validation E.S., T.V., and R.d.J.; formal analysis, E.S., T.V., and R.d.J.; investigation, E.S., T.V., and R.d.J.; resources, E.S., T.V., and R.d.J.; writing—original draft preparation, E.S., T.V., and R.d.J.; writing—review and editing, E.S., T.V., and R.d.J.; visualization, E.S., T.V., and R.d.J.; supervision, T.V., and R.d.J.; All authors have read and agreed to the published version of the manuscript.

Funding: This work is based on research supported in part by the Department of Science and Innovation (DSI) of South Africa. The grant holder acknowledges that opinions, findings, and conclusions or recommendations expressed in any publication generated by DSI-supported research are those of the authors and that the DSI accepts no liability whatsoever in this regard.

Data Availability Statement: The credit card fraud data set analysed in Section 6.2 can be found here: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud> (accessed on 20 February 2023).

Acknowledgments: This research benefitted from insightful discussions with Hennie Venter and comments made by Charl Pretorius and two referees.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- (Ai and Norton 2003) Ai, Chunrong, and Edward C. Norton. 2003. Interaction terms in logit and probit models. *Economics Letters* 80: 123–29.
- (Allison 2014) Allison, Paul D. 2014. Measures of fit for logistic regression. Paper presented at the Proceedings of the SAS Global Forum 2014 Conference. Washington, DC, USA. March 23–26, 2014.
- (Agusta 2019) Agusta, Zahra Putri. 2019. Modified balanced random forest for improving imbalanced data prediction. *International Journal of Advances in Intelligent Informatics* 5: 58–65.
- (Baesens et al. 2016) Baesens, Bart, Daniel Roesch, and Harald Scheule. 2016. *Credit Risk Analytics: Measurement Techniques, Applications, and Examples in SAS*. Hoboken: John Wiley & Sons.
- (Breiman 2001) Breiman, Leo. 2001. Random forests. *Machine Learning* 45: 5–32.
- (Crook 2014) Crook, Jonathan. 2014. *Personal Communication at the Fifth International Conference on Mathematics in Finance, Skukuza, Kruger National Park, South Africa, 24–29 August 2014.*
- (De Jongh et al. 2015) De Jongh, Riaan., Erika De Jongh, Marius Pienaar, Heather Gordon-Grant, Marien Oberholzer, and Leonard Santana. 2015. The impact of pre-selected variance inflation factor thresholds on the stability and predictive power of logistic regression models in credit scoring. *ORiON* 31: 17–37.
- (Engelmann and Rauhmeier 2006) Engelmann, Bernd, and Robert Rauhmeier. 2006. *The Basel II Risk Parameters: Estimation, Validation, and Stress Testing*. Berlin/Heidelberg: Springer Science & Business Media.
- (Frost 2019) Frost, Jim. 2019. *Introduction to Statistics: An Intuitive Guide for Analyzing Data and Unlocking Discoveries*. Statistics by Jim Publishing, State College, Pennsylvania, USA. ISBN 978-1-7354311-0-9.
- (Giner-Baixaui et al. 2021) Giner-Baixaui, Carlos, Juan Tinguaro Rodríguez, Alejandro Álvaro-Meca, and Daniel Vélez. 2021. Modelling Interaction Effects by Using Extended WOE Variables with Applications to Credit Scoring. *Mathematics* 9: 1903.
- (Gilpin et al. 2018) Gilpin, Leilani H., David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: An overview of interpretability of machine learning. Paper presented at 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), Turin, Italy, October 1–3, pp. 80–89.
- (Hand 2009) Hand, David J. 2009. Measuring classifier performance: A coherent alternative to the area under the ROC curve. *Machine Learning* 77: 103–23.
- (Hand and Anagnostopoulos 2022) Hand, David J, and Christoforos Anagnostopoulos. 2022. Notes on the H-measure of classifier performance. *Advances in Data Analysis and Classification*. <https://doi.org/10.1007/s11634-021-00490-3>
- (Hand and Henley 1997) Hand, David J, and William E. Henley. 1997. Statistical classification methods in consumer credit scoring: A review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 160: 523–41.
- (Hariharan et al. 2017) Hariharan, Siddharth, Siddhesh Tirolkar, Alok Porwal, Avik Bhattacharya, and Aurore Joly. 2017. Random forest-based prospectivity modelling of greenfield terrains using sparse deposit data: An example from the Tanami Region, Western Australia. *Natural Resources Research* 26: 489–507.

- (Hilbe 2009) Hilbe, Joseph M. 2009. *Logistic Regression Models*. Boca Raton: Chapman and Hall/CRC.
- (James et al. 2021) James, Gareth, Witten Daniela, Hastie Trevor, and Tibshirani Robert. 2021. *An Introduction to Statistical Learning with Applications in R*, 2nd ed. New York: Springer. Available online: https://doi.org/10.1007/978-1-0716-1418-1_1 (accessed on 20 February 2023).
- (Jiang 2021) Jiang, Yixiao. 2021. Semiparametric Estimation of a Corporate Bond Rating Model. *Econometrics* 9: 23.
- (Kaggle 2021) Kaggle. 2021. *Credit Card Fraud Detection Dataset*. Available online: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud> (accessed 12 June 2021).
- (Kleinbaum et al. 2005) Kleinbaum, David and Mitchel Klein. Regression. 2005. *Logistic Regression: A Self-Learning Text*. New York: Springer, p. 22.
- (Lessmann et al. 2015) Lessmann, Stefan, Bart Baesens, Hsin-Vonn Seow, and Lyn C. Thomas. 2015. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research* 247: 124–36.
- (McCullagh and Nelder 1989) McCullagh, Peter, and John A Nelder. 1989. Monographs on statistics and applied probability. *Generalized Linear Models* (second edition), Chapman and Hall (London and New York). <https://www.utstat.toronto.edu/~brunner/oldclass/2201s11/readings/glmbook.pdf> (accessed on 20 February 2023)
- (McFadden and Zarembka 1974) McFadden, Daniel, and Paul Zarembka. 1974. *Frontiers in Econometrics*. New York: Academic Press.
- (Prorokowski 2019) Prorokowski, Lukasz. 2019. Validation of the backtesting process under the targeted review of internal models: Practical recommendations for probability of default models. *Journal of Risk Model Validation* 13: 109–47.
- (Rendle 2010) Rendle, Steffen. 2010. Factorization machines. Paper presented at the 2010 IEEE International Conference on Data Mining, Sydney, NSW, Australia, December 13–17, pp. 995–1000.
- (Rendle 2012) Rendle, Steffen. 2012. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3: 1–22.
- (SAS Institute Inc. 2010) SAS Institute Inc. 2010. *Predictive Modelling Using Logistic Regression (SAS Course Notes)*. Cary: SAS Institution Inc.
- (Schaeben 2014) Schaeben, Helmut. 2014. A mathematical view of weights-of-evidence, conditional independence, and logistic regression in terms of Markov random fields. *Mathematical Geosciences* 46: 691–709.
- (Schaeben 2020) Schaeben, Helmut. 2020. Comment on “Modified Weights-of-Evidence Modeling with Example of Missing Geochemical Data”. *Complexity* 2020: 1–4.
- (Sharma 2011) Sharma, Dhruv. 2011. Evidence in favor of weight of evidence and binning transformations for predictive modeling. Available at SSRN 1925510. <http://dx.doi.org/10.2139/ssrn.1925510>
- (Shtatland et al. 2000) Shtatland, Ernest, S, Sara Moore, and Mary. B. Barton. 2000. Why we need an R-square measure of fit (and not only one) in PROC LOGISTIC and PROC GENMOD. Paper presented at the Proceedings of the Twenty-Fifth Annual SAS® Users Group International Conference, Cary: SAS Institute Inc., pp. 256–25. April 9–12, 2000, Indiana Convention Center, Indianapolis, Indiana
- (Siddiqi 2012) Siddiqi, Naeem. 2012. *Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring*. Hoboken: John Wiley & Sons.
- (Siddiqi 2017) Siddiqi, Naeem. 2017. *Intelligent Credit Scoring: Building and Implementing Better Credit Risk Scorecards*. Hoboken: John Wiley & Sons.
- (Slabber et al. 2022) Slabber, Erika, Tanja Verster, and Riaan De Jongh. 2022. Algorithms for estimating the parameters of factorisation machines. *South African Statistical Journal* 56: 69–89.
- (Slabber et al. 2021) Slabber, Erika, Tanja Verster, and Riaan De Jongh. 2021. Advantages of Using Factorisation Machines as a Statistical Modelling Technique. *South African Statistical Journal* 55: 125–44.
- (Tjur 2009) Tjur, Tue. 2009. Coefficients of determination in logistic regression models—A new proposal: The coefficient of discrimination. *The American Statistician* 63: 366–72.
- (Venter and De Jongh 2023) Venter, Hennie, and Riaan De Jongh. 2023. Variable selection by searching for good subsets. *Accepted for publication in the South African Statistical Journal*.
- (Wang et al. 2018) Wang, Qiang, Thanh-Tung Nguyen, Joshua Z. Huang, and Thuy Thi Nguyen. 2018. An efficient random forests algorithm for high dimensional data classification. *Advances in Data Analysis and Classification* 12: 953–72.
- (Zeng 2014) Zeng, Guoping. 2014. A necessary condition for a good binning algorithm in credit scoring. *Applied Mathematical Sciences* 8: 3229–42.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Chapter 5

Conclusion

In this thesis, we investigated factorisation machines (FMs) as a modelling technique and compared them with generalised linear models (GLMs). We demonstrated that FMs perform well in areas where GLMs are frequently used as the preferred modelling paradigm. In addition, FMs outperform GLMs in sparse data settings and when the number of predictor variables or features outnumbers the number of observations. We discussed some of the limitations of FM fitting routines available in open-source and commercially available software. Because of these limitations, we programmed our own fitting routines in SAS. In a regression context, these included coordinate descent and nonlinear programming algorithms that minimise the weighted sum of squared errors (WSSE) and the weighted sum of absolute deviations (WSAD). While WSSE is the most common loss function used to fit FMs, WSAD provides a new robust alternative for fitting FMs. We demonstrated that these routines are computationally accurate and efficient. In particular, we showed that the robust FM can pick up associations between predictors even in the presence of outliers. Also, when outliers are present the robust FM outperforms standard FMs and GLMs. In a logistic regression context, we used nonlinear programming to create two fitting routines. The first minimise a weighted sum of logit loss and the other maximise the weighted log-logistic likelihood. We proved that, given the correct encoding of the target variable, the two abovementioned optimisation algorithms yield the same parameter estimates. Compared to ordinary logistic regression, we concluded that FMs are worthy competitors, but outperform the former when the input matrix is sparse and when the number of predictors outnumbers the number of observations. We argued that LFMs are unlikely to outperform WoE logistic regression which is usually the preferred method for scorecard development. As a last remark, we hope that the papers contained in the thesis will stimulate interest in FMs among statisticians and model developers in business.

As far as recommendations for future research is concerned. We have developed tailor-made routines for fitting FMs in a regression and logistic regression context. This could also be done for other GLMs, for example, ordinal regression, Poisson regression, and multinomial logistic regression. Note that, the algorithms we developed in the second paper, although not tailor-made for ordinal logistic regression, performed well on such a problem. However, a tailor-made algorithm will provide the model builder with more confidence when applying FMs to a range of problems containing an ordinal target variable. In the introductory chapter, we mentioned several new variants of FMs that were introduced recently. To mention a few, these include Attentional FMs, Field-aware FMs, Bayesian FMs, Kernel FMs, Deep FMs, Co-FMS and convolutional FMs. The application of these in a business environment is unexplored and warrants thorough investigation in the near future. Another issue is the optimum choice of the number of factors to

use for a specific data set. In the cases we studied, it seemed that a choice of two factors generally leads to good generalisation, but will this hold in all cases? This question should be researched in more detail and a method (cross-validation?) developed that recommends the optimal choice of factors. Many application areas of FMs exist such as the use of FMs in feature selection. This is well worth exploring given the ever-increasing sizes of the data sets that we are confronted with. Machine learners generally concentrate on the prediction ability of FMs, leaving research into statistical inference untouched. This could be a fruitful research area for statisticians.

Appendix A

Author guidelines for SASJ and Risks



Author guidelines

All manuscripts must be submitted via the [online submission system](#) as a single PDF document. The manuscript can be composed using any typesetting software (including TeX and Microsoft Word) and first submissions are not required to be in the exact format required by the journal.

All submitted manuscripts are refereed through a double-blind peer review process. Once the manuscript has undergone peer review, the corresponding author will be contacted by the Editor with feedback on the paper.

Papers accepted for publication must be transferred to the correct TeX template (available for download [here](#)) by the authors. The editorial staff at SASJ will not take responsibility for this process, but in some cases assistance can be made available on request.

Please use British (UK-English) spelling and grammar consistently throughout your manuscript.

Official format of the South African Statistical Journal

All articles published in the South African Statistical Journal should adhere to the following set of guidelines to ensure uniformity and consistency of publications.

Abstract

The aim of the abstract is to provide a concise description of your article. It should preferably be no more than 300 words and contain a minimum of symbols and references.

Keywords

Keywords should be listed below the abstract and should appear in alphabetical order, each keyword starting with a capital letter and separated from the previous keyword by a comma.

Main body

For the main body of the article, the following guidelines should be adhered to:

- ☐ Sections are numbered consecutively using Arabic numerals.
- ☐ Section titles should be in sentence case and unnecessary capitalisation should be avoided, e.g. “A brief history of Lebesgue integration” instead of “A Brief History of Lebesgue Integration”.
- ☐ The first paragraph of each section or subsection has no indentation of the left margin.
- ☐ All subsequent paragraphs in a section or subsection are indented at the left margin.
- ☐ 1.2 spacing is used except for list items.
- ☐ Full stops are used after labels of tables, figures, theorems, remarks, lemmas, corollaries and examples (e.g. Theorem 1. or Example 1.).
- ☐ Equations that are referenced in the text must be numbered sequentially on the right-hand side of the page using Arabic numerals in brackets. Equations that are not referenced should not have equation numbers.

- Displayed equations (i.e., equations that appear on their own line of text and are centred on the page) should contain appropriate punctuation.
- Inline equations should be typeset in such a way that line spacing is not disturbed. This can be done by avoiding using constructions which are too large, such as elaborate fractions, double exponents and brackets. Inline symbolic fractions should be typeset using negative exponents, for example n^{-1} instead of $\frac{1}{n}$. Inline numerical fractions such as $\frac{1}{2}$ or $\frac{53}{1000}$ are acceptable.
- Equations are referenced by simply stating the equation number in parentheses, e.g., “... as defined in (1)”. It is preferable not to use the word “Equation” with the equation number when referencing.
- When references are cited as nouns, then they must be written either as “Abramowitz and Stegun (1970)” or as “Abramowitz and Stegun (1970, page 100).”

Definitions, remarks and examples

A definition is stated starting with the word **Definition** in bold, its number and a full stop. Definitions are numbered consecutively using Arabic numerals. The definition statement is then typeset in roman font (i.e. not italic). An example:

Definition 1. This is a definition.

Remarks and examples follow the same rule.

Theorems and proofs

A theorem is stated starting with the word **Theorem** in bold, the number of the theorem and a full stop. Theorems are numbered consecutively using Arabic numerals. The statement of the theorem is then typeset in italics. The proof starts with the word “*Proof*” in italics, then a full stop and the proof follows. The end of the proof is indicated by a solid square. An example:

Theorem 1. *Here we state the theorem.*

Proof. Here we provide a proof. ■

Lemmas, corollaries and propositions are treated in the same way.

Figures

The figure label and description are placed directly below the figure. Figures are numbered in the order that they are cited using Arabic numerals. The word “**Figure**” is in bold and the number is followed by a full stop. The description follows in the normal font and ends with a full stop. An example:

THE FIGURE COMES FIRST.

Figure 1. Here we provide a short description. The label appears below the figure.

Tables

Table names and descriptions are placed directly above the table. Tables are numbered using Arabic numerals in the order they are cited. Table labels start with the word “**Table**” and the number is followed by a full stop, all in bold. Then a description follows in normal font, terminated by a full stop. An example:

Table 1. Here we provide a short description. The table follows.

Greek letter name	Small letter	Capital letter
alpha	α	A
beta	β	B
gamma	γ	Γ
delta	δ	Δ

As in the example shown in Table 1, horizontal lines should be kept to a minimum and vertical lines in tables should be avoided completely if possible.

Acknowledgements

Acknowledgements may be included as a separate section before the references. Acknowledgements should be kept concise. No section number is used.

Appendices

Appendices are placed at the back of the article and numbered alphabetically if there is more than one, e.g. Appendix A: Descriptive title A, Appendix B: Descriptive title B, etc. No section number is used.

References

Only references cited in the text should be included. No section number is used. The format of references is illustrated by the following examples:

Book

ABRAMOWITZ, M. AND STEGUN, I. (1970). *Handbook of Mathematical Functions*. Dover Publications, New York.

Article in a journal

BOLLERSLEV, T., CHOU, R. Y., AND KRONER, K. F. (1992). ARCH modelling in Finance: a review of the theory and empirical evidence. *Journal of Econometrics*, **39**, 5–59.

GBUR, E. E. AND TRUMBO, B. E. (1995). Key words and phrases—The key to scholarly visibility and efficiency in an information explosion. *American Statistician*, **49**, 29–33.

Proceedings article

WOLFINGER, R. D. (1999). Fitting nonlinear mixed models with the new NLMIXED procedure. *In Proceedings of the 24th Annual SAS Users Group International Conference (SUGI 24)*. Miami Beach, FL, USA, pp. 278–284.

Chapter in a book

BOLLERSLEV, T., ENGLE, R. F., AND NELSON, D. B. (1994). ARCH models. *In* ENGLE, R. F. AND MCFADDEN, D. C. (Editors) *Handbook of Econometrics*. North-Holland: Amsterdam, pp. 2959–3038.

Note the use of the small caps font for the author names. Note also that the titles of books are typeset in title case (i.e. the first letter of each word is capitalised) whereas the titles of journal articles employ normal sentence case (i.e., only the first letter of the first word, the first letter following punctuation such as a semicolon are capitalised). Journal titles are written out in full and not abbreviated.

Risks

Instructions for Authors

Shortcuts

- Manuscript Submission Overview Manuscript Preparation
- Preparing Figures, Schemes and Tables
- Supplementary Materials, Data Deposit and Software Source Code Research and Publication Ethics
- Reviewer Suggestions English Corrections
- Preprints and Conference Papers Authorship
- Editorial Independence Conflict of Interests
- Editorial Procedures and Peer-Review
- Promoting Equity, Diversity and Inclusiveness Within MDPI Journals

Submission Checklist

Please:

1. Read the Aims & Scope to gain an overview and assess if your manuscript is suitable for this journal.
2. Use the Microsoft Word template or LaTeX template to prepare your manuscript.
3. Make sure that issues about publication ethics, copyright, authorship, figure formats, data and references format have been appropriately considered.
4. Ensure that all authors have approved the content of the submitted manuscript.
5. Authors are encouraged to add a biography (optional) to the submission and post it to SciProfiles.

Manuscript Submission Overview

Types of Publications

Risks has no restriction on the maximum word count or number of Figures/Tables included in the text, provided that the text is concise and comprehensive. Full experimental details must be provided so that the results can be reproduced. *Risks* requires that authors publish all experimental controls and make full datasets available where possible (see the guidelines on Supplementary Materials and references to unpublished data).

Manuscripts submitted to *Risks* should neither be published previously nor be under consideration for publication in another journal. The main article types are listed below, and a comprehensive list of article types can be found [here](#).

Submission Process

Manuscripts for *Risks* should be submitted online at susy.mdpi.com. The submitting author, who is generally the corresponding author, is responsible for the manuscript during the submission and peer-review process. The submitting author must ensure that all eligible co-authors have been included in the author list (read the criteria to qualify for authorship) and that they have all read and approved the submitted version of the manuscript. To submit your manuscript, register and log in to the submission website. Once you have registered, click [here](#) to go to the submission form for *Risks*. All co-authors can see the manuscript details in the submission system, if they register and log in using the e-mail address provided during manuscript submission.

Accepted File Formats

Authors are encouraged to use the Microsoft Word template or LaTeX template to prepare their manuscript. Using the template file will substantially shorten the time to complete copy-editing and publication of accepted manuscripts. The total amount of data for all files must not exceed 120 MB. If this is a problem, please contact the Editorial Office risks@mdpi.com. Accepted file formats are:

- **Microsoft Word:** Manuscripts prepared in Microsoft Word must be converted into a single file before submission. When preparing manuscripts in Microsoft Word, we encourage you to use the *Risks* Microsoft Word template file. Please insert your graphics (schemes, figures, etc.) in the main text after the paragraph of its first citation. **LaTeX:** Manuscripts prepared in LaTeX must be collated into one ZIP folder (including all source files and images, so that the Editorial Office can recompile the submitted PDF). When preparing manuscripts in LaTeX, we encourage you to use the *Risks* LaTeX template files. You can now also use the online application [writeLaTeX](#) to submit articles directly to *Risks*. The MDPI LaTeX template file should be selected from the [writeLaTeX](#) template gallery.

- **Supplementary files:** May be any format, but it is recommended that you use common, non-proprietary formats where possible (see below for further details).

Disclaimer: Usage of these templates is exclusively intended for submission to the journal for peer-review, and strictly limited to this purpose and it cannot be used for posting online on preprint servers or other websites.

Free Format Submission

Risks now accepts free format submission:

- We do not have strict formatting requirements, but all manuscripts must contain the required sections: Author Information, Abstract, Keywords, Introduction, Materials & Methods, Results, Conclusions, Figures and Tables with Captions, Funding Information, Author Contributions, Conflict of Interest, and other Ethics Statements. Check the Journal *Instructions for Authors* for more details.
- Your references may be in any style, if you use the consistent formatting throughout. It is essential to include author(s) name(s), journal or book title, article, or chapter title (where required), year of publication, volume and issue (where appropriate) and pagination. DOI numbers (Digital Object Identifier) are not mandatory but highly encouraged. The bibliography software package *EndNote*, *Zotero*, *Mendeley*, *Reference Manager* are recommended.
- When your manuscript reaches the revision stage, you will be requested to format the manuscript according to the journal guidelines.

Cover Letter

A cover letter must be included with each manuscript submission. It should be concise and explain why the content of the paper is significant, placing the findings in the context of existing work. It should explain why the manuscript fits the scope of the journal.

Any prior submissions of the manuscript to MDPI journals must be acknowledged. If this is the case, it is strongly recommended that the previous manuscript ID is provided in the submission system, which will ease your current submission process. The names of proposed and excluded reviewers should be provided in the submission system, not in the cover letter.

All cover letters are required to include the statements:

- We confirm that neither the manuscript nor any parts of its content are currently under consideration or published in another journal.
- All authors have approved the manuscript and agree with its submission to (journal name).

Author Biography

Authors are encouraged to add a biography (maximum 150 words) to the submission and post it to SciProfiles. This should be a single paragraph and should contain the following points:

1. Authors' full names followed by current positions.
2. Education background including institution information and year of graduation (type and level of degree received).
3. Work experience.
4. Current and previous research interests.
5. Memberships of professional societies and awards received.

Manuscript Preparation

General Considerations

- **Research manuscripts** should comprise:
 - Front matter: Title, Author list, Affiliations, Abstract, Keywords
 - Research manuscript sections: Introduction, Results, Discussion, Materials and Methods, Conclusions (optional).
 - Back matter: Supplementary Materials, Acknowledgments, Author Contributions, Conflicts of Interest, References.
- **Review manuscripts** should comprise the front matter, literature review sections and the back matter. The template file can also be used to prepare the front and back matter of your review manuscript. It is not necessary to follow the remaining structure. Structured reviews and meta-analyses

should use the same structure as research articles and ensure they conform to the PRISMA guidelines.

- **Graphical Abstract:**

A graphical abstract (GA) is an image that appears alongside the text abstract in the Table of Contents. In addition to summarizing the content, it should represent the topic of the article in an attention-grabbing way.

Moreover, it should not be the same as the Figure in the paper or just a simple superposition of several subfigures. Note that the GA must be original and unpublished artwork. Any postage stamps, currency from any country, or trademarked items should not be included in it.

The GA should be a high-quality illustration or diagram in any of the following formats: PNG, JPEG, TIFF, or SVG. Written text in a GA should be clear and easy to read, using one of the following fonts: Times, Arial, Courier, Helvetica, Ubuntu or Calibri.

The minimum required size for the GA is 560 × 1100 pixels (height × width). The size should be of high quality to reproduce well.

- **Acronyms/Abbreviations/Initialisms** should be defined the first time they appear in each of three sections: the abstract; the main text; the first figure or table. When defined for the first time, the acronym/abbreviation?
/Initialism should be added in parentheses after the written-out form. **SI Units** (International System of Units) should be used. Imperial, US customary and other units should be converted to SI units whenever possible.
- **Equations:** If you are using Word, please use either the Microsoft Equation Editor or the MathType add-on. Equations should be editable by the editorial office and not appear in a picture format.
- **Research Data and supplementary materials:** Note that publication of your manuscript implies that you must make all materials, data, and protocols associated with the publication available to readers. Disclose at the submission stage any restrictions on the availability of materials or information. Read the information about Supplementary Materials and Data Deposit for additional guidelines.
- **Preregistration:** Where authors have preregistered studies or analysis plans, links to the preregistration must be provided in the manuscript.
- **Guidelines and standards:** MDPI follow standards and guidelines for certain types of research. See https://www.mdpi.com/editorial_process for further information.

Front Matter

These sections should appear in all manuscript types.

- **Title:** The title of your manuscript should be concise, specific, and relevant. It should identify if the study reports (human or animal) trial data, or is a systematic review, meta-analysis, or replication study. Please do not include abbreviated or short forms of the title, such as a running title or head. These will be removed by our Editorial Office.
- **Author List and Affiliations:** Authors' full first and last names must be provided. The initials of any middle names can be added. The PubMed/MEDLINE standard format is used for affiliations: complete address information including city, zip code, state/province, and country. At least one author should be designated as corresponding author, and his or her email address and other details should be included at the end of the affiliation section. Please read the criteria to qualify for authorship.
- **Abstract:** The abstract should be a total of about 200 words maximum. The abstract should be a single paragraph and should follow the style of structured abstracts, but without headings: 1) Background: Place the question addressed in a broad context and highlight the purpose of the study; 2) Methods: Describe briefly the main methods or treatments applied. Include any relevant preregistration numbers, and species and strains of any animals used. 3) Results: Summarize the article's main findings; and 4) Conclusion: Indicate the main conclusions or interpretations. The abstract should be an objective representation of the article: it must not contain results which are not presented and substantiated in the main text and should not exaggerate the main conclusions.
- **Keywords:** Three to ten pertinent keywords need to be added after the abstract. We recommend that the keywords are specific to the article, yet reasonably common within the subject discipline.

Research Manuscript Sections

- **Introduction:** The introduction should briefly place the study in a broad context and highlight why it is important. It should define the purpose of the work and its significance, including specific hypotheses being tested. The current state of the research field should be reviewed carefully, and key publications cited. Please highlight controversial and diverging hypotheses when necessary. Finally, briefly mention the main aim of the work and highlight the main conclusions. Keep the introduction comprehensible to scientists working outside the topic of the paper.
- **Results:** Provide a concise and precise description of the experimental results, their interpretation as well as the experimental conclusions that can be drawn.

- **Discussion:** Authors should discuss the results and how they can be interpreted in perspective of previous studies and of the working hypotheses. The findings and their implications should be discussed in the broadest context possible, and limitations of the work highlighted. Future research directions may also be mentioned. This section may be combined with Results.
- **Materials and Methods:** They should be described with sufficient detail to allow others to replicate and build on published results. New methods and protocols should be described in detail while well-established methods can be briefly described and appropriately cited. Give the name and version of any software used and make clear whether computer code used is available. Include any pre-registration codes.
- **Conclusions:** This section is not mandatory but can be added to the manuscript if the discussion is unusually long or complex.
- **Patents:** This section is not mandatory but may be added if there are patents resulting from the work reported in this manuscript.

Back Matter

- **Supplementary Materials:** Describe any supplementary material published online alongside the manuscript (figure, tables, video, spreadsheets, etc.). Please indicate the name and title of each element as follows Figure S1: title, Table S1: title, etc.
- **Funding:** All sources of funding of the study should be disclosed. Clearly indicate grants that you have received in support of your research work and if you received funds to cover publication costs. Note that some funders will not refund article processing charges (APC) if the funder and grant number are not clearly and correctly identified in the paper. Funding information can be entered separately into the submission system by the authors during submission of their manuscript. Such funding information, if available, will be deposited to FundRef if the manuscript is finally published. Please add: "This research received no external funding" or "This research was funded by [name of funder] grant number [xxx]" and "The APC was funded by [XXX]" in this section. Check carefully that the details given are accurate and use the standard spelling of funding agency names at <https://search.crossref.org/funding>, any errors may affect your future funding.
- **Acknowledgments:** In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).
- **Author Contributions:** Each author is expected to have made substantial contributions to the conception or design of the work; or the acquisition, analysis, or interpretation of data; or the creation of new software used in the work; or have drafted the work or substantively revised it; AND has approved the submitted version (and version substantially edited by journal staff that involves the author's contribution to the study); AND agrees to be personally accountable for the author's own contributions and for ensuring that questions related to the accuracy or integrity of any part of the work, even ones in which the author was not personally involved, are appropriately investigated, resolved, and documented in the literature.
For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, X.X. and Y.Y.; Methodology, X.X.; Software, X.X.; Validation, X.X., Y.Y. and Z.Z.; Formal Analysis, X.X.; Investigation, X.X.; Resources, X.X.; Data Curation, X.X.; Writing – Original Draft Preparation, X.X.; Writing – Review & Editing, X.X.; Visualization, X.X.; Supervision, X.X.; Project Administration, X.X.; Funding Acquisition, Y.Y.", please turn to the CRediT taxonomy for the term explanation. For more background on CRediT, see here. **"Authorship must include and be limited to those who have contributed substantially to the work. Please read the section concerning the criteria to qualify for authorship carefully".**
- **Data Availability Statement:** In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Please refer to suggested Data Availability Statements in section "MDPI Research Data Policies". You might choose to exclude this statement if the study did not report any data.
- **Conflicts of Interest:** Authors must identify and declare any personal circumstances or interest that may be perceived as influencing the representation or interpretation of reported research results. If there is no conflict of interest, please state "The authors declare no conflict of interest." Any role of the funding sponsors in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript, or in the decision to publish the results must be declared in this section. If there is no role, please state "The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results". For more details, please see Conflict of Interest.
- **References:** A reference list is always arranged alphabetically. All sources are listed by the last names of the authors and listed individually at the end of the manuscript. We recommend preparing the references with a bibliography software package, such as EndNote, ReferenceManager or Zotero to avoid typing mistakes and duplicated references. We encourage citations to data, computer code and other citable research material. If available online, you may use reference style 8. below.
- Citations and References in Supplementary files are permitted if they also appear in the main text and in the reference list.

All the references mentioned in the text should be cited in the “Author-Date” format, for example (Woodward 1987), (Schuman and Scott 1987); An author- date citation in running text or at the end of a block quotation consists of the last (family) name of the author, followed by the year of publication of the work in question. In this context, author may refer not only to one or more authors or an institution but also to one or more editors, translators, or compilers. No punctuation appears between author and date. Abbreviations such as ed. or trans. are omitted.

The Reference list should include the full title as recommended by the Chicago style guide. The style file for endnote, MDPI.ens, can be found at <http://mdpi-res.com/data/chicago-mdpi-v2.ens> References should be described as follows depending on the type of work:

- Journal Articles:
 1. Author 1, and Author 2. Year. Title of the Article. *Journal Title* 6: 100–10.
- Books and Book Chapters:
 2. Author 1, and Author 2. 2008. *Book Title*, rev. ed. Publication place: Publisher, pp. 154–96.
 3. Author 1, and Author 2. 2008. Title of the chapter. In *Book Title*, 2nd ed. Edited by Editor 1 and Editor 2. Publication place: Publisher, vol. 3, pp. 54–96.
- Unpublished work, submitted work, personal communication:
 4. Author 1, and Author 2. Title of Unpublished Work. *Journal Title*, phrase indicating stage of publication.
 5. Author 1 (University, City, State, Country), and Author 2 (Institute, City, State, Country). Year. Personal communication.
- Conference Proceedings:
 6. Author 1, Author 2, and Author 3. Year. Title of Presentation. In *Title of the Collected Work* (if available). Paper presented at Name of the Conference, Location of Conference, Date of Conference. Edited by Editor 1, Editor 2 and Editor 3 (if available). Publication place: Publisher (if available); Abstract Number (optional), Pagination (optional).
- Thesis:
 7. Author 1. Year. Title of Thesis. Level of Thesis, Degree-Granting University, City, Country. Date (if available). Identification information (if available).
- Websites:
 8. Author 1, and Author 2. Year. Title of the article. *Magazine/Journal Name*. Available online: URL (accessed on Day Month Year).

Unlike published works, websites may change over time or disappear, so we encourage you create an archive of the cited website using a service such as WebCite. Archived websites should be cited using the link provided as follows:

 9. Title of Site. Available online: URL (archived on Day Month Year). See the Reference List

and Citations Guide for more detailed information.

Preparing Figures, Schemes and Tables

- File for Figures and Schemes must be provided during submission in a single zip archive and at a sufficiently high resolution (minimum 1000 pixels width/height, or a resolution of 300 dpi or higher). Common formats are accepted; however, TIFF, JPEG, EPS and PDF are preferred. *Risks* can publish multimedia files in articles or as supplementary materials. Please contact the editorial office for further information.
- All Figures, Schemes and Tables should be inserted into the main text close to their first citation and must be numbered following their number of appearance (Figure 1, Scheme I, Figure 2, Scheme II, Table 1, etc.).
- All Figures, Schemes and Tables should have a short explanatory title and caption.
- All table columns should have an explanatory heading. To facilitate the copy-editing of larger tables, smaller fonts may be used, but no less than 8 pt. in size. Authors should use the Table option of Microsoft Word to create tables.
- Authors are encouraged to prepare figures and schemes in color (RGB at 8-bit per channel). There is no additional cost for publishing full color graphics.

[Return to top]

Supplementary Materials, Data Deposit and Software Source Code

MDPI Research Data Policies

MDPI is committed to supporting open scientific exchange and enabling our authors to achieve best practices in sharing and archiving research data. We encourage all authors of articles published in MDPI journals to share their research data. Individual journal guidelines can be found at the journal ‘Instructions for Authors’ page. Data sharing policies concern the minimal dataset that supports the central findings of a published study. Generated data should be publicly available and cited in accordance with journal guidelines.

MDPI data policies are informed by TOP Guidelines and FAIR Principles.

Where ethical, legal or privacy issues are present, data should not be shared. The authors should make any limitations clear in the Data Availability Statement upon submission. Authors should ensure that data shared are in accordance with consent provided by participants on the use of confidential data.

Data Availability Statements provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study.

Below are suggested Data Availability Statements: Data available in a

- publicly accessible repository.
The data presented in this study are openly available in [repository name e.g., FigShare] at [doi], reference number [reference number].
- Data available in a publicly accessible repository that does not issue DOIs Publicly available datasets were analyzed in this study. This data can be found here: [link/accession number]
- Data available on request due to restrictions e.g. privacy or ethical The data presented in this study are available on request from the corresponding author. The data are not publicly available due to [insert reason here]
- 3rd Party Data
Restrictions apply to the availability of these data. Data was obtained from [third party] and are available [from the authors / at URL] with the permission of [third party].
- Data sharing not applicable.
No new data were created or analyzed in this study. Data sharing is not applicable to this article.
- Data is contained within the article or supplementary material The data presented in this study are available in [insert article or supplementary material here]

Data citation:

- [dataset] Authors. Year. Dataset title; Data repository or archive; Version (if any); Persistent identifier (e.g., DOI).

Computer Code and Software

For work where novel computer code was developed, authors should release the code either by depositing in a recognized, public repository or uploading as supplementary information to the publication. The name and version of all software used should be clearly indicated.

Supplementary Material

Additional data and files can be uploaded as "Supplementary Files" during the manuscript submission process. The supplementary files will also be available to the referees as part of the peer-review process. Any file format is acceptable; however we recommend that common, non-proprietary formats are used where possible. For more information on supplementary materials, please refer to https://www.mdpi.com/authors/layout#_bookmark83.

Unpublished Data

Restrictions on data availability should be noted during submission and in the manuscript. "Data not shown" should be avoided: authors are encouraged to publish all observations related to the submitted manuscript as Supplementary Material. "Unpublished data" intended for publication in a manuscript that is either planned, "in preparation" or "submitted" but not yet accepted, should be cited in the text and a reference should be added in the References section. "Personal Communication" should also be cited in the text and reference added in the References section. (See also the MDPI reference list and citations style guide).

Remote Hosting and Large Data Sets

Data may be deposited with specialized service providers or institutional/subject repositories, preferably those that use the DataCite mechanism. Large data sets and files greater than 60 MB must be deposited in this way. For a list of other repositories specialized in scientific and experimental data, please consult databib.org or re3data.org. The data repository name, link to the data set (URL) and accession number, doi or handle number of the data set must be provided in the paper. The journal Data also accepts submissions of data set papers.

References in Supplementary Files

Citations and References in Supplementary files are permitted provided that they also appear in the reference list of the main text.

Research and Publication Ethics

Research Ethics

Research Involving Human Subjects

When reporting on research that involves human subjects, human material, human tissues, or human data, authors must declare that the investigations were carried out following the rules of the Declaration of Helsinki of 1975 (<https://www.wma.net/what-we-do/medical-ethics/declaration-of-helsinki/>), revised in 2013. According to point 23 of this declaration, an approval from the local institutional review board (IRB) or other appropriate ethics committee must be obtained before undertaking the research to confirm the study meets national and international guidelines. As a minimum, a statement including the project identification code, date of approval, and name of the ethics committee or institutional review board must be stated in Section 'Institutional Review Board Statement' of the article.

Example of an ethical statement: "All subjects gave their informed consent for inclusion before they participated in the study. The study was conducted in accordance with the Declaration of Helsinki, and the protocol was approved by the Ethics Committee of XXX (Project identification code)."

For non-interventional studies (e.g. surveys, questionnaires, social media research), all participants must be fully informed if the anonymity is assured, why the research is being conducted, how their data will be used and if there are any risks associated. As with all research involving humans, ethical approval from an appropriate ethics committee must be obtained prior to conducting the study. If ethical approval is not required, authors must either provide an exemption from the ethics committee or are encouraged to cite the local or national legislation that indicates ethics approval is not required for this type of study. Where a study has been granted exemption, the name of the ethics committee which provided this should be stated in Section 'Institutional Review Board Statement' with a full explanation regarding why ethical approval was not required.

A written informed consent for publication must be obtained from participating patients. Data relating to individual participants must be described in detail, but private information identifying participants need not be included unless the identifiable materials are of relevance to the research (for example, photographs of participants' face that show a particular symptom). Patients' initials or other personal identifiers must not appear in any images. For manuscripts that include any case details, personal information, and/or images of patients, authors must obtain signed informed consent for publication from patients (or their relatives/guardians) before submitting to an MDPI journal. Patient details must be anonymized as far as possible, e.g., do not mention specific age, ethnicity, or occupation where they are not relevant to the conclusions. A template permission form is available to download. A blank version of the form used to obtain permission (without the patient names or signature) must be uploaded with your submission. Editors reserve the right to reject any submission that does not meet these requirements.

You may refer to our sample form and provide an appropriate form after consulting with your affiliated institution. For the purposes of publishing in MDPI journals, a consent, permission, or release form should include unlimited permission for publication in all formats (including print, electronic, and online), in sublicensed and reprinted versions (including translations and derived works), and in other works and products under open access license. To respect patients' and any other individual's privacy, please do not send signed forms. The journal reserves the right to ask authors to provide signed forms if necessary.

If the study reports research involving vulnerable groups, an additional check may be performed. The submitted manuscript will be scrutinized by the editorial office and upon request, documentary evidence (blank consent forms and any related discussion documents from the ethics board) must be supplied. Additionally, when studies describe groups by race, ethnicity, gender, disability, disease, etc., explanation regarding why such categorization was needed must be clearly stated in the article.

Ethical Guidelines for the Use of Animals in Research

The editors will require that the benefits potentially derived from any research causing harm to animals are significant in relation to any cost endured by animals, and that procedures followed are unlikely to cause offense to most readers. Authors should particularly ensure that their research complies with the commonly accepted '3Rs [1]':

- Replacement of animals by alternatives wherever possible, Reduction in number of animals used, and
- Refinement of experimental conditions and procedures to minimize the harm to animals.

Authors must include details on housing, husbandry, and pain management in their manuscript.

For further guidance authors should refer to the Code of Practice for the Housing and Care of Animals Used in Scientific Procedures [2], American Association for Laboratory Animal Science [3] or European Animal Research Association [4].

If national legislation requires it, studies involving vertebrates or higher invertebrates must only be carried out after obtaining approval from the appropriate ethics committee. As a minimum, the project identification code, date of approval and name of the ethics committee or institutional review board should be stated in Section 'Institutional Review Board Statement'. Research procedures must be carried out in accordance with national and institutional regulations. Statements on animal welfare should confirm that the study complied with all relevant legislation. Clinical studies involving animals and interventions outside of routine care require ethics committee oversight as per the American Veterinary Medical Association. If the study involved client-owned animals, informed client consent must be obtained and certified in the manuscript report of the research. Owners must be fully informed if there are any risks associated with the procedures and that the research will be published. If available, a high standard of veterinary care must be provided. Authors are responsible for correctness of the statements provided in the manuscript.

If ethical approval is not required by national laws, authors must provide an exemption from the ethics committee if one is available. Where a study has been granted exemption, the name of the ethics committee that provided this should be stated in Section 'Institutional Review Board Statement' with a full explanation on why the ethical approval was not required.

If no animal ethics committee is available to review applications, authors should be aware that the ethics of their research will be evaluated by reviewers and editors. Authors should provide a statement justifying the work from an ethical perspective, using the same utilitarian framework that is used by ethics committees. Authors may be asked to provide this even if they have received ethical approval.

MDPI endorses the ARRIVE guidelines (arriveguidelines.org/) for reporting experiments using live animals. Authors and reviewers must use the ARRIVE guidelines as a checklist, which can be found at <https://arriveguidelines.org/sites/arrive/files/documents/ARRIVE%20Compliance%20Questionnaire.pdf>. Editors reserve the right to ask for the checklist and to reject submissions that do not adhere to these guidelines, to reject submissions based on ethical or animal welfare concerns or if the procedure described does not appear to be justified by the value of the work presented.

1. NSW Department of Primary Industries and Animal Research Review Panel. Three Rs. Available online: <https://www.animaletics.org.au/three-rs>
2. Home Office. Animals (Scientific Procedures) Act 1986. Code of Practice for the Housing and Care of Animals Bred, Supplied or Used for Scientific Purposes. Available online: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/388535/CoPanimalsWeb.pdf
3. American Association for Laboratory Animal Science. The Scientific Basis for Regulation of Animal Care and Use. Available online: <https://www.aalas.org/about-aalas/position-papers/scientific-basis-for-regulation-of-animal-care-and-use>
4. European Animal Research Association. EU regulations on animal research. Available online: <https://www.eara.eu/animal-research-law>

Research Involving Cell Lines

Methods sections for submissions reporting on research with cell lines should state the origin of any cell lines. For established cell lines the provenance should be stated and references must also be given to either a published paper or to a commercial source. If previously unpublished *de novo* cell lines were used, including those gifted from another laboratory, details of institutional review board or ethics committee approval must be given, and confirmation of written informed consent must be provided if the line is of human origin.

An example of Ethical Statements:

The HCT116 cell line was obtained from XXXX. The MLH1⁺ cell line was provided by XXXXX, Ltd. The DLD-1 cell line was obtained from Dr. XXXX. The DR-GFP and SA-GFP reporter plasmids were obtained from Dr. XXX and the Rad51K133A expression vector was obtained from Dr. XXXX.

Research Involving Plants

Experimental research on plants (either cultivated or wild) including collection of plant material, must comply with institutional, national, or international guidelines. We recommend that authors comply with the Convention on Biological Diversity and the Convention on the Trade in Endangered Species of Wild Fauna and Flora.

For each submitted manuscript supporting genetic information and origin must be provided. For research manuscripts involving rare and non-model plants (other than, e.g., *Arabidopsis thaliana*, *Nicotiana benthamiana*, *Oryza sativa*, or many other typical model plants), voucher specimens must be deposited in an accessible herbarium or museum. Vouchers may be requested for review by future investigators to verify the identity of the material used in the study (especially if taxonomic rearrangements occur in the future). They should include details of the populations sampled on the site of collection (GPS coordinates), date of collection, and document the part(s) used in the study where appropriate. For rare, threatened or endangered species this can be waived but it is necessary for the author to describe this in the cover letter.

Editors reserve the rights to reject any submission that does not meet these requirements.

An example of Ethical Statements:

Torenia fournieri plants were used in this study. White-flowered Crown White (CrW) and violet-flowered Crown Violet (CrV) cultivars selected from 'Crown Mix' (XXX Company, City, Country) were kindly provided by Dr. XXX (XXX Institute, City, Country).

Arabidopsis mutant lines (SALKxxxx, SAILxxxx,...) were kindly provided by Dr. XXX, institute, city, country).

Clinical Trials Registration

Registration

MDPI follows the International Committee of Medical Journal Editors (ICMJE) guidelines which require and recommend registration of clinical trials in a public trials' registry at or before the time of first patient enrollment as a condition of consideration for publication.

Purely observational studies do not require registration. A clinical trial not only refers to studies that take place in a hospital or involve pharmaceuticals, but also refer to all studies which involve participant randomization and group classification in the context of the intervention under assessment.

Authors are strongly encouraged to pre-register clinical trials with international clinical trials register and cite a reference to the registration in the Methods section. Suitable databases include clinicaltrials.gov, the EU Clinical Trials Register and those listed by the World Health Organisation International Clinical Trials Registry Platform.

Approval to conduct a study from an independent local, regional, or national review body is not equivalent to prospective clinical trial registration. MDPI reserves the right to decline any paper without trial registration for further peer- review. However, if the study protocol has been published before the enrolment, the registration can be waived with correct citation of the published protocol.

CONSORT Statement

MDPI requires a completed CONSORT 2010 checklist and flow diagram as a condition of submission when reporting the results of a randomized trial.

Templates for these can be found here or on the CONSORT website (<http://www.consort-statement.org>) which also describes several CONSORT checklist extensions for different designs and types of data beyond two group parallel trials. At minimum, your article should report the content addressed by each item of the checklist.

[Return to top] [_](#)

Sex and Gender in Research

We encourage our authors to follow the 'Sex and Gender Equity in Research – SAGER – guidelines' and to include sex and gender considerations where relevant. Authors should use the terms sex (biological attribute) and gender (shaped by social and cultural circumstances) carefully to avoid confusing both terms. Article titles and/or abstracts should indicate clearly what sex(es) the study applies to. Authors should also describe in the background, whether sex and/or gender differences may be expected; report how sex and/or gender were accounted for in the design of the study; provide disaggregated data by sex and/or gender, where appropriate; and discuss respective results. If a sex and/or gender analysis was not conducted, the rationale should be given in the Discussion. We suggest that our authors consult the full guidelines before submission.

[Return to top] [_](#)

Borders and Territories

Potential disputes over borders and territories may have particular relevance for authors in describing their research or in an author or editor correspondence address and should be respected. Content decisions are an editorial matter and where there is a potential or perceived dispute or complaint, the editorial team will attempt to find a resolution that satisfies parties involved.

MDPI stays neutral about jurisdictional claims in published maps and institutional affiliations.

Publication Ethics Statement

Risks is a member of the Committee on Publication Ethics (COPE). We fully adhere to its Code of Conduct and to its Best Practice Guidelines.

The editors of this journal enforce a rigorous peer-review process together with strict ethical policies and standards to ensure to add high quality scientific works to the field of scholarly publication. Unfortunately, cases of plagiarism, data falsification, image manipulation, inappropriate authorship credit, and the like, do arise. The editors of *Risks* take such publishing ethics issues very seriously and are trained to proceed in such cases with a zero-tolerance policy.

Authors wishing to publish their papers in *Risks* must abide to the following:

- Any facts that might be perceived as a possible conflict of interest of the author(s) must be disclosed in the paper prior to submission.
- Authors should accurately present their research findings and include an objective discussion of the significance of their findings.
- Data and methods used in the research need to be presented in sufficient detail in the paper, so that other researchers can replicate the work.
- Raw data should preferably be publicly deposited by the authors before submission of their manuscript. Authors need to at least have the raw data readily available for presentation to the referees and the editors of the journal, if requested. Authors need to ensure appropriate measures are taken so that raw data is retained in full for a reasonable time after publication.
- Simultaneous submission of manuscripts to more than one journal is not tolerated.
- The journal accepts exact translations of previously published work. All submissions of translations must conform with our policies on translations.
- If errors and inaccuracies are found by the authors after publication of their paper, they need to be promptly communicated to the editors of this journal so that appropriate actions can be taken. Please refer to our policy regarding Updating Published Papers.
- Your manuscript should not contain any information that has already been published. If you include already published figures or images, please obtain the necessary permission from the copyright holder to publish under the CC-BY license. For further information, see the Rights and Permissions page.
- Plagiarism, data fabrication and image manipulation are not tolerated.
 - **Plagiarism is not acceptable** in *Risks* submissions.

Plagiarism includes copying text, ideas, images, or data from another source, even from your own publications, without giving any credit to the original source.

Reuse of text that is copied from another source must be between quotes and the original source must be cited. If a study's design or the manuscript's structure or language has been inspired by previous works, these works must be explicitly cited.

All MDPI submissions are checked for plagiarism using the industry standard software iThenticate. If plagiarism is detected during the peer review process, the manuscript may be rejected. If plagiarism is detected after publication, an investigation will take place and action taken in accordance with our policies.

- **Image files must not be manipulated or adjusted in any way** that could lead to misinterpretation of the information provided by the original image.

Irregular manipulation includes: 1) introduction, enhancement, moving, or removing features from the original image; 2) grouping of images that should obviously be presented separately (e.g., from different parts of the same gel, or from different gels); or 3) modifying the contrast, brightness, or color balance to obscure, eliminate or enhance some information.

If irregular image manipulation is identified and confirmed during the peer review process, we may reject the manuscript. If irregular image manipulation is identified and confirmed after publication, we may correct or retract the paper.

Our in-house editors will investigate any allegations of publication misconduct and may contact the authors' institutions or funders if necessary. If evidence of misconduct is found, appropriate action will be taken to correct or retract the publication. Authors are expected to comply with the best ethical publication practices when publishing with MDPI.

Citation Policy

Authors should ensure that where material is taken from other sources (including their own published writing) the source is clearly cited and that where appropriate permission is obtained.

Authors should not engage in excessive self-citation of their own work.

Authors should not copy references from other publications if they have not read the cited work.

Authors should not preferentially cite their own or their friends', peers', or institution's publications.

Authors should not cite advertisements or advertorial material.

In accordance with COPE guidelines, we expect that “original wording taken directly from publications by other researchers should appear in quotation marks with the appropriate citations.” This condition also applies to an author’s own work. COPE have produced a discussion document on citation manipulation with recommendations for best practice.

Reviewer Suggestions

During the submission process, please suggest three potential reviewers with the appropriate expertise to review the manuscript. The editors will not necessarily approach these referees. Please provide detailed contact information (address, homepage, phone, e-mail address). The proposed referees should neither be current collaborators of the co-authors nor have published with any of the co-authors of the manuscript within the last five years. Proposed reviewers should be from different institutions to the authors. You may identify appropriate Editorial Board members of the journal as potential reviewers. You may suggest reviewers from among the authors that you frequently cite in your paper.

English Corrections

To facilitate proper peer-reviewing of your manuscript, it is essential that it is submitted in grammatically correct English. Advice on some specific language points can be found [here](#).

MDPI provides minor English editing by native English speakers for all accepted papers, included in the APC. The APC does not cover extensive English editing. Your paper could be returned to you at the English editing stage of the publication process if extensive editing is required. You may choose to use a paid language-editing service, such as MDPI’s Author Services, before submitting your paper for publication. If you use an alternative service that provides a confirmation certificate, please send a copy to the Editorial Office. Authors from economically developing countries or nations should consider registration with AuthorAid, a global research community that provides networking, mentoring, resources, and training for researchers.

Preprints and Conference Papers

Risks accepts submissions that have previously been made available as preprints if they have not undergone peer review. A preprint is a draft version of a paper made available online before submission to a journal.

MDPI operates *Preprints*, a preprint server to which submitted papers can be uploaded directly after completing journal submission. Note that *Preprints* operates independently of the journal and posting a preprint does not affect the peer review process. Check the *Preprints* instructions for authors for further information.

Expanded and high-quality conference papers can be considered as articles if they fulfill the following requirements: (1) the paper should be expanded to the size of a research article; (2) the conference paper should be cited and noted on the first page of the paper; (3) if the authors do not hold the copyright of the published conference paper, authors should seek the appropriate permission from the copyright holder; (4) authors are asked to disclose that it is conference paper in their cover letter and include a statement on what has been changed compared to the original conference paper. *Risks* does not publish pilot studies or studies with inadequate statistical power.

Unpublished conference papers that do not meet the above conditions are recommended to be submitted to the Proceedings Series journals.

Authorship

MDPI follows the International Committee of Medical Journal Editors (ICMJE) guidelines which state that, to qualify for authorship of a manuscript, the following criteria should be observed:

- Substantial contributions to the conception or design of the work; or the acquisition, analysis, or interpretation of data for the work; AND
- Drafting the work or revising it critically for important intellectual content; AND
- Final approval of the version to be published; AND
- Agreement to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Those who contributed to the work but do not qualify for authorship should be listed in the acknowledgments. More detailed guidance on authorship is given by the International Council of Medical Journal Editors (ICMJE).

Any change to the author list should be approved by all authors including any who have been removed from the list. The corresponding author should act as a point of contact between the editor and the other authors and should keep co-authors informed and involve them in major decisions about the publication. We reserve the right to request confirmation that all authors meet the authorship conditions.

For more details about authorship please check MDPI ethics website.

Reviewers Recommendation

Authors can recommend potential reviewers. Journal editors will check to make sure there are no conflicts of interest before contacting those reviewers and will not consider those with competing interests. Reviewers are asked to declare any conflicts of interest. Authors can also enter the names of potential peer reviewers they wish to exclude from consideration in the peer review of their manuscript, during the initial submission progress. The editorial team will respect these requests so long as this does not interfere with the objective and thorough assessment of the submission.

Editorial Independence

Lack of Interference with Editorial Decisions

Editorial independence is of utmost importance and MDPI does not interfere with editorial decisions. All articles published by MDPI are peer reviewed and assessed by our independent editorial boards, and MDPI staff are not involved in decisions to accept manuscripts. When making an editorial decision, we expect the academic editor to make their decision based only upon:

- The suitability of selected reviewers.
- Adequacy of reviewer comments and author response; Overall scientific quality of the paper.

In all of our journals, in every aspect of operation, MDPI policies are informed by the mission to make science and research findings open and accessible as widely and rapidly as possible.

Editors and Editorial Staff as Authors

Editorial staff or editors shall not be involved in processing their own academic work. Submissions authored by editorial staff/editors will be assigned to at least two independent outside reviewers. Decisions will be made by other Editorial Board Members who do not have a conflict of interest with the author. Journal staff are not involved in the processing of their own work submitted to any MDPI journals.

Conflict of Interests

According to The International Committee of Medical Journal Editors, "Authors should avoid entering into agreements with study sponsors, both for-profit and non-profit, that interfere with authors' access to all of the study's data or that interfere with their ability to analyze and interpret the data and to prepare and publish manuscripts independently when and where they choose."

All authors must disclose all relationships or interests that could inappropriately influence or bias their work. Examples of potential conflicts of interest include but are not limited to financial interests (such as membership, employment, consultancies, stocks/shares ownership, honoraria, grants or other funding, paid expert testimonies and patent-licensing arrangements) and non-financial interests (such as personal or professional relationships, affiliations, personal beliefs).

Authors can disclose potential conflicts of interest via the online submission system during the submission process. Declarations regarding conflicts of interest can also be collected via the MDPI disclosure form. The corresponding author must include a summary statement in the manuscript in a separate section "Conflicts of Interest" placed just before the reference list. The statement should reflect all the collected potential conflict of interest disclosures in the form.

See below for examples of disclosures:

Conflicts of Interest: Author A has received research grants from Company A. Author B has received a speaker honorarium from Company X and owns stocks in Company Y. Author C has been involved as a consultant and expert witness in Company Z. Author D is the inventor of patent X.

If no conflicts exist, the authors should state:

Conflicts of Interest: The authors declare no conflicts of interest. [Return to top]

Editorial Procedures and Peer-Review

Initial Checks

All submitted manuscripts received by the Editorial Office will be checked by a professional in-house *Managing Editor* to determine whether they are properly prepared and whether they follow the ethical policies of the journal. Manuscripts that do not fit the journal's ethics policy or do not meet the standards of the journal will be rejected before peer-review. Manuscripts that are not properly prepared will be returned to the authors for revision and resubmission. After these checks, the *Managing Editor* will consult the journals' *Editor-in-Chief* or *Associate Editors* to determine whether the manuscript fits the scope of the journal and whether it is scientifically sound. No judgment on the potential impact of the work will be made at this stage. Reject decisions at this stage will be verified by the *Editor-in-Chief*.

Peer-Review

Once a manuscript passes the initial checks, it will be assigned to at least two independent experts for peer-review. A single-blind review is applied, where authors' identities are known to reviewers. Peer review comments are confidential and will only be disclosed with the express agreement of the reviewer.

In the case of regular submissions, in-house assistant editors will invite experts, including recommendations by an academic editor. These experts may also include *Editorial Board Members* and Guest Editors of the journal. Potential reviewers suggested by the authors may also be considered. Reviewers should not have published with any of the co-authors during the past three years and should not currently work or collaborate with any of the institutions of the co-authors of the submitted manuscript.

Optional Open Peer-Review

The journal operates optional open peer-review: *Authors are given the option for all review reports and editorial decisions to be published alongside their manuscript. In addition, reviewers can sign their review, i.e., identify themselves in the published review reports.* Authors can alter their choice for open review at any time before publication, but once the paper has been published changes will only be made at the discretion of the *Publisher* and *Editor-in-Chief*. We encourage authors to take advantage of this opportunity as proof of the rigorous process employed in publishing their research. To guarantee impartial refereeing, the names of referees will be revealed only if the referees agree to do so, and after a paper has been accepted for publication.

Editorial Decision and Revision

All the articles, reviews and communications published in MDPI journals go through the peer-review process and receive at least two reviews. The in-house editor will communicate the decision of the academic editor, which will be one of the following:

- *Accept after Minor Revisions:*
The paper is in principle accepted after revision based on the reviewer's comments. Authors are given five days for minor revisions.
- *Reconsider after Major Revisions:*
The acceptance of the manuscript would depend on the revisions. The author needs to provide a point-by-point response or provide a rebuttal if some of the reviewer's comments cannot be revised. A maximum of two rounds of major revision per manuscript is normally provided. Authors will be asked to resubmit the revised paper within a suitable time frame, and the revised version will be returned to the reviewer for further comments. If the required revision time is estimated to be longer than 2 months, we will recommend that authors withdraw their manuscript before resubmitting to avoid unnecessary time pressure and to ensure that all manuscripts are sufficiently revised.
- *Reject and Encourage Resubmission:*
If additional experiments are needed to support the conclusions, the manuscript will be rejected, and the authors will be encouraged to re-submit the paper once further experiments have been conducted.
- *Reject:*
The article has serious flaws, and/or makes no original significant contribution. No offer of resubmission to the journal is provided.

All reviewer comments should be responded to in a point-by-point fashion. Where the authors disagree with a reviewer, they must provide a clear response.

Author Appeals

Authors may appeal a rejection by sending an e-mail to the Editorial Office of the journal. The appeal must provide a detailed justification, including point-by-point responses to the reviewers' and/or Editor's comments using an appeal form. Appeals can only be submitted following a "reject and decline resubmission" decision and should be submitted within three months from the decision date. Failure to meet these criteria will result in the appeal not being considered further. The *Managing Editor* will forward the manuscript and related information (including the identities of the referees) to a designated *Editorial Board Member*. The Academic Editor being consulted will be asked to provide an advisory recommendation on the manuscript and may recommend acceptance, further peer-review, or uphold the original rejection decision. This decision will then be validated by the *Editor-in-Chief*. A reject decision at this stage is final and cannot be reversed.

Production and Publication

Once accepted, the manuscript will undergo professional copy-editing, English editing, proofreading by the authors, final corrections, pagination, and publication on the www.mdpi.com website.

Promoting Equity, Diversity and Inclusiveness Within MDPI Journals

Our Managing Editors encourage the Editors-in-Chief and Associate Editors to appoint diverse expert Editorial Boards. This is also reflective in our multi-national and inclusive workplace. We are proud to create equal opportunities without regard to gender, ethnicity, sexual orientation, age, religion, or socio-economic status. There is no place for discrimination in our workplace and editors of MDPI journals are to uphold these principles in high regard.