Explanation Mining: Post Hoc Interpretability of Latent Factor Models for Recommendation Systems

Georgina Peake*
Channel 4 Television
London, United Kingdom
gpeake@channel4.co.uk

Jun Wang
Computer Science, University College London
London, United Kingdom
junwang@cs.ucl.ac.uk

ABSTRACT

The widescale use of machine learning algorithms to drive decision-making has highlighted the critical importance of ensuring the interpretability of such models in order to engender trust in their output. The state-of-the-art recommendation systems use *black-box* latent factor models that provide no explanation of *why* a recommendation has been made, as they abstract their decision processes to a high-dimensional latent space which is beyond the direct comprehension of humans. We propose a novel approach for extracting explanations from latent factor recommendation systems by training association rules on the output of a matrix factorisation black-box model. By taking advantage of the interpretable structure of association rules, we demonstrate that predictive accuracy of the recommendation model can be maintained whilst yielding explanations with high fidelity to the black-box model on a unique industry dataset. Our approach mitigates the accuracy-interpretability trade-off whilst avoiding the need to sacrifice flexibility or use external data sources. We also contribute to the ill-defined problem of evaluating interpretability.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computing methodologies → Learning latent representations**; *Rule learning*; Learning from implicit feedback;

KEYWORDS

interpretability; recommendation systems; association rules; latent factor models; explanations; black-box; white-box

ACM Reference Format:
Georgina Peake and Jun Wang. 2018. Explanation Mining: Post Hoc Interpretability of Latent Factor Models for Recommendation Systems. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3220072

1 INTRODUCTION

One of the fundamental trade-offs in machine learning is that of accuracy and interpretability. The models that have been demonstrated to produce the most accurate results for non-linear problems are inherently complex and non-transparent. For example, the state-of-the-art deep learning techniques are considered *black-box models* as their complex architecture of hidden layers obfuscate their internal decision processes. On the other end of the spectrum, decision rules are considered an *interpretable model* as they are easy to comprehend and explain because they provide an explicit textual representation of their decisions, alongside their prediction. However, their simplicity comes at the cost of inflexibility and potentially high bias as their decisions are based on simple single variable splits (e.g. IF has fur THEN mammal ELSE reptile), which are often not an accurate assumption for the true relationship between a set of variables.

To date, the key measure of success and progress in machine learning research has been achieving the highest possible predictive performance. Today, as machine learning systems are gaining widespread adoption across industry as a means to drive decision-making, important questions are being raised around whether we can trust models to behave logically, non-discriminatingly and handle spurious data. In order to develop trust in a model, users need to understand *why* a model made a prediction, beyond solely its predictive accuracy.

As such, establishing approaches that provide an interpretable representation of the decision processes of black-box models, whilst maintaining strong predictive performance, is a highly challenging yet critical research agenda for the machine learning community, in order to ensure we can fairly and responsibly derive the significant benefits available from machine learning.

Post hoc interpretablity describes a set of approaches which seek to interpret a black-box model after it has been trained, by extracting explanations from the output of the model. As no changes are made to the black-box model itself, predictive performance can be maintained, but with the added benefit of improved interpretability, therefore reducing the accuracy-interpretability trade-off.

This paper explores these themes by applying a novel post hoc interpretability approach to a latent factor model for recommendation systems. We prove that approximating a black-box model with an interpretable model can maintain the high predictive accuracy of the recommendation model whilst improving interpretability by extracting explanations that can be used to understand model behaviour.

2 RELATED WORK

There is currently no silver bullet solution for the interpretability problem; different approaches exist for different problem domains

[*] Research was completed whilst part-time studying at UCL.

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3220072

(e.g. classification [7] vs. ranking [18]), model families (e.g. support vector machines [10] vs. neural networks [8]), applications (e.g. knowledge base extraction [33] vs. recommendation systems[1]), and motivations (e.g. model debugging [22, 25] vs. user acceptance [11, 19]) as well as constraints such as legal requirements [14].

We limit our review to approaches to extract explanations from latent factor models for recommendation systems. The state of the art recommendation systems using latent factor models such as matrix factorisation and deep neural networks are *black-box models*; their internal decision processes cannot be directly interpreted by humans as by finding lower dimensional representations of the user and item data, they abstract away from the interactions between users and items. The two predominant approaches in the recommendation literature are constraining the matrix factorisation objective function and using external data sources.

2.0.1 Constrained Matrix Factorisation. A common approach to explain matrix factorisation recommendations is to update the matrix factorisation objective function in a way that motivates the model to find explainable items, therefore combining recommendations and explanations into one model. This involves putting some constraint on the matrix factorisation model, such that it is no longer a black-box. For example, [6, 20] use Non-negative Matrix Factorisation (NNMF) by applying the constraint that the latent factors must all be positive or zero. This aids interpretability as a positive latent factor vector can be described by the sum of its positive parts, each of which can be individually interpretable. Abdollahi and Nasraoui [1] create an Explainable Matrix Factorisation (EMF) model by adding a term to the matrix factorisation objective function which captures an ‘explainability’ score of an item for a user based on the prevalence of the item in a user’s neighbourhood. Abdollahi and Nasraoui [3] extends [1] using a probabilistic formulation of EMF. Abdollahi and Nasraoui [2] also explain Restricted Boltzmann Machines (RBMs) for collaborative filtering based on [32] by adding an additional visible layer with a node for each item with a value equal to the explainability score of an item to the active user. This creates an RBM where the the joint distribution of the visible and hidden units is conditional on the explainability scores, therefore prioritising the recommendation of items that are explainable to a user as with [1]. Heckel et al. [18] use a generative model to identify overlapping groups of users and items with similar patterns which they use to constrain the latent factors to explicitly model user and item participation. The approach in [18] of identifying communities of users has similar motivations to our approach of approximating the matrix factorisation model with rules from a user’s neighbourhood ("local rules") in order to infer latent patterns.

The constrained matrix factorisation approaches have the strong advantage of not requiring a separate data source to generate explanations. They also avoid the need to uncouple the explanations and recommendations into separate models as is required with post hoc interpretability. In some instances, the constrained models outperform their unconstrained version [2, 3] in contradiction to the accuracy-interpretability trade-off. However, as these approaches impose constraints on the black-box model in some way, the model loses flexibility which risks sacrificing some accuracy if the most explainable recommendations are not highly correlated with the most relevant explanations. This suggests a trade-off between accuracy and interpretability still exists. In addition, as they require direct changes to the objective function, these approaches are model-specific and therefore may suffer from switching costs which could occur when changes are required to models in a production system [29].

Hu et al. [21] successfully extract explanations from a matrix factorisation model without constraining the matrix factorisation objective function by providing a link to the item-oriented neighbourhood approach whereby recommendations can be explained by the items which have the highest similarity to the target item and prevalence in a user’s viewing history. This is a powerful intuition; however, as with the constrained matrix factorisation approaches, it relies on a specific formulation of the objective function and therefore lacks the model flexibility described in [28].

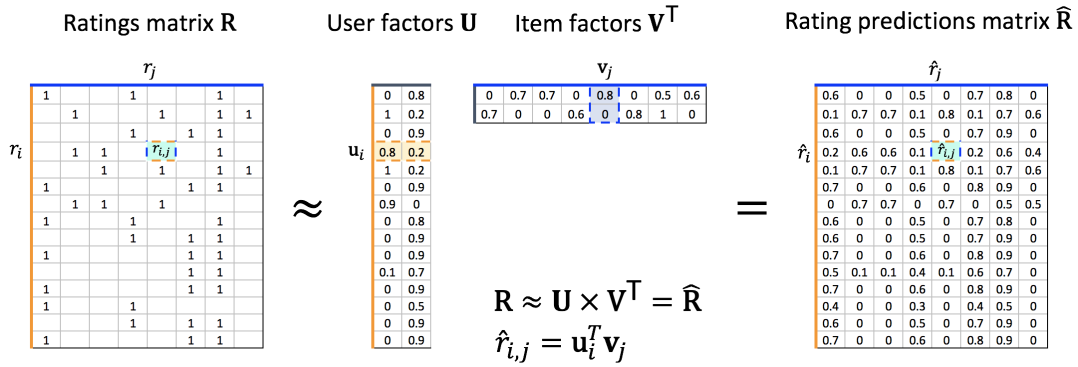
2.0.2 External data sources. When an external textual data source is available (e.g. product reviews [36] or news articles [15]), topic modelling can be used to discover the themes present in the text [9, 15, 31] which can then be combined with the rating predictions to provide an interpretation of the latent factors.

Using an additional data source can provide fruitful insight, however it suffers from two significant drawbacks. Most crucially, additional data sources can be very expensive or otherwise infeasible to obtain. Secondly, even if data can be obtained, if the data sources used for explanations are not correlated with the user-item rating data used to generate recommendations then the explanations will not accurately reflect the reasons for the recommendations, or will not be generated.

2.0.3 Post hoc interpretability. To our best knowledge, there has not previously been an application of the post hoc interpretability approach to latent factor models for recommendation systems. The closest approach to our own is by Sanchez et al [33] who use first-order logic rules and Bayesian networks as proxies for a more complex matrix factorisation model for knowledge base completion as in [30]. In essence, the aim is to maintain fidelity by finding an interpretable model which closely approximates the more complex model, whilst ensuring interpretability by providing a descriptive representation which is interpretable to humans. The authors find that an accuracy-interpretability trade-off persists as the association rules have lower fidelity to the matrix factorisation model but high interpretability, whereas Bayesian networks were a stronger proxy model but less interpretable. This model-agnostic approach does not require an additional data source and avoids the potential biases introduced by incorporating explainability into the objective function.

Ribeiro et al. [29] suggest recommendation systems as a potential application for learning interpretable models on black-box models. To the best of our knowledge, applying this approach to the ranking problem has not yet been attempted in the literature. This is the unique application where this paper seeks to contribute: applying the model-agnostic approach to recommendation systems using matrix factorisation.

Sanchez et al. [33] highlight several open research questions including what representations are good proxies for complex matrix factorisation models; how is the choice of proxy impacted by the problem domain, the expertise of its users and the motivation for explanations; and how should the proxies be evaluated. We investigate these questions as we expand upon the work done in [33] by



	1			1			1	
		1			1		1	1
				1		1	1	
r_i		1	1		$r_{i,j}$		1	
			1		1		1	1
	1					1	1	
		1	1		1			
	1			1			1	
				1		1	1	
	1					1	1	
						1	1	
	1					1	1	
	1			1				
				1		1	1	
	1					1	1	

	0	0.8
	1	0.2
	0	0.9
$\mathbf{u}_i$	0.8	0.2
	1	0.2
	0	0.9
	0.9	0
	0	0.8
	0	0.9
	0	0.9
	0.1	0.7
	0	0.9
	0	0.5
	0	0.9
	0	0.9

0	0.7	0.7	0	0.8	0	0.5	0.6
0.7	0	0	0.6	0	0.8	1	0

	0.6	0	0	0.5	0	0.7	0.8	0
	0.1	0.7	0.7	0.1	0.8	0.1	0.7	0.6
	0.6	0	0	0.5	0	0.7	0.9	0
$\hat{r}_i$	0.2	0.6	0.6	0.1	$\hat{r}_{i,j}$	0.2	0.6	0.4
	0.1	0.7	0.7	0.1	0.8	0.1	0.7	0.6
	0.7	0	0	0.6	0	0.8	0.9	0
	0	0.7	0.7	0	0.7	0	0.5	0.5
	0.6	0	0	0.5	0	0.7	0.8	0
	0.6	0	0	0.5	0	0.7	0.9	0
	0.7	0	0	0.6	0	0.8	0.9	0
	0.5	0.1	0.1	0.4	0.1	0.6	0.7	0
	0.7	0	0	0.6	0	0.8	0.9	0
	0.4	0	0	0.3	0	0.4	0.5	0
	0.6	0	0	0.5	0	0.7	0.9	0
	0.7	0	0	0.6	0	0.8	0.9	0

$$\mathbf{R} \approx \mathbf{U} \times \mathbf{V}^{\mathsf{T}} = \widehat{\mathbf{R}}$$
$$\hat{r}_{i,j} = \mathbf{u}_i^T \mathbf{v}_j$$

Figure 1: Matrix factorisation is an example of a latent factor model, which detects a given number of factors, k (also known as rank), by compressing the user-item matrix R into two lower rank matrices, a user, U (dim(U) $= n \times k$), and item, V (dim(V) $= m \times k$). As the user and item factors share a joint latent space of dimensionality k, the approximate preference, $\hat{r}_{ij}$ of user i for item j can be predicted by the dot product of of the user and item latent factors: $\hat{r}_{ij} = \mathbf{u}_i^\top \mathbf{v}_j$. The resulting rating predictions matrix, $\hat{\text{R}}$ is an approximation of the original ratings matrix, R

using local rules generated from predictions in a user's neighbourhood, motivated by [18], in addition to global rules generated on all predictions. Local fidelity was recognised as an essential criterion of an explanation model by [29].

3 BACKGROUND

3.1 Collaborative Filtering

Collaborative filtering relies only on data of past user behaviour as the basis for making predictions of the preferences of a user. Latent factor models, popularised by [23], are a collaborative filtering approach which predict items to recommend to users based on finding low-dimensional feature vectors to represent users and items. These vectors store latent factors which are unobservable characteristics of the users and items that the model has inferred from the pattern of user-item interactions. An example of a user latent factor could be the strength of a user's preference for crime genre films. The corresponding item factor would represent how typical the film is of the crime genre.

The matrix factorisation approach is illustrated in Figure 1. To predict user ratings, matrix factorisation for implicit feedback as per [21] seeks to minimise the ℓ_2 regularised squared difference between binary preference variables, p_{ij} (=1 when $r_{ij} > 0$; 0 otherwise) and predicted ratings, $\mathbf{u}_i^\top \mathbf{v}_j$, weighted by our confidence in the user's observed preference, c_{ij} (a function of the user rating r_{ij} and confidence parameter α). The regularisation parameter, λ, controls the magnitude of the latent factors to ensure the model does not overfit on the training data. Mathematically, we have the objective function:

$$\min_{\mathbf{u},\mathbf{v}} \sum_{i,j \in R} c_{ij}(p_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)^2 + \lambda \Big(\sum_i ||\mathbf{u}_i||^2 + \sum_j ||\mathbf{v}_j||^2 \Big) \quad (1)$$

To minimise Equation 1, we use Alternating Least Squares (ALS) for efficient optimisation as advised by [21] given the density of the user-item matrix when using implicit feedback datasets ($m.n$ non-null user-item pairs). ALS involves iteratively updating the user and item factors until convergence.

By taking the inner product of the resulting user and item factors, we generate the rating predictions matrix $\hat{\mathbf{R}}$. To generate recommendations for each user, we filter out items which exist in the user's training data, and recommend the top N items with the highest predicted ratings.

Latent factor models for collaborative filtering such as matrix factorisation have been shown to significantly improve the accuracy of recommendations beyond simpler collaborative filtering models such as neighbourhood methods. However, as the latent factors detect unobservable characteristics from the patterns of past user behaviour, they are inherently uninterpretable, especially when the dimension of the latent space exceeds three which can be visualised.

3.2 Association Rule Mining

Association Rule Mining is a data mining technique used to identify relationships between categorical items in a collection. An example of an association rule is the implication {diapers ⇒ beer} where diapers is the *antecedent* (LHS) of the rule and beer is the *consequent* (RHS) of the rule. This rule indicates that if a customer purchases diapers, they are likely to also purchase beer. This format is analogous to IF... THEN... decision rules and provides a natural interpretation of the form {explanation ⇒ recommendation}. For conciseness, we refer the reader to [4] for the basic definitions of association rules. For the purposes of this paper, we are only concerned with itemsets and rules of size 2, i.e. there can only be two items in the rule, one in the antecedent and one in the consequent.

Association rules are extracted from itemsets based on their strength in the collection as evaluated by support, confidence and lift interestingness measures. A rule {X ⇒ Y} holds with support *supp* if *supp*% of the transactions contain X ∪ Y. It is a frequency measure of how often we can expect to observe the pattern. The rule holds with confidence *conf* if *conf*% of transactions that contain X also contain Y as defined by the ratio of their supports *supp*(X ∪ Y)/*supp*(X). It is an estimate of the conditional probability Pr(Y|X) and therefore, predictive ability of the rule. The rule's lift measures the dependency between two items defined as the combined support of the items, *supp*(X ∪ Y), divided by the support if the items were independent, $supp(X) \times supp(Y)$. It measures the predictive ability of the rule relative to the full dataset.

We use the Apriori algorithm [5] to mine association rules due to its simplicity for demonstrating the concepts we explore.

Association rules can be used in their own right as a collaborative filtering recommendation system by considering each user profile in a user-item matrix as a transaction, i.e. the set of items a user has provided their preference for (e.g. products bought, movies watched). Rules can then be generated using a rule mining approach such as [5].

The resulting recommendation rules reflect relationships between items which co-occur in the observed user preferences and can be used to predict the unobserved preferences. Explicitly, to generate a set of recommendations for a user, Sarwar et al. [34] propose the following approach:

(1) The full set of rules are filtered to a subset where the antecedent (X) has previously been liked by the user, but the consequent (Y) is unobserved
(2) The filtered rules are then ranked by support, confidence, or another statistic of the relationships
(3) The top N consequent items of the filtered rules are recommended to the user

We adapt this method as part of our post hoc intepretability approach outlined in Section 4.2.

4 PROPOSED APPROACH

4.1 Theory

As described by Doshi-Velez and Kim in their ICML tutorial, a risk of using interpretable models to mimic a complex model is that there may be a gap between what the actual model is doing and your mimic model is doing. Choosing a simple model which has a logical relationship to the complex model is more likely to yield loyal explanations. We believe there are some natural similarities between matrix factorisation and association rules that could make rules a strong proxy model, particularly for the recommendation problem.

Matrix factorisation models user behaviour by generating user and item factors in a latent space which encodes the model's belief of the similarities and weightings between users and items as learned from the training data. The matrix factorisation predictions captures each user's overall interest in each item's characteristics [24]. The predicted user profiles, $\hat{R}_i$ therefore represent the model's belief of user preferences. Association rule models have been described as a natural representation of how humans make decisions [35] given their *'and', 'or'* structure and have been succesfully applied to modelling user behaviour [13, 17].

As both matrix factorisation and association rule models are used to model user behaviour, we believe they have sufficient similarities that will allow us to exploit the expressive power of association rules to describe how the matrix factorisation model behaves. The explanation models discover patterns of items that frequently co-occur together in the matrix factorisation predicted user preferences; this allows us to understand the conditions of a user's training data that will cause the model to make a certain recommendation. These simple sets of conditions provide a representation of the logic the model used to score the user and item factors.

4.2 Approach

Our proposed approach is illustrated in Figure 2. We outline the corresponding steps in detail below.

1. Input data: Our input data is a user-item matrix of observed interactions between users and items. To account for the uncertainty in inferring preferences based on interactions, we convert this implicit feedback into preference and confidence variables as described by [21]. This results in a dense, long matrix of user-item interaction, preference and confidence values.

2. Train Matrix Factorisation model (black-box): Matrix Factorisation is performed on the training user-item data from step 1 to predict the unobserved ratings by minimising the objective function from Equation 1.

To learn the optimal user and item factors to minimise Equation 1, Alternating Least Squares is used.

3. Output predictions: The output of matrix factorisation training is a completed user-item matrix with predicted ratings of all items, for all users. This is the input to step 4.

To process the output to generate recommendations, the predicted ratings are filtered to remove items from the training data as we do not want to recommend items a user has previously seen. The top N predicted items for each user form our *recommendation sets* which we will use in evaluation of the matrix factorisation model.

4. Train Association Rules (interpretable model): For our baseline explanation model, the set of transactions, T, is generated by taking the top D predicted items for each user from the unfiltered matrix factorisation output (i.e. including observed preferences). Using all user transactions will explain the global behaviour of the matrix factorisation model. We later extend this to consider local behaviour by using transactions from a smaller community of users.

From this point, we proceed with the approach for generating recommendation rules described in Section 3.2. Rule criteria are specified: min_supp = 0.1, min_conf = 0.1, min_lift = 0.1, max_len = 2. We set low thresholds in order to ensure interesting rules can be generated for items in the long-tail that have a low support in the collection.

5. Output association rules. The output of step 4 is a list of all rules representing relationships between items in the matrix factorisation predictions and their corresponding support, confidence and lift measures. In order to generate explanations for the matrix factorisation recommendations for a given user i, we filter the rules to a subset where the antecedent, X, is in user i's training data, and the consequent, Y, is not in the training data such that the rule $\{X \Rightarrow Y\}$ provides the explanation "Because you watched X, we recommend Y".

We order the resulting subset by a specified interestingness measure (support/confidence/lift) and store the top N consequents of the rules as our *explainable items*, with the corresponding antecedents as explanations.

Algorithm 1 details how to generate the explanation model as described in steps 4 and 5 above.

To evaluate the quality of the explanation model, the explanation set is compared to the recommendation set for each user. Where a matrix factorisation recommendation is present in the consequent

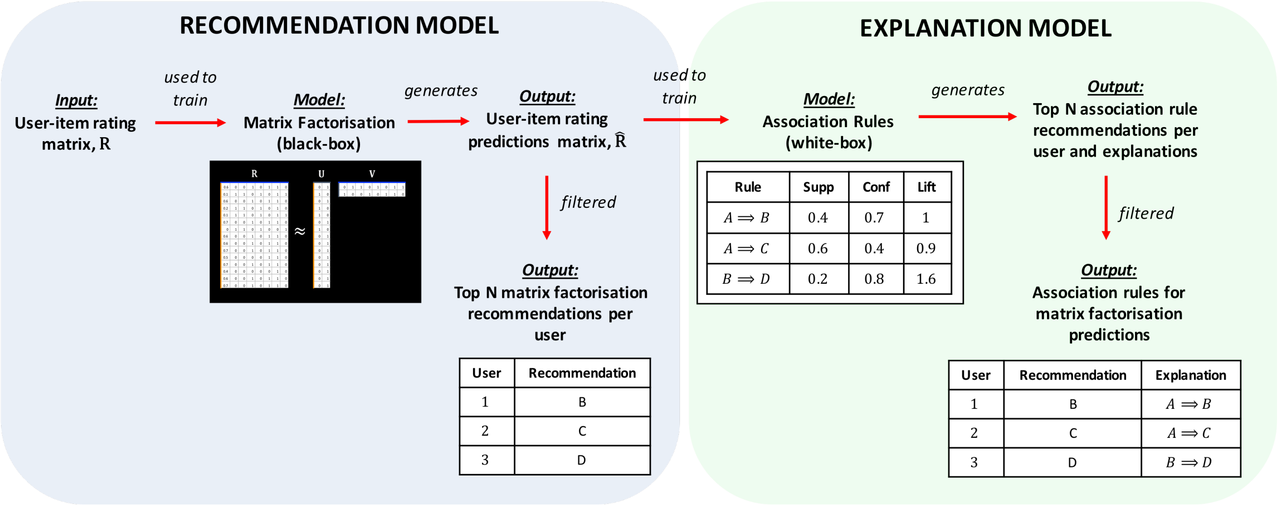

RECOMMENDATION MODEL
Input:
User-item rating matrix, R
used to train
Model:
Matrix Factorisation (black-box)
R
U
V
≈
generates
Output:
User-item rating predictions matrix, R̂
filtered
Output:
Top N matrix factorisation recommendations per user
User | Recommendation
1 | B
2 | C
3 | D
used to train
EXPLANATION MODEL
Model:
Association Rules (white-box)
Rule | Supp | Conf | Lift
A ⇒ B | 0.4 | 0.7 | 1
A ⇒ C | 0.6 | 0.4 | 0.9
B ⇒ D | 0.2 | 0.8 | 1.6
generates
Output:
Top N association rule recommendations per user and explanations
filtered
Output:
Association rules for matrix factorisation predictions
User | Recommendation | Explanation
1 | B | A ⇒ B
2 | C | A ⇒ C
3 | D | B ⇒ D

Figure 2: Schematic diagram of proposed approach for generating an explanation model by training association rules on the output of a black-box matrix factorisation recommendation model. Step numbers correspond to step-by-step approach description

Algorithm 1 Approximate matrix factorisation using global association rules

```
Input: Matrix factorisation predictions $\hat{R}$; training data
For each user $i$, generate a transaction list $T_i$ of the index of top D matrix factorisation predictions, $\hat{R}_i$
Generate the set $\mathcal{Z}_i$ of rules $(X \Rightarrow Y)$ that satisfy $min\_supp$, $min\_conf$, $min\_lift$ criteria from all transactions $T$.   ▹ Rules generated by the apriori algorithm using the apyori [27] Python package
for all users, $i = 1 \dots N$ do
    Compute the list {unseen} of items $Y$ where $X \Rightarrow Y$ if $X \in$ {train} and $Y \notin$ {train}.
    Order {unseen} by supp/conf/lift. Compute {recommended} = {unseen}[: top_n]
    Return list of recommended items, {recommended} and corresponding rules $X \Rightarrow Y$ as explanations.
end for
Output: {unseen}
```

of a rule, an explanation can be provided by the antecedent. Our full evaluation approach is outlined in Section 5.2.

5 EXPERIMENTS

We aim to prove that approximating a black-box model with an interpretable model can maintain the high predictive accuracy of the black-box model whilst improving interpretability by extracting explanations that can be used to understand model behaviour.

To investigate this, we mine association rules on the predictions of a matrix factorisation recommendation model. Our baseline experiment considers rules built on predictions for *all* users; we refer to these as *global* rules. The global rules assign a rule with the same weight for predictions of all user-item interactions. However, given the main assumption of collaborative filtering is that users who have previously displayed similar preferences for an item are likely to also share preferences on other items, the global rules are not appropriate for approximating these neighbourhoods of similar users. As such, we investigate whether a smaller neighbourhood of the most similar users to an active user might generate higher fidelity rules that explain how the model behaves in the vicinity of the user. We experiment with *local* rules mined from a fixed size neighbourhood of users based on cosine similarity of the matrix factorisation predictions, and cluster rules mined from variable size groups of users who are close in the latent space. We hypothesise that these more personalised explanations will have higher fidelity to the black-box recommendation model than the global rules.

We also compare the explanatory performance of an interpretable recommendation model to our black-box model with post hoc interpretability in order to prove the motivations of the approach; explicitly, we investigate whether the black-box model can outperform the predictive performance of the interpretable model, whilst achieving comparable or superior interpretability.

5.1 Data description

5.1.1 Channel 4. Channel 4 is a British public service broadcaster. Channel 4 provide access to their previously aired episodes to watch on their Video On Demand (VOD) platform, All 4 (http://www.channel4.com). For this research paper, data collected is from clickstream data of users interactions with the All 4 platform during the period January 2016 to June 2017. This is an example of an implicit feedback dataset where our *items* are TV programmes. A rating is calculated as the number of unique episode views for a programme by a user during the time period. A view is defined as a video watched until at least 50% completed.

We generate our training set of user views from January 2016 - December 2016 and take a subsample of 7327 users. Our test dataset is generated from the same users for the period January 2017 - June 2017. This temporal split allows us to simulate the recommendation scenario during evaluation as for each user.

To be included in the sample, we imposed constraints that users must have had greater than 5 views and programmes must have had greater than 10 views in order to avoid adding noise to the data

Table 1: Data Summary

	Channel 4		MovieLens	
	Training	Test	Training	Test
# users	7327	4215	942	456
# programmes	681	671	1386	1117

with users and programmes for which we do not have sufficient information.

We remove any programmes that are not in the intersection of training and test programmes to avoid evaluating on test programmes which were not available during the training period and therefore could not be recommended.

For the TV programme recommendation scenario, it is trivial to predict that a user will continue to watch a programme they have previously seen. As such, for our recommendation implementation, we do not recommend any items a user has previously seen. To account for this during evaluation, we remove programmes from a user's test set if the programmes exist in the user's training set.

To account for a potential bias towards programmes which have more episodes and are naturally watched more frequently (for example, soaps such as 'Hollyoaks' have daily episodes, compared to one-off documentaries such as '12 Year Old Lifer'), the ratings are normalised by dividing the number of views for a programme by the number of available historical episodes of the programme for each user.

The resulting dataset statistics are summarised in Table 1.

5.1.2 MovieLens 100K. In order to account for any potential sampling biases in the Channel 4 data collection and prove the robustness of our approach, we repeat our analyses on the MovieLens 100K dataset [16]. The MovieLens 100K dataset is a benchmark dataset used to assess and compare recommendation systems. The data consists of explicit feedback of user ratings of movies on a scale of 1-5 collected from the MovieLens website (http://movielens.org). Each user in the dataset has rated at least 20 movies. The ratings for each user are normalised by the user's average rating and the ratings are converted to implicit, positive-only data by thresholding the ratings such that a rating greater than 3 is considered presence of a view $p_{i,j} = 1$, and 0 otherwise. These preprocessing steps are in line with the standard approaches in the literature [18, 23]. The dataset statistics are summarised in Table 1.

5.2 Evaluation approach

As highlighted by [12], evaluating interpretability is an ill-defined problem due to the multiple motivations and lack of consensus on the definition of interpretable machine learning. Our proposed evaluation approach is a unique contribution to the interpretability literature for how to evaluate post hoc interpretability.

It is important to recognise the distinction between evaluating the recommendation models and evaluating the explanation models. The post hoc interpretability approach decouples recommendations and explanations by generating a separate explanation model. As such, in order to ensure the evaluation approach is as loyal to the problem as possible, we must also treat the evaluation of the recommendation and explanation models separately.

5.2.1 Evaluating recommendation models - Generalisation. To assess the generalisation performance of our recommendation model, we use the boolean retrieval metrics precision and recall to evaluate the model's predictive performance on an unseen test set. When applied to the recommendation problem in the offline setting, the *retrieved set* refers to the items recommended to a user, and the *relevant set* refers to the items in a user's test set (as these are the user's 'correct' preferences which the model should predict):

Precision measures the proportion of items recommended which were also in the user's test set (i.e. how many of the retrieved items were relevant). *Recall* measures the proportion of test set items which were recommended (i.e. how many of the relevant items were retrieved).

5.2.2 Evaluating explanation models - Model Fidelity. The aim of this project is to produce a proxy model which approximates a complex model with high fidelity. As such, we define Model Fidelity which measures the faithfulness of the explanation model in reproducing the results of the complex recommendation model. We extend the standard confusion matrix to add an explainable variable to allow us to illustrate these definitions as shown in Table 2.

Model Fidelity will be our core measure of success for this project as the success of an explanation model should be independent of the predictive success of the model it is explaining.

Abdollahi and Nasraoui [1] use information retrieval metrics for their model-based explanations. We adapt recall to the post hoc interpretability scenario. In this context, we are considering the association rule model as the retrieval system and therefore the *retrieved set* are the items found by the association rules; we will refer to these items as *explainable items*. As we are aiming to reproduce the results from the recommendation model, the *relevant set* are the matrix factorisation recommendations; we will refer to these items as *recommended items*. The hit set is the intersection of the *explainable items* and the *recommended items*, i.e. the matrix factorisation (MF) recommendations that the association rule (AR) system can explain.

As such, we define **Model Fidelity** as the proportion of matrix factorisation recommendations that can be explained by the association rule explanation model:

$$\text{Model Fidelity} = \frac{|\text{MF recommended items} \cap \text{AR retrieved items}|}{|\text{MF recommended items}|} \tag{2}$$

$$= \frac{|\text{explainable items} \cap \text{recommended items}|}{|\text{recommended items}|}$$

$$= \frac{\text{A+C}}{\text{A+C+E+G}}$$

5.3 Implementation

5.3.1 Matrix factorisation recommendation model. To build our matrix factorisation model, we tune the rank k, regularisation parameter λ and the confidence parameter α, using exhaustive grid search and a training/validation split of the full training data of 65%/35%.

We select the top N predicted items as our recommendation set where N = 20.

Using the best performing parameters on the validation set (measured by recall), we retrain the model on the full training data and evaluate performance on the unseen test set. Our most perfomant

Table 2: Extended confusion matrix showing test, recommended and explainable variables

	Explainable		Not explainable	
	Recommended	Not recommended	Recommended	Not recommended
Test	A	B	E	F
Not test	C	D	G	H

Table 3: Matrix Factorisation generalisation results - Channel 4 (C4) and MovieLens (ML)

Data	Rank	λ	α	Metric	Set	Result
C4	100	0.3	0.0005	Precision	Validation	0.13273
					Test	0.04548
				Recall	Validation	0.31299
					Test	0.32529
ML	10	0.1	0.1	Precision	Validation	0.18904
					Test	0.28596
				Recall	Validation	0.29744
					Test	0.31152

model on the Channel 4 dataset is a rank 100 model as detailed in the top panel of Table 3. Given the high dimensionality of the latent space ($k = 100$), this is our most complex model and highly uninterpretable. Therefore, this model will be the focus of our experiments on extracting explanations of the model behaviour.

We perform the same hyperparameter tuning on the MovieLens dataset. The best MovieLens model is a rank 10 model as detailed in the bottom panel of Table 3. This is comparatively low in complexity but the recall scores are comparable to the optimal rank 100 Channel 4 model. We suggest that these differences may partly arise from the nature of the raw datasets of explicit (MovieLens) and implicit (Channel 4) feedback, as well as nuances in the nature of movie compared to TV recommendation scenarios. As we proceed with our experiments, we focus on the results for the Channel 4 dataset, whilst using MovieLens as a robustness check.

Further performance gains are likely achievable by further parameter tuning and extending the model to incorporate biases or temporal effects as described in [23]. However, our results are in line with the literature [18] and as the focus of our research is on the interpretability of the models and the recommendations will not be presented to users, these results are appropriate for our purposes.

5.3.2 Experiment 1: Using global association rules to approximate matrix factorisation recommendation model. For global rules, we build each explainable model on the top D matrix factorisation predictions for *all* user profiles to create a set of transactions, T. We vary D to analyse the impact of transaction sizes on fidelity and find that 30 items is the optimal number of items for a transaction, suggesting that little additional information is provided by the further items. We vary the interestingness measure used to order the global rules using support, confidence and lift as defined in Section 3.2. The full model fidelity results for global rules to explain the rank 100 recommendation model are reported in the top panel of Table 4 where D=30.

The global rules are only able to explain 50% of predictions. As the global rules are mined from all users, the most popular items will regularly feature in the top matrix factorisation predictions and subsequently dominate support in the global rules. Such rules will have fidelity for users interested in popular content but will not personalise to users with specific interests. Further, as we limit ourselves to size 2-itemset rules in order to ensure the explanations are simple and interpretable, the global rules are limited in their explanatory power of the complex scoring logic of the matrix factorisation model.

Table 4: Model Fidelity of association rules for explaining recommendation model (rank 100), by number of neighbours (K) for global and local rules; number of clusters for cluster rules; and interestingness measures

Rules	K	Interestingness	Model Fidelity
Global	N	Support	0.522369
		Confidence	0.568602
		Lift	0.423591
Local	10	Support	0.828272
		Confidence	0.842889
		Lift	0.412679
	50	Support	0.791095
		Confidence	0.817715
		Lift	0.452805
	100	Support	0.770759
		Confidence	0.799536
		Lift	0.44886
Rules	**# clusters**	**Interestingness**	**Model Fidelity**
Cluster	3000	Support	0.842794
		Confidence	0.852559
		Lift	0.66025
	700	Support	0.767592
		Confidence	0.799768
		Lift	0.460257
	10	Support	0.598649
		Confidence	0.640951
		Lift	0.423059

5.3.3 Experiment 2: Using local association rules based on a neighbourhood of users with high cosine similarity of matrix factorisation predictions. To analyse whether rules mined from a subset of similar users are a more suitable proxy representation of the matrix factorisation recommendation model than global rules, we build local rules for each user and investigate the impact of the number of users in the neighbourhood (K) by varying K ∈ [10,50,100].

Table 4 compares the fidelity of the explainable items from the global and local rules to the recommended items.

As hypothesised, global rules are worst performing, given basing rules on all users is not a valid reflection of any one user's preferences as discussed. We observe that the rules generated from smallest neighbourhood size of K=10 have the highest fidelity to the matrix factorisation recommendations when ordering rules by confidence or support.

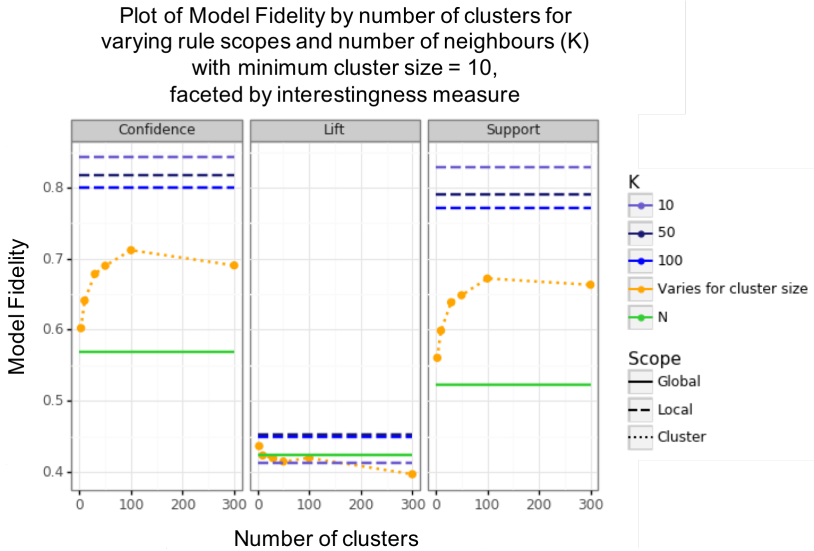


Figure 3: Model Fidelity for global, local, and cluster rules for optimal 100 rank model with minimum cluster size of 10. Number of clusters varies from 10 to 300

Table 5: Explanation examples from global and cluster rules (# clusters = 10) in format {explanation ⇒ recommendation}

Global rules
24 Hours in A&E ⇒ The Undateables
24 Hours in A&E ⇒ First Dates
24 Hours in A&E ⇒ Naked Attraction
24 Hours in A&E ⇒ Tattoo Fixers
24 Hours in A&E ⇒ Gogglebox
Cluster rules
Born to Be Different ⇒ One Born Every Minute
Born to Be Different ⇒ 24 Hours in Police Custody
Born to Be Different ⇒ 999: What's Your Emergency?
24 Hours in A&E ⇒ Gogglebox
Kids on the Edge ⇒ Royal Navy School

5.3.4 Experiment 3: Using local association rules based on cluster of similar users in latent space. In experiment 2, each user is assumed to have the same number of nearest neighbours. This is not a realistic assumption as a user with niche interests (for example, programmes about trains) is likely to have a small number of users who represent his/her preferences, in comparison to a user who is interested in popular content.

As such, we use hierarchical agglomerative clustering to group users into variable sized communities in the latent space using complete linkage and cosine similarity. The flexibility for clusters to vary in size was the primary motivation for using this cluster method compared to alternative techniques such as K-means which typically results in spherical and equally sized clusters. Rules are generated at the cluster-level by generating transactions from the matrix factorisation predictions from each user in the cluster. We tune the number of clusters as this impacts the sizes of the communities; many clusters will result in a smaller mean cluster size.

The bottom panel of Table 4 displays the cluster rules when the latent space is grouped into clusters of users based on their matrix factorisation recommendations. Investigating the impact of the number of clusters, by dividing the latent space into 10 clusters we achieve greater model fidelity than the global rules for all interestingness measures. This provides evidence for our hypothesis that cluster rules are a more representative proxy of the matrix factorisation model as they take into account the similarity of users in line with the collaborative filtering assumption described in Section 3.1. Table 5 shows the top 5 rules to explain the matrix factorisation recommendations from the global and cluster rules for one sample user. The rows in bold denote rules that successfully reproduced a matrix factorisation recommended item and provide a corresponding explanation of a programme the user has previously watched. From a qualitative analysis of the rules, we can recognise that the user has previously watched documentaries. As described in Experiment 1, the global rules are dominated by popular programmes and are unable to reproduce the interest in documentaries that the matrix factorisation model has identified. The top 5 cluster rules successfully reproduce 4 matrix factorisation recommendations; the explanations for the recommendations make sense as these are all other documentaries or factual entertainment programmes.

Comparing the local rules of fixed size neighbourhood and cluster rules of variable size neighbourhood, we observe that the cluster rules manage to exceed the model fidelity of the best performing local rules (K=10) when there are 3000 clusters. Cluster rules have the additional computational advantage over local rules as the rules do not need to be mined for each individual user whilst the number of clusters is less than the total number of users, N. However, we recognise that our approach degenerates at high numbers of clusters as you can find single user clusters such that the rules are mined on only their own matrix factorisation predictions (1 transaction). This will result in a set of pairwise rules between all top D items in the recommendations with equal support, confidence and lift. This does not give us any information about how the matrix

Table 6: Recall of interpretable association rules and black-box matrix factorisation recommendation systems compared to test set (generalisation)

Recommendation Model	Rank	Recall
Matrix Factorisation	100*	0.32529
	100	0.23887
Recommendation Model	**Interestingness**	**Recall**
Association Rules	Support	0.26119
	Confidence	0.2849
	Lift	0.10407

factorisation model is behaving; it is purely a description of what it predicts for the single user based on their training data.

Therefore, we set a minimum number of users in a cluster for the items to be considered *explainable*. For our recommendation scenario, we set the minimum cluster size to 10, in line with the size 10 user neighbourhoods in experiment 2. The results of repeating the previous analyses with this constraint on the rank 100 optimal model are shown in Figure 3. The cluster rules now cannot exceed 70% model fidelity demonstrating that this constraint was too harsh as the local rules outperform the cluster rules.

This observation highlights the need to establish a balanced view of explainability which describes both the fidelity of the explanations and whether they are valuable for end users. For example, when the approach degenerates for very small neighbourhood sizes, whilst you would achieve high model explainability scores according to our metrics, we also need a more qualitative evaluation metric that captures that the explanations are not informative to the user. By determining what would be an acceptable balance of these metrics, you could tune the size of the neighbourhood (for example) to achieve this. As such, user testing should be an important component of interpretability evaluation to understand the priorities of such metrics.

In light on this, analysis of Figure 3 which incorporates the minimum cluster size constraint demonstrates how this could work in practice. We observe an elbow in the model fidelity scores when the number of clusters is set to 100 for the confidence interestingness measure. This suggests that the explanations generated by these rules may be more fit for purpose, despite the higher model fidelity achieved by the local rules.

5.3.5 Experiment 4: Comparison of interpretable model and post hoc interpretability. The strong motivation of using post hoc interpretability instead of interpretable models is that the high accuracy performance from the complex model can be maintained, whilst improving a user's understanding of the model through explanations. We now consider whether we have satisfied our aim to demonstrate the benefits of post hoc intepretability by comparing generalisation performance of our black-box latent factor recommendation model and an interpretable association rule *recommendation model*.

To train our interpretable recommendation model, we follow the process outlined in Section 3.2 as per [34]. We experiment with different strength criteria for the interpretable model. As the complex matrix factorisation model is trained on all user profiles, we generate global rules for our interpretable model by considering each user profile as a transaction to ensure a fair comparison. If we were to extract rules from a clustering of the training user-item matrix then we would be providing the interpretable model with additional information and therefore not providing a fair comparison.

Table 6 compares the predictive performance of the recommendation models on an unseen test set, measured by recall. We report the results of the interpretable model with the highest recall when ordering the rules by confidence with strength criteria (min_supp, min_conf, min_lift) ∈ (0.001, 0.001, 1.0), alongside the corresponding support and lift ordering results. As hypothesised, the interpretable recommendation rule models are not flexible enough to model the true user preferences as compared to our optimal complex model (100*), providing further validation of the post hoc interpretability approach.

We also compare the interpretable model to an unoptimised matrix factorisation model of the same rank $k = 100$. It's notable that the interpretable model is able to outperform the unoptimised high rank model as the high flexibility of the black-box model causes it to fit noise.

6 CONCLUSIONS

We set out to prove that approximating a black-box model with an interpretable model can maintain the high predictive accuracy of the recommendation model whilst improving interpretability by extracting explanations that can be used to understand model behaviour. We have shown that predictive performance of complex models can be maintained whilst gaining the benefits of increased interpretability; our complex 100 dimensional model achieves the same accuracy of 32.5% whilst gaining interpretability where up to 84% of the model's recommendations can be explained, therefore successfully mitigating the accuracy-interpretability trade-off. This is a very positive result as it enables explanations without the need for expensive external data sources or constraining an existing model, that limit the existing approaches. We have provided further evidence of the benefits of post hoc interpretability by comparing performance to an interpretable association rule recommendation model. As hypothesised, the intepretable model cannot match the flexibility of matrix factorisation to model user behaviour.

We have also contributed to the ill-defined problem of evaluating interpretability. We outline a sound approach to evaluating post hoc interpretability for recommendation systems by recognising the important distinction between evaluating the recommendation and explanation models.

We decided to limit our explanation rules to itemsets of size 2 in order to ensure interpretability and avoid cognitive overload. However, this limits the power of association rules to model the complex user preferences as represented by the latent space. We believe more interesting and loyal rules could be generated by considering larger itemsets, with greater than 1 item in either the antecedent or consequent. Psychology research [26] has shown that humans can process up to 7 ± 2 cognitive entities (e.g. IF... THEN... rule criteria) at one time. In addition, improvements in fidelity could potentially be achieved by using more flexible proxy models such as Bayesian networks as used in [33] to generate explanations.

As the computational complexity of association rules scales with the number of transactions and number of items in a dataset, we limited the size of our top N recommendations to 20 and used small industry datasets. Given the superior performance of matrix factorisation over simple neighbourhood methods is most evident on large datasets, it could be worthwhile incorporating methods for scalable rule mining to our approach.

Our approach generates explanations of the format: “Because you watched X, we recommend Y”. Given the multiple reasons for why a user may watch a programme, a promising future extension would be to generalise this approach to handle alternative explanation types. As recognised by [28], explanation flexibility is one of the key benefits of post-hoc interpretability as it allows the complex model to remain fixed whilst using multiple explanation models of varying levels of interpretability. The explanation type could then be personalised to the user.

As potential future work, it would be interesting to enhance our model using textual data, in a similar approach to the topic modelling approaches used for explainable collaborative filtering to understand the themes associated with a show that result in it being explainable for another show recommendation. This enhancement would further help to facilitate alternative explanations, for example “Because of your interest in theme X, we recommend Y”.

ACKNOWLEDGMENTS

This research was supported by Channel 4 Television.

REFERENCES

[1] Behnoush Abdollahi and Olfa Nasraoui. 2016. Explainable Matrix Factorization for Collaborative Filtering. *WWW (Companion Volume)* (2016), 5–6. https://doi.org/10.1145/2872518.2889405

[2] Behnoush Abdollahi and Olfa Nasraoui. 2016. Explainable Restricted Boltzmann Machines for Collaborative Filtering. *2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)* Whi (2016). arXiv:1606.07129

[3] Behnoush Abdollahi and Olfa Nasraoui. 2017. Using Explainability for Constrained Matrix Factorization. *Proceedings of the Eleventh ACM Conference on Recommender Systems - RecSys '17* (2017), 79–83. https://doi.org/10.1145/3109859.3109913

[4] Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. 1993. Mining Association in Large Databases. *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93* (1993), 207–216. https://doi.org/10.1145/170036.170072

[5] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. *Proceedings of the 20th International Conference on Very Large Data Bases* (1994), 487–499.

[6] Marharyta Aleksandrova, Armelle Brun, Anne Boyer, and Oleg Chertov. 2014. What about interpreting features in matrix factorization-based recommender systems as users? *CEUR Workshop Proceedings* 1210, 1 (2014).

[7] David Baehrens and T Schroeter. 2010. How to explain individual classification decisions. *The Journal of Machine Learning Research* 11 (2010), 1803–1831. arXiv:0912.1128 http://dl.acm.org/citation.cfm?id=1859912

[8] Bart Baesens, Rudy Setiono, Christophe Mues, and Jan Vanthienen. 2003. Using neural network rule extraction and decision tables for credit-risk evaluation. *Management science* 49, 3 (2003), 312–329. https://doi.org/10.1287/mnsc.49.3.312.12739

[9] Yang Bao, Fang Hui, and Jie Zhang. 2014. TopicMF: Simultaneously Exploiting Ratings and Reviews for Recommendation. *Aaai* (2014), 2–8. http://www.ntu.edu.sg/home/zhangj/paper/aaai14-bao.pdf

[10] Nahla Barakat and Andrew P. Bradley. 2010. Rule extraction from support vector machines: A review. *Neurocomputing* 74, 1-3 (2010), 178–190. https://doi.org/10.1016/j.neucom.2010.02.016

[11] R Caruana, H Kangarloo, J D Dionisio, U Sinha, and D Johnson. 1999. Case-based explanation of non-case-based learning methods. *Proceedings. AMIA Symposium* (1999), 212–5. http://www.ncbi.nlm.nih.gov/pubmed/10566351{%}0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2232607

[12] Finale Doshi-Velez and Been Kim. 2017. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv* Ml (2017), 1–13. arXiv:arXiv:1702.08608v2 https://arxiv.org/pdf/1702.08608.pdf

[13] Jerome H. Friedman and Nicholas I. Fisher. 1999. Bump Hunting in High-Dimensional Data. *Statics and Computing* 9 (1999), 123–143. https://doi.org/10.1023/A:1008894516817

[14] Bryce Goodman and Seth Flaxman. 2016. EU regulations on algorithmic decision-making and a "right to explanation". *2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)* Whi (2016), 26–30. arXiv:1606.08813 http://arxiv.org/abs/1606.08813

[15] Prem K. Gopalan, Laurent Charlin, and David Blei. 2014. Content-based recommendations with Poisson factorization. *NIPS Advances in Neural Information Processing Systems* (2014), 3176–3184. http://papers.nips.cc/paper/5360-content-based-recommendations-with-poisson-factorization

[16] F Maxwell Harper and Joseph A Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2015), 19:1—-19:19. https://doi.org/10.1145/2827872

[17] John R Hauser, Olivier Toubia, Theodoros Evgeniou, Rene Befurt, and Daria Dzyabura. 2010. Disjunctions of Conjunctions, Cognitive Simplicity, and Consideration Sets. *Journal of Marketing Research* 47, 3 (2010), 485–496. https://doi.org/10.1509/jmkr.47.3.485

[18] Reinhard Heckel, Michail Vlachos, Thomas Parnell, and Celestine Duenner. 2017. Scalable and interpretable product recommendations via overlapping co-clustering. *Proceedings - International Conference on Data Engineering* (2017), 1033–1044. https://doi.org/10.1109/ICDE.2017.149 arXiv:1604.02071

[19] Jonathan L Herlocker, Joseph a Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (2000), 241–250. https://doi.org/10.1145/358916.358995 arXiv:48

[20] Thomas Hofmann. 2004. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems* 22, 1 (2004), 89–115. https://doi.org/10.1145/963770.963774

[21] Yifan Hu, Chris Volinsky, and Yehuda Koren. 2008. Collaborative filtering for implicit feedback datasets. *Proceedings - IEEE International Conference on Data Mining, ICDM* (2008), 263–272. https://doi.org/10.1109/ICDM.2008.22

[22] Pang Wei Koh and Percy Liang. 2017. Understanding Black-box Predictions via Influence Functions. *arXiv preprint arXiv:1703.04730* (2017). arXiv:1703.04730 http://arxiv.org/abs/1703.04730

[23] Yehuda Koren. 2009. Collaborative filtering with temporal dynamics. *Proc. of KDD '09* (2009), 447–456. https://doi.org/10.1145/1557019.1557072

[24] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/MC.2009.263 arXiv:ISSN 0018-9162

[25] Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding Neural Networks through Representation Erasure. *Arxiv* (2016). arXiv:1612.08220 http://arxiv.org/abs/1612.08220

[26] George A. Miller. 1956. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review* 63, 2 (1956), 81–97. https://doi.org/10.1037/h0043158 arXiv:arXiv:1011.1669v3

[27] Yu Mochizuki. 2016–. apyori: PyPI package for Python. (2016–). https://pypi.org/project/apyori/ [Online; accessed 20/05/2018].

[28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Model-Agnostic Interpretability of Machine Learning. *ICML Workshop on Human Interpretability in Machine Learning* Whi (2016). arXiv:1606.05386

[29] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?". *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16* (2016), 1135–1144. https://doi.org/10.1145/2939672.2939778 arXiv:1602.04938

[30] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation Extraction with Matrix Factorization and Universal Schemas. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* June (2013), 74–84. http://www.aclweb.org/anthology/N13-1008

[31] Marco Rossetti, Fabio Stella, and Markus Zanker. 2013. Towards explaining latent factors with topic models in collaborative recommender systems. *Proceedings - International Workshop on Database and Expert Systems Applications, DEXA* (2013), 162–167. https://doi.org/10.1109/DEXA.2013.26

[32] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. 2007. Restricted Boltzmann machines for collaborative filtering. *Proceedings of the 24th international conference on Machine learning - ICML '07* (2007), 791–798. https://doi.org/10.1145/1273496.1273596

[33] Ivan Sanchez, Tim Rocktaschel, Sebastian Riedel, and Sameer Singh. 2015. Towards Extracting Faithful and Descriptive Representations of Latent Variable Models. *AAAI Spring Symposium on Knowledge Representation and Reasoning* 3 (2015), 35–38.

[34] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2000. Analysis of recommendation algorithms for e-commerce. *Proceedings of the 2nd ACM conference on Electronic commerce - EC '00* (2000), 158–167. https://doi.org/10.1145/352871.352887 arXiv:117

[35] Tong Wang, Cynthia Rudin, Finale Doshi-Velez, Yimin Liu, Erica Klampfl, and Perry Macneille. 2015. Bayesian Or's of And's for Interpretable Classification with Application to Context Aware Recommender Systems. (2015), 1–40. arXiv:1504.07614

[36] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval - SIGIR '14* (2014), 83–92. https://doi.org/10.1145/2600428.2609579