RECOMMENDING TOP-N BANKING PRODUCTS TO CUSTOMER BASED ON IMPLICIT DATASET

Surya Prakash Tripathi

Under the supervision of
Vinay Katiyar

Thesis

A thesis submitted in partial fulfilment of the requirements of Liverpool John Moores University for the degree of Masters in Data Science

FEBRUARY 2020

Abstract

A primary task of recommendation system is to improve the personalization of the Product/information/content by analyzing the available data.
Nowadays too much information is available on the web and even on banking channels (internet banking, mobile banking or branch banking etc.). So, it is important to make sure that correct information should be passed to correct person and it should not be lost in between.
To make sure this we need to present personalize service to customer as per their needs. We also need to monitor the behaviour (purchase pattern) of our customer continuously without explicitly taking their time for rating our product. As people are nowadays very busy in their life hence taking explicit feedback is difficult. So, we need to develop a system which analyze implicit feedback of the customer and present appropriate services/product to the customer.
We have used matrix factorization and model based collaborative filtering approach to develop a recommendation system on implicit dataset to address the issues of personalization of the product. We have included notion of confidence with Alternating least squares (ALS) to address how strong is our understanding about the user's behaviour on the product.
Confidence parameter is important because in implicit data value zero can have following meanings: Either user is not aware of the product or he/she is not able to buy the product due to Price or availability, or even it may be the chance that user find that product irrelevant for him/her. So, if we see zero in front of any product then we need to make sure that what is the confidence of this particular situation. If a user play one song just one time and other song 1000 times then both these songs will be recorded as one on implicit dataset so buy seeing this we can say our confidence is high that user likes this song if he is playing that multiple times. If user has played any song single time then it can be by mistake or maybe he/she just tried but did not like that song. So, we need to understand the confidence of these actions and same needs to be incorporated in our algorithm to provide better recommendation experience.

Keywords: Implicit feedback, Recommendation System, ALS, Banking, Matrix factorization, Machine Learning, Collaborative Filtering.

List of Figures

Dedication

I dedicate my dissertation work to my family. A special feeling of gratitude to my loving father Shri Mahesh Narayan Tripathi and My mother Srimati Poonam Devi, Special thanks to my brother Shashi Prakash Tripathi and Ved Prakash Tripathi who have never left my side and are very special.

Acknowledgment

I would like to express my sincere appreciation to my thesis supervisor, Mr. Vinay Katiyar and Prof. Manoj who has the substance of a genius: he convincingly guided and encouraged me to be professional and do the right thing even when the road got tough. Without his continuous help and guidance, the goal of this project would not have been realized.

My student mentor, Aunnesha and Ishrat have been very valuable for me. Their support and wisdom definitely improved my skills in doing research.

I am also grateful to my academic committee members for their constructive comments in improving the research in this thesis.

Table of Contents

Chapter 1 Introduction

1.1 Background of the study

During the last few decades, with the rise of content over the web, it's very crucial to understand your audience and present them information which they are looking for or what they need? but due to heavy amount of information on the web or applications, users usually miss the relevant information.

Presenting information/item as per the user's taste can increase the probability of buying that item or reading that information (in case of article/news).

Now the question is how to understand the taste of the user, Infact the taste of millions of users and personalize the taste of each of these users so that they can get the information/item what they like or looking for.

The answer of above questions are Recommendation System and Wikipedia defines Recommendation system as follow:

"*A recommender system or a recommendation system (sometimes replacing 'system' with a synonym such as platform or engine) is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. They are primarily used in commercial applications.*" [2]

With the rise of applications like YouTube, Amazon, Netflix and Flipkart the recommendation system started taking more and more importance in our life and below facts support this statement:

- More than 80% TV shows/movies people watch on **Netflix** are discovered through the Netflix's Recommendation system.
- 20% to 35% sale volume of Amazon is derived from Recommendation system.
- Google news recommendations generate 38% more clickthrough.

Netflix, Amazon and Google news spend heavily on Recommendation system and with bunch of Recommendation algorithms (Ensemble) they try to understand the taste of their user and suggest them personalized information. And this is one of the main reason for the success of these companies.

Recommendation system is heavily used in E-commerce, Video streaming and music streaming Products but still **Banking** industry is bit away from this algorithm and they are not leveraging the Recommendation system as media streaming or E-commerce companies are leveraging.

As Banking industry also deals with Products (like Credit card, debit, Loans, Savings, term deposit etc.) hence know your customer (KYC) plays an important role and catering product based on their age, category, financial status becomes critical because this can increase the probability of buying (impact on revenue) that product. Important thing is to note that Banking industry mostly deals with implicit data and they have almost negligible amount of explicit data.

Most of the research paper presented the implementation of recommendation system on ratings dataset (Explicit dataset) but this research proposes the approach of recommendation system on implicit dataset which is cheaper to collect and also it is accurate in predicting the user's behaviour. But main problem with banking implicit dataset is that it only contains the purchased products record in the binary term (0 or 1 if user purchased any product then it's value stored as 1 else 0). And with this much limited amount of information about the product, it is very difficult to understand the user's feeling about the product.

To overcome this problem, this research presents an approach to transform binary details into confidence number which means higher the confidence number better would be the probability of buying or purchasing the product. This research presents a mathematical approach to convert the binary information to a confidence number using a scoring approach. This scoring approach is developed by analyzing global and local feature of the product.

1.2 Research Problem

At present time collecting user's feedback explicitly is very difficult, expensive, time consuming and inaccurate as well hence recommendation product developed on explicit feedback (in order to recommend items or users taste) is not performing well. So, studying about user's purchase pattern and user's activity on the purchased product becomes essential (as it is cost effective as well) to understand users view about the product. This way to collect users feedback on the purchased product is known as implicit feedback wherein user explicitly does not give their opinion about the product.

Implicit feedback data in financial institutions mostly contain binary values which is different from other domains i.e. if someone is listening any song then this implicit data can be number of times user have played that song and by capturing this value we can understand users interest in that song but this situation is not the same for banking products.

In response to this problem, this research proposes to develop top-n recommendation system on implicit feedback data by using a scoring technique to understand user's confidence in purchasing any product.

1.3 Aim

To Propose a Recommendation System algorithm for recommending top-n banking product to existing customer of the bank/Financial Institution (FI) based on implicit dataset (purchase history of the customer with the bank) available with the bank/FI.

1.4 Objective

Following are the objective of this research:

- To propose a scoring technique which can help in understanding user's confidence on performed action using implicit data (binary in nature).
- To propose approach of cross selling of products to existing customer.
- To propose Top-n performing products among existing customer.

1.5 Scope the study

In this study, user's buying pattern of the banking product along with their demographic details are analyzed to propose recommendation system algorithm.

In this research data of last 3 years of one financial institution are analyzed which consists of 150,000 data points.

This study is conducted on data of Financial Institution which have branches only in one country and hence it is limited to FIs which have customers mostly in one country.

This study is limited to implicit feedback data (binary in nature).

1.6 Significance of the study

As in current scenario, almost everyone is using Financial products of different FIs/banks.
it's difficult to find that any particular customer is using more than three products of same FI/Bank though the same customer is using more than three financial products at the same time.
Important thing here is to note that almost every FI/bank have similar services or Products but the way it is catered to their audience is different and this is the main reason of attracting customers of their competitive FI/Bank.
The main reason of less cross selling among the FIs/Bank's existing customer, is lack of understating their customers taste or in other words, we can say that they are unable to serve personalized services to their customer.
Understanding the taste of customer is very important in order to get the attention from the customer because products offered to salaried, student and senior citizen will get different response. For example, if any long-term deposit product is offered to working Professional and Senior citizen both without understanding their taste then probability of buying the long-term deposit product by senior citizen will be very less because at this stage they don't really need long term deposit products.
Banks/FIs are not using recommendation system as much as e-commerce and media streaming companies are using. FIs business is different and their data collection is also different than media streaming and ecommerce companies. FIs/Bank deals more with implicit data and to be precise its more in binary nature and hence leveraging the importance of implicit data with Recommender system becomes crucial for providing personalized services to the customers.
Implicit feedback has following two kinds of data:

- Recording count of action on product in binary form i.e. either 0 or 1.
- Recording count of action on product in non-binary form (base 10)

This research will benefit to the FIs which are recording the action on product in binary form.
This research will benefit the FIs/Bank in cross selling their products.

1.7 Importance of Recommendation System in Banking Products

With the advancement in technology, banking is very much accessible through mobile and internet. Due to this it's very important to understand the taste of customer and cater them personalized services time to time to their existing customers.

As Doorstep banking is adopted by almost all the Financial Institutions (FI), hence customer need not to visit the branch for buying the Products which fulfill their requirement. In place of this process, Bank representative visits the customers address and collect necessary document and handover the product as per their request.

With the help of Recommendation system, Financial institutions not only can attract new customers but also, they can sell their product to existing customer (cross selling).

Selling product to new customer is costlier as Bank/FI has to collect the data from different channels but **selling the product to existing customer is cheaper** as data of existing customer is already available in the Banks/FI database. Probability of knowing existing customers taste is higher than knowing the taste of new customer as Bank/FI is aware of the purchasing power, demographic and credit bureau details of the existing customer.

1.8 Structure of the Study

Structure of the thesis is as follows:

Chapter 2, is structured in following three parts: First part contains definition related to recommendation system development, types of recommendation system, importance of recommendation system, real time uses of recommendation system. Second part of this chapter explains types of rating data set and third and final part explains related studies about recommendation system and challenges involved in the development of recommendation system

Chapter 3 of this thesis elaborates the research methodology used in this research. This chapter presents the overall thesis methodology right from data selection to model deployment. It also explains the analytical framework used in this research. This chapter also explains the approach of converting implicit (binary form) data to a score and this score reflects the confidence of user for

purchased product. Last section of this chapter talks about different tools and libraries like Pandas, Numpy, Python, Implicit, scikit-learn, seaborn and matplotlib, used in this research.

Chapter 4 of this thesis talks about analysis part. In this chapter following things are discussed: Data set details, pre-processing of the data set, Exploratory data analysis (EDA), imputation techniques for outlier and NA value, Exploratory data analysis results, general difficulties encountered in this research.

Chapter 5 of this thesis talks about the design part involved in this thesis. In this chapter different aspect of design is discussed right from input, model mechanism, performance tuning and output. This chapter discusses about the pre-processing of the data (which is essential step to leverage the maximum result from the model), different models used in this research, parameter tuning of the model, and selecting the best model among all the models used for this solving the problem statement of this research.

Chapter 6 of this thesis talks about the results and evaluation of this study. In this chapter different metrices of evaluation of this study is explained and also provides the output of the result of each evaluation methodology. This chapter also explains about the online and offline evaluation techniques and also highlights the constraint involved in online evaluation of this study.

Chapter 7 of this thesis talks about the conclusion and future work. This chapter explains how aim and objective of this research is achieved and also highlights the contribution to knowledge.
In the last section, this chapter explains the areas which can be improved (by using advance techniques such as deep matrix factorization models and other mathematical form for developing scoring approach) and considered for future work.

Chapter 2 Literature Review

This chapter will focus on the studies/research which were done in the past and related to this research.

2.1 Recommendation System

Recommendation system simply means recommending user's taste based on past purchase of the user.
There are different types of models in recommendation system.

2.2 Basic Models of Recommendation System

Following are the basic models of Recommendation system:

- Collaborative Filtering
- Content Based Recommendation System
- Knowledge Based Recommendation System
- Utility Based Recommendation System
- Demographic Recommendation System
- Hybrid and Ensemble-Based Recommender System

2.2.1 Collaborative Filtering

Collaborative filtering recommends item/product based on users collaborative behaviour.
Basically, this method finds similar users and refer those items which were consumed by these similar users. Let's say A and B are similar user. A has consumed i1, i2, i3 and i7 items whereas B has consumed i1, i2, i4 and i5 and both A and B have rated these items.
So, this approach will recommend i4 and i5 items to user A if B's rating is Positive for these items. And this approach will recommend i3 and i7 items to user B if A's rating is Positive for these items.
Most of the models for Collaborative filtering focus on leveraging either user-item or item -

user correlations for the prediction process. The main challenge in designing collaborative filtering methods is that the rating matrix is sparse. Because most of the users have viewed only small fractions of the large universe of items/movies/content. As a result, most the ratings are unspecified.

Following two types of methods commonly used in collaborative filtering:

1. **Memory based methods**

   This method is also known as neighborhood – based collaborative filtering algorithm. In this method rating of user item combinations are predicted on the basis of their neighborhoods. These neighborhoods can be defined in one of the two ways:

   - **User - based collaborative filtering**

     In this approach, like-minded users of target user say A are used in order to make recommendation for user A. This approach first determines users which are similar to the target user A and recommends rating for the unobserved ratings of A by computing weighted averages of this peer group (Users similar to user A).

   - **Item - based collaborative filtering**

     In this approach, to predict the rating of target item I by user A, set of items are determined that are similar to target item I and rated by user A. The ratings in item set I, are used to predict whether the user A will like item I.

2. **Model Based Methods**

   This approach builds a model based on the ratings data set. Usually some information is extracted from the dataset and use that as a "model" to make recommendations without using the complete dataset every time like standard machine learning model development approach. This approach offers both speed and scalability.

   Some examples of such model-based methods include decision trees, Bayesian methods and **Latent factor models.**

   Methods of these family e.g. latent factor models, have a good coverage on sparse matrix data as well.

2.2.2 Content - Based Recommendation System

In content-based recommendation systems, the descriptive attributes of items are used to make recommendations.

Consider a situation where User A has rated the product “Amazon pay Credit card” highly, but we do not have access to the rating of other users. Therefore, it’s not possible to apply collaborative filtering in this problem. However, the Product description of product “Amazon Pay credit card” put this in the same bucket as other Credit product like Loan, over draft and cash credit etc. In such cases, these products can be recommended to user A.

2.2.3 Knowledge - Based Recommendation System

This kind of approach is useful in the context of items that are not purchased frequently e.g.: Banking products. Usually a recommender system falls under the category of Knowledge – based recommendation system if it makes recommendation not on user’s ratings history but on specific demand made by the user. e.g.: If any user is looking for a house then it has specific requirement like number of bathroom, floor space and number of bed room. In this case if recommendation system is asking users input to provide these details and not only just recommending based on historical rating data then this kind of approach is referred as Knowledge – based recommendation system.

2.2.4 Utility - Based Recommendation System

In utility – based recommendation system, a utility function is defined on the product features in order to compute the probability of a user liking the item.

The main challenge in utility - based methods is defining an appropriate utility functions for the user at hand.

2.2.5 Demographic Recommendation System

In Demographic recommendation system, the demographic information about the user is leveraged to learn classifiers that can map specific demographics to rating or buying properties.

2.2.6 Hybrid and Ensemble - Based Recommendation System

Most recommender system nowadays uses Hybrid approach. This boost the prediction/recommendation power of the recommendation system. Combining Collaborative filtering, content based filtering and demographic recommendation system is an example of hybrid recommendation system. These approaches combined to overcome the issues faced by individual approaches.

2.3 Rating Data type

Usually recommendation is done on following two types of data: Explicit data and Implicit data.
In Explicit type of rating user usually provide the feedback explicit. It can be numeric or textual feedback. Numerical type feedback can be on the scale of 10 or 5 or on any other scale. On other hand, in implicit feedback user does not provide the feedback explicitly. Feedback here can be extracted from any of the following ways: user's purchase pattern, number of times user has performed any particular activity like Number of times any particular song is played or number of times any web page is clicked etc.

2.3.1 Explicit Data

Explicit data is a data where we have some sort of ratings like rating on 1 to 5 scales or 1 to 10 scales. In this type of data, we know how much user has liked or disliked any particular content or item.
This Data is tough to get as user has to spend some time in rating the data which most of the time does not happens.
Usually this type of rating is almost not present in banking industry. When any customer buys any product then there is no way to rate the product because sometime customer buys the product because of their needs like Loans or Salary account and banking products are one-time purchase products.
As our research is based on Implicit data hence we will be talking more on the implicit data and models / algorithm which deals with implicit data or perform better on implicit data.

2.3.2 Implicit Data

This data is gathered from the user's behavior, with no ratings. Following data falls under this category: dataset that contains what items was purchased by user, how many times a user has clicked any article or module, how many times a particular movie was watched by customer, how many times a song was played by the user, how long user was on any page/movie/song.
But in this type of data it's hard to understand the emotions (Positive or Negative) of the user.
Confidence is an essential parameter in order to understand the user's interest in the item or product.
Let's understand this using an example: If a user has played a song one time then it's hard to say if he has liked that song or not. In other words, Confidence of saying likes or dislikes in this case is very low but if user has played that song 500 times then confidence increases and we can say with very high confidence that user likes the song.
This type of data is not that much costlier to collect as explicit data is. Nowadays most of the companies are relying on the implicit data.

2.4 Related studies

Following are the studies which are related to this research:

2.4.1 Preference- Confidence approach on implicit data

In 2008, Y. Hu, Y. Koren and C. Volinsky et al [1] published a paper on "Collaborative Filtering for Implicit Feedback Datasets". In this paper, they mentioned that implicit user observations should be transformed into two paired magnitudes: Preferences and confidence levels. In other words, for each user - item pair, they derived from the input data an estimate to whether the user would like or dislike the item("Preference") and couple this estimate with a confidence level. They provided a latent factor algorithm that directly addresses the preference - confidence paradigm. In this solution they merged the preference (p) for an item with the confidence (c) we have for that preference.

If a user u consumed item i then we have an indication that u likes item i ($p_{ui} = 1$). On the other hand, if u never consumed an item i then we believe no preference ($p_{ui} = 0$). However, our thinking is very much related to confidence levels. p_{ui} values are defined by binarizing rui values:

$$p_{ui} = \begin{cases} 1, & r_{ui} > 0 \\ 0, & r_{ui} = 0 \end{cases}$$

In general, as rui grows, we have a stronger indication that user indeed likes the item. Consequently, they introduced a set of variables cui, which measures confidence in observing p_{ui}. They proposed this in below equation:

$$c_{ui} = 1 + \alpha\, r_{ui}$$

They found α = 40 to produce good result in their experiment. Here, confidence value can be calculated by solving the magnitude of r. Value of r will increase as many times user plays any song or view or click any item/article. Rate of which confidence increases is set through a linear scaling factor α. 1 is added so we have a minimal confidence even if $\boldsymbol{\alpha} * \boldsymbol{r}$ equals zero. The aim here is to find the vector for each user (x_U) and item (y_i) in feature dimensions which means we aim to minimize the following cost function:

$$\min_{x*,y*} \sum_{r_{u,i} \text{ is known}} (r_{ui} - x^T{}_u y_i)^2 + \lambda(\| x_u \|^2 + \| y_i \|^2)$$

In this paper, Global minimum can be calculated by solving user factor and item factor. Perform differentiation on the above equation which gives below equation and this can be used to minimize the loss of our users:

$$x_U = (Y^T C^u + \lambda I)^{-1} Y^T C^u p(u)$$

And this for minimizing it for our items:

$$y_i = \left(X^T C^i + \lambda I\right)^{-1} X^T C^i p(i)$$

Product of ***Y-transpose***, ***Cu*** and ***Y*** can be split as mentioned below:

$$Y^T C^u Y = Y^T Y + Y^T (C_u - I) Y$$

We have got Y^TY and X^TX which is independent of ***u*** and ***i*** which means we can compute it and calculation can be made much less intensive. So, following are the equation of final user and item:

$$x_U = (Y^TY + Y^T(C^u - I)Y + \lambda I)^{-1}Y^TC^up(u)$$
$$y_i = \left(X^TX + X^T\left(C^i - I\right)X + \lambda I\right)^{-1}X^TC^ip(i)$$

- X and Y: Our randomly initialized user and item matrices. These will get alternatingly updated.
- Cu and Ci: Our confidence values.
- λ: Regularizer to reduce overfitting (we're using 0.1).
- p(u) and p(i): Binary value of user's preference. Its value is one or more than one if we know that and it would be zero if we don't.
- I (eye): The identity matrix. Identity matrix which is a Square matrix with ones on the diagonal and zeros everywhere else.

2.4.2 Regularization

There is always chances of overfitting while training the data set. Overfitting means your model Is performing very well on training data while it is failing to perform well on test data as compared to its performance on train data set. To overcome the problem of overfitting regularization is used. Regularization is nothing but the addition of coefficient term in the cost function. This term is added to penalize the complex learning if by learning this model is moving towards overfitting. There are two types of regularization: L1 and L2.

- **L1 or LASSO Regularization**
  In this type of regularization, the absolute value of coefficient is added in the cost function.
- **L2 or Ridge Regularization**
  In this type of Regularization, the square of the coefficient is added in the cost function.

In short, regularization helps model in learning the trend and avoid learning noise in the data set.

2.4.3 Web application-based recommendation approach

In 2015, Adebayo A O, Agbola I S, Ayangbade A O, Obajimi O O et al [2] published a paper on Bank Products Recommender, the aim of this paper was to develop a web application which presents professional consultation about bank products to current and future customer of the bank.

2.4.4 Recommendation for Bon card customer on explicit data

In 2018, Abdorreza and Martin et al [3] published a paper on "Presenting Bank Service Recommendation for Bon Card Customers". In this paper, they presented an architecture in banking area for recommending the specific POS for available customers with the help of powerful approach called Singular value decomposition. In this study, the proposed approach was based on SVD method for prediction of rating values and recommend service to customer for purchasing. This research used the cumulative energy of PCA to estimate the optimal dimension for decomposition and creation of low-rank matrix. This research was performed on Bon Card dataset of Iranian private sector bank.

2.4.5 Intelligent approach for attracting churning customers using PSO and K-means

In 2016, Shafiei Gol, Elham, Ahmadi, abbas, Mohebi, Azadeh et al [6] published a paper on Intelligent approach for attracting churning customers in banking industry. In this paper, they focused on churning customer and presented banking services using customized recommendation system (RS) based on collaborative filtering to make them come back again. This customized RS uses particle swarm optimization (PSO) and K-means clustering.

2.4.6 Personalization of Banking Services using Memory- based Collaborative Filtering

In 2013, Himan Abdollahpouri, Alireza Abdollahpouri et al [7] published a paper on "An Approach for Personalization of Banking Services in Multi-channel Environment Using Memory- based Collaborative Filtering" In this paper, they have presented a design for implementation of

service recommendation in a multi-channel infrastructure with the help of a strong method called memory-based collaborative filtering. This approach has the ability to process the information which is coming into the system from different banking channels according to the recent activity done by the customer in the past and services which are most popular among the customers.

2.5 Matrix Factorization

Matrix factorization is very widely used in the recommendation system and it became popular after Netflix ten-million-dollar challenge. Matrix factorization usually helps in discovering latent feature in the user-item matrix. The biggest problem in recommendation system data set is sparsity which means there are lot of item where ratings are not provided by user. If we find those hidden features by which we can understand how user rate the movie/product then we can fill those unfilled ratings. This challenge in identifying the hidden feature (also known as latent feature) can be solved effectively using matrix factorization. Usually in matrix factorization, latent features are less than the total number of feature and total number of data points present in the data set. This seems logical because of latest feature comes out to be equal to the number of features present in the data set then there is no use of identifying the latent feature. Singular value decomposition (SVD) and singular value decomposition plus (SVD++) are also used widely to unfold latent features in the model.

2.6 Challenges in recommendation system

There are following challenges in recommendation system:

2.6.1 Sparsity

Sparsity is one of the very common problem in recommendation system as it's very rare that user rate all the products available in the list. This is very common type of problem with explicit data but in case of implicit data, it will be either 0 or 1 (binary case) or will be a finite number or 0 (base 10). Feature reduction and data point reduction techniques are usually used to remove the

unnecessary user and item which are not contributing in pattern learning and this helps in reducing sparsity in user-item matrix.

2.6.2 Cold-Start Problem

At initial point of any business case, usually we get relatively very less number of rating/feedback. Based on this available rating in hand, it is very difficult to understand the taste of large user space (a core concept of Collaborative filtering). There are lot of techniques to overcome this problem and one of them is content based approach wherein we use content of the product to overcome this problem.

2.6.3 One Time Need

There are many scenarios wherein user purchase/buy things without their interest or in some cases its mainly a one-time purchase i.e. salary account.
In case of salary account, user usually buy the account without any interested. It just a one-time need and sometime user don't get much option to buy this product because of unavailability of other FIs salary account product (a very common case of bulk employee onboarding).
Second scenario is junior account, wherein user does not have any option for other FI product because of their age (less than 10 years).
So, recommending product based on above pattern would not be useful as user is not going to buy more than 1 salary account.

2.7 Summary

Most of the application developed or research done on the FI system so far, utilized explicit data and hence recommendation system is not as successful as it is in ecommerce or media streaming domain.
Recommendation system usually built on following two kinds of dataset: Implicit and Explicit.
Explicit data is costlier to collect and sometime user provide the feedback just to bypass the feedback section hence in most cases it does not represent user's true behaviour. On the hand,

Implicit feedback does not require any additional effort to capture the user's feedback. It records feedback of user based on user's action performed on the system. Implicit feedback in most cases truly represent the behaviour of the user on the system. If user is listening any song multiple times then confidence of liking that song increases and using that kind of song a pattern can be decoded by the system for that user in order to recommend other songs of this type/category.

In this research, we have utilized preference-confidence approach proposed by Y. Hu, Y. Koren and C. Volinsky [1] in 2008 to develop a recommendation algorithm for implicit data which is in binary form.

Understanding user's confidence for any activity in binary form is very difficult hence author has used a scoring technique to generate a score which is similar to number of times user has performed an action on any activity.

Chapter 3 Research Methodology

3.1 Introduction

This chapter focuses on research methodology of this dissertation. In other words, in this chapter author has explained the research strategy, data collection, data analysis approach, data pre-processing, data transformation, proposed method, modelling and evaluation.

3.2 Research Methodology

This section will cover research methodologies of this dissertation. Following are the various stages of this research:

3.2.1 Data Selection

As this research is about recommendation system and that too based on implicit feedback kind of data hence it's essential that researcher select/collect the data which have action of customer in implicit form and along with action of customer their demographic and financial details should also be present. Demographic and financial details will help researcher in understanding more about their customer.

Implicit in nature here refers that user action should be recorded in implicit manner e.g. Product purchase, number of time customer has done the transaction, number of times customer has visited the branch etc.

This research belongs to Financial domain and collecting data from FIs are very difficult due to its confidentiality.

For our research we have picked the data from Kaggle which is an online community of data science professionals/practitioners.

This Kaggle data set is uploaded by Santander Bank which is a well-known Spanish bank.

Disclaimer: This is to note that data set does not include any real Santander Spain's customer, and thus it is not representative of Spain's customer base.

3.2.2 Preprocessing of the Data

Data Preprocessing is very important to understand the pattern in the data. Data Preprocessing steps depends on the data in hand. We have done following activities as part of the preprocessing steps:

- Missing values identification and Imputation
- Outlier detection
- Feature renaming from Spanish to English language
- Feature exclusion based on variance in the feature

3.2.3 Data Transformation

Dataset selected for this research is in wide format. Researcher has converted the wide format data into long format (narrow format). In long format, a new row (data point) is inserted for purchased product for every user and all product features is parked under one feature. Suppose there are 10 products then these 10 Products will be represented by one umbrella say Products and if user A owns three products out of these ten products e.g. loan, pension and credit card then in long format, three rows will be inserted for user A as depicted in figure 1.

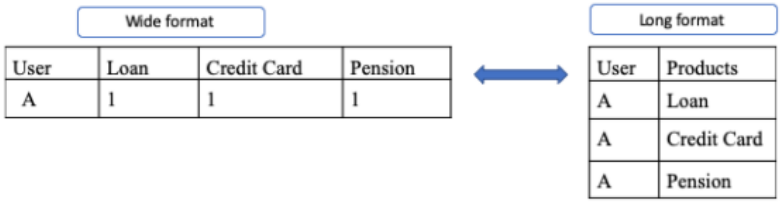

Figure 1 Long vs wide format

3.2.4 Scoring Approach

As in this dataset user's action is captured in binary form hence it becomes essential that researcher unfold the strength of user's action on each product and to achieve this following scoring approach is proposed:

1. Normalize the data
2. Calculated the importance using global and local feature of the product.

3. Assign the weight to each feature according to their importance.

Score = ß1*[X1] + ß2*[X2] +……+ ßn*[Xn]

Where ß1, ß2,….., ßn are weightage and X1, X2,…..,Xn are features.

We had tried hit and trial method and for our setup we have got following values of coefficients for following feature:

Coefficients:

ß1=0.1, ß2=0.15, ß3=0.25, ß4=0.2, ß5=0.1, ß6=0.1, ß7=0.05, ß8=0.05

Feature:
Gross_income_household, Deceased_Index, Cust_resi, Cust_Seniority_Months, channel_value, Segment, age, ind.

3.2.5 Modeling

In this research, Model based collaborative filtering approach is used to recommend the product to customer. Researcher has performed experiment with Alternating Least Squares (ALS) model and found this model suitable in this research, for model based collaborative filtering approach.

Alternating Least Squares (ALS) with confidence paradigm as mentioned in Y. Hu, Y. Koren and C. Volinsky [1] paper is used for this research. This implementation of Y. Hu, Y. Koren and C. Volinsky [1] paper ("Collaborative Filtering for Implicit Feedback Datasets") is already done in the Implicit package of Python hence we have utilized Implicit package to achieve best fit model for this research's dataset.

Researcher achieved convergence in 50^{th} iteration with regularization value 0.1 and factor value 20.

We have divided the dataset into 70:30 ratio i.e. 70% for training the model and 30% data we have used to test the model.

3.2.6 Evaluation

We have opted offline evaluation methodology as we do not have access of real time systems.
We have used hold out strategy to evaluate the model performance on different performance metrices.
Following metrices are used to evaluate the model:

3.2.6.1 Precision

Precision in classifications means the ratio of correct positive and predicted positive but in case of recommendation system this definition gets twisted slightly. In classification problem, prediction is either true of false but in case of recommendation system we define this in terms of relevant recommendation (which means how relevant was the recommendation for user) and total recommended products.
So, precision in the recommendation system is defined as ratio of total number of recommendation that are relevant and total number of products recommended.

$$Precision = \frac{\text{Total number of recommended products that are relevant}}{\text{Total number of recommended products}}$$

Mean average precision @k is derived using above formula and precision is the key term for this metric.

3.2.6.2 Mean average precision at k (MAP@k)

MAP@k stands for mean average precision at k and k stands for number of recommended products (if we are developing top-6 recommendation system.
MAP@k can be understood by following example. In this metric, order of recommendation matters.
Assume there are 24 products and target of this recommendation system is to recommends top-10 products for each user and recommendation order is as defined in following table.

Products row represents the products recommended to the user and it is mentioned in the recommended order which means credit card is recommended as first product and debit card is recommended as the second product and checking product is recommended as the last product. Relevant row here indicates whether recommended product is relevant to the user or not. If its value is 1 then it means it is relevant and 0 means it is not relevant product for the user. Ranking row represent the order of recommendation of the products.

User 1:

Ranking	1	2	3	4	5	6	7	8	9	10
Products	Credit card	Debit card	Home loan	Pension	Junior accounts	Savings accounts	Retirement accounts	Certificates of deposit	Mortgage	Checking accounts
Relevant	1	1	0	0	0	1	1	1	0	1
Recall	1/6=0.166	2/6=0.33	0	0	0	3/6=0.5	4/6=0.66	5/6=0.833	0	6/6=1
Precision	1	2/2=1	0	0	0	3/6=0.5	4/7=0.57	5/8=0.62	0	6/10=0.6

In this scenario, ten products were recommended and only six products are relevant for the user. And precision in this case can be calculated for each product as follows:

Precision for first recommended product = 1/1 =1

Precision for second recommended product = 2/2 =1

Precision for third recommended product = 0

Precision for fourth recommended product = 0

Precision for fifth recommended product = 0

Precision for sixth recommended product = 3/6 = 0.5

Precision for seventh recommended product = 4/7 = 0.57

Precision for eighth recommended product = 5/8 = 0.62

Precision for ninth recommended product = 0

Precision for tenth recommended product = 6/10 = 0.6

So, average precision in the case will be = Precision of each product/ total relevant product

So, average precision in the case will be = (1+1+0+0+0+0.5+0.57+0.62+0+0.6)/6 = 0.715

Suppose for user 2, average precision is 0.61

Then, mean average precision (MAP) will be = (0.715+ 0.61)/2 = 0.6625

So, mean average precision (MAP) is sum of average precision for the recommendation of each user divided by number of user.

3.2.6.3 Recall

Recall in machine learning is also known as sensitivity that is true positive rate of the model.
Number of relevant product from all relevant product and this is calculated based on 10 recommended products i.e. recall@10.

$$Recall = \frac{\text{Total number of recommended products that are relevant}}{\text{Total number of relevant products}}$$

3.2.6.4 AUC-ROC

Area under the curve is a good estimator for understanding the stability of the model. More the value under the curve better the model would be in segregating the classes. For Perfect model it's value will be one.
Every user has its own recommendation hence we have used mean AUC to check the model performance.

3.3 Proposed Method

To achieve the aim and objective of this research, Collaborative filtering and Alternating least square (ALS) is proposed. Alternating least squares model is used to finding the similarities between the users. As similarity is the core concept of collaborative filtering hence we are calculating similarity between the users using Alternating least squares.
As researcher is keen in using notion of confidence in implicit data (binary form) hence researcher has gone with the approach which was presented by Y. Hu, Y. Koren and C. Volinsky [1] in their paper on "Collaborative Filtering for Implicit Feedback Datasets". In this approach notion of confidence is included along with Alternating least squares model (ALS) model.
Koren and C. Volinsky approach is implemented in Implicit package (this works well for implicit

data) but this package does not accept the data in binary form.

To overcome this situation, researcher have proposed a scoring technique which takes input in binary form and provides the output in base 10. This output (base 10 form) represents the strength of user's action (Product purchase in this case).

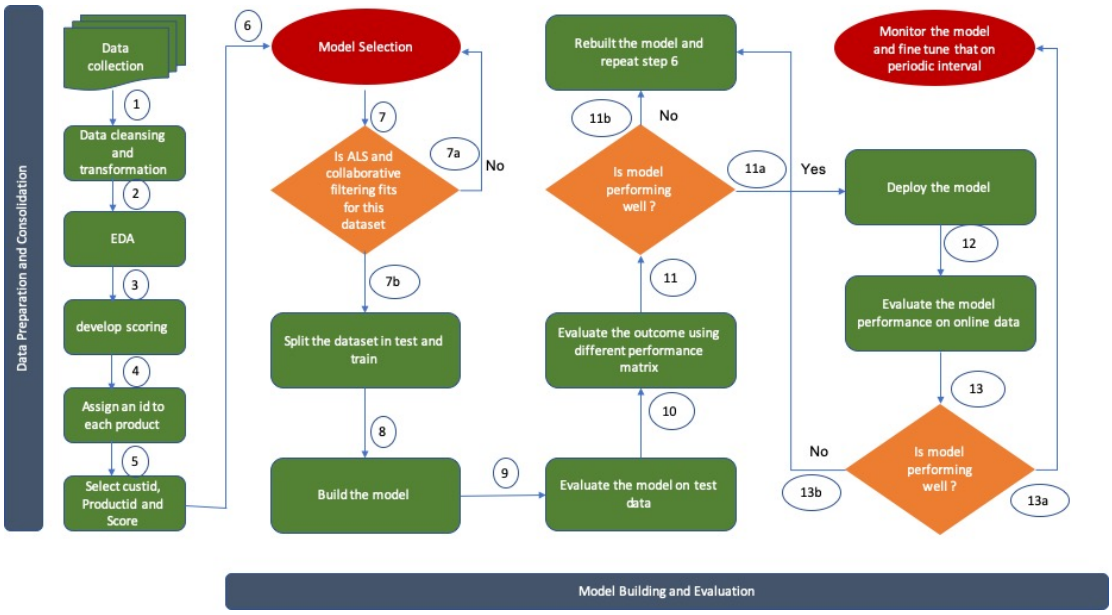

Figure 2 Process flow

Output of the proposed scoring technique is similar to the action which is recorded as number of times a user visits on any site or number of times any particular song is played by the same user. Matrix factorization technique is used to unfold the latent feature and this makes entire process computationally efficient.

- Using this scoring approach, score of each product for each customer is calculated.
- Each product is assigned with a product id (integer form).
- This score, product id and user id are passed to ALS algorithm of implicit package.
- Implicit package calculates their own similarity score to recommend the item.
- Implicit package internally calculates the dot product of user vector and item vector in order to calculate the similarity score.

3.3.1 Explanation of Process flow

Step 1-5 belongs to Data preparation and consolidation and step 6-13 belongs to model building and evaluation.

Data collection is done in step 1 and in step 2 collected data set is cleansed and transformed.

Post the step 2, Exploratory data analysis (EDA) is performed to understand the data set in step 3. Scoring is developed for each user on every product. This reflects the user's confidence on purchased product. This process is done in step 4. Post the step 4, a numeric id is generated for each product and this is done in step 5. In step 6, a subset of data set is collected for further processing. This subset contains following three features: user id, productid and score.

In step 7, model selection is done and model based collaborative filtering is selected as model for this research.

In step 7b, if selected model performs well on the dataset then it moved to step 8 else it is moved to step 7 by traversing through step 7a.

In step 8, data set is divided in test and train set. Train set is used for training the model while test set is used to test the model.

In step 9, Model building is done. Model is built by using Alternating least squares (ALS) and collaborative filtering. Model building is done with the help of implicit library of python.

In step 10 and 11, Model is evaluated on different metrices eg: mean value precision at k, recall and area under the curve- receiver operating characteristic curve.

In step 11a and 11b, decision is taken on the build model. If model performs well then it moves to next step else it would move to step 7 for model selection and development.

In step 12, model is deployed and in step 13, model's performance is monitored online and if it is doing fairly bad then process moves to step 7. If model performs well after deployment then also monitoring of the model is done continuously and based on the online feedback received, model performance tuning will be done. This step will be iterative and continuously feedback will be taken online and performance tuning will be done accordingly.

3.4 Analytical Framework

A framework is very essential and helpful to achieve business objective of any problem.
In this research, researcher has used CRISP-DM framework (Cross Industry Standard Process for Data Mining).
CRISP- DM framework involves following steps:

- Business Understanding
- Data Understanding
- Data Modelling
- Model Evaluation
- Model Deployment

The purpose of this framework was to perform the study in structural and efficient way.
Following are the explanation of stages of this framework.
First stage of this framework is business understanding which are aims and objective of this study.
Once the business understating is done, next stage comes and that is data understanding. This step helps in understanding about the data in hand for the analysis.
This step includes data related stuffs and all the relevant action on the data is performed which is required to make the data ready for model building
In next stage, model selection and model building are done and once model is build this framework allows to move to next stage that is model evaluation.
In model evaluation stage, model is evaluated on different metrices and if it performs well then it is passed to next step that is model deployment.
Model deployment is the last phase of this framework. Model deployment and monitoring comes in this stage.

3.5 Tools and Libraries

This section will cover details of the tools and technical stack covered in this research:

3.5.1 Python

Python 3.7 is used as coding language in this research. Jupyter notebook is used as IDE.

As Python have wide range of built in libraries for data analysis, scientific research and it's open source software hence researcher have chosen Python to go with in this research.
Python syntax are simple and easy to understand and also its very popular language in data science community. Python libraries like scikit-learn, Pandas, Numpy, seaborn and matplotlib makes job of any researcher very easy in performing research on analytics project.

3.5.2 Pandas

Pandas library is written in Python and it is widely used for manipulating and wrangling the data. Pandas build in libraries helps in reading and writing the dataset in just one line. It supports reading the file in many formats like csv, excel, tsv etc. Pandas built in functions support manipulation of data, data aggregation, data summarization, data structure conversion etc.

3.5.3 Numpy

Numpy is also one of the widely used open source library in Python.
It performs numerical computation very efficiently. It supports computation on multidimensional arrays and matrices.

3.5.4 Matplotlib

Matplotlib is one of the widely used open source libraries of Python for data visualization. It is 2D plotting library. Visualization graphs like boxplot, bar plot, pyplot, qqplot, histogram can be drawn by matplotlib using just few lines of commands.

3.5.5 Seaborn

Seaborn is one of the widely used open source library of Python for data visualization.
It is built on the top of Matplotlib and uses pandas in backend for data aggregation and numerical computation.

It offers many built in themes and colors. It is very efficient in performing visualization on categorical data.

3.5.6 Scikit-learn

Scikit-learn is an open source Machine learning library. It is written in Python.
It offers libraries of various machine learning algorithm like Regression, Classification, Clustering, Dimensionality Reduction, Support vector machine.
It has also built in function for model evaluation metrices like Area under the receiving operating characteristic curve (AUC-ROC curve), precision, recall etc.

3.5.7 Implicit Package

The mathematical foundation behind this package is based on the concept introduced in the paper "Collaborative Filtering for Implicit Feedback Datasets" presented by of Y. Hu, Y. Koren and C. Volinsky [1].
This package accepts the data in numeric form and does not accept categorical data. This package uses confidence- preference paradigm along with model based collaborative filtering to generate recommendation for the user.

3.6 Summary

In this chapter, researcher has explained methodologies and analytical framework used for this research.
Researcher has also explained the process followed in this research and explained the techniques used for this research like modeling.
End to end flow of this research is explained in this chapter right from data selection, data transformation, modeling, evaluation, visualization and approach to handle the implicit binary data (Scoring approach).
This chapter also provides the details of tools and libraries used in this research.

Chapter 4 Analysis

4.1 Introduction

This chapter focuses on analysis part of this research which includes analysis, hypothesis formed on the dataset, specific and general difficulties encountered in this research and exploratory data analysis results.

4.2 About the Dataset

This Dataset is collected from Kaggle.
This Dataset contains 13647309 data points and 48 features. Analysis is performed on 150000 dat a points and 48 features. This dataset contains following three levels of information: Demograph--ic, Type of Customer and Product purchase details. This dataset contains 24 products of the Sant ander bank.

4.3 Exploratory Data Analysis (EDA)

Exploratory data analysis refers to the initial investigation which we perform on the data to discover patterns in the data. Following are the few objectives of the EDA:

- Understand about the data
- Understand the distribution of the data
- Identifying collinearity in the variables
- Univariate analysis
- Bi-variate analysis
- Multi variate analysis
- Segmented Univariate analysis
- Identifying top features for our business objectives

4.3.1 Missing values identification and Imputation

In the data set of this research missing values are identified in 15 features out of 48. For Categorical variables, researcher have used mode to impute the missing values and median to impute continuous variables.

Researcher had found only 1.8% NA values in Payroll and Sex features hence instead of imputing those features, researcher have dropped rows that was containing NA values in these features.

4.3.2 Outlier detection

In the data set of this research outlier detection and treatment is performed.

Outliers are identified in the following features: Age (it has values less than 18 to 2 years also but that seems for junior account hence not treating this as outlier), Customer seniority in months and Gross income.

These outliers were identified using box plot and Quantile function. Sometime box plot does not give clear picture about the outlier hence quantile is also calculated before marking the data points as outlier.

In this research data points of any feature are treated as outlier if they satisfy following equation:

If data points of any feature are less than First quartile - 1.5 * IQR

Or data points of any feature are greater than Third quartile + 1.5 IQR

Where IQR is Q3 - Q1 and Q3 is 75 percentiles and Q1 is 25 percentiles.

4.3.3 Feature renaming from Spanish to English language

In the data set of this research, renaming of all the features are done from Spanish to its English form due to lack of understanding of Spanish language.

Original feature of tis data set is in Spanish however renaming of data points are not done.

4.3.4 Feature exclusion based on variance in the feature

In the data set of this research variance of the feature is checked and if there is no variance in the feature then it is dropped from the data set. In this research threshold value is considered as 98% which means if any feature is containing more than 98% values of same category/number then that feature is dropped from the data set.
Very few features had less variance in the data.

4.4 Exploratory data Analysis Results

Exploratory data analysis is primary step in understanding the data set and this helps in selecting important features and also helps in selecting the sample. This analysis helps in visualizing the problem statement against the available data set which means it helps in forming the hypothesis as well.
Results of Exploratory data analysis are discussed one by one as follows:

4.4.1 Product Popularity

There were few instances where age had played significant role in product purchase.
In this research, it has found that Junior account is more popular for age group which is less than or equal to 18 years. But on further analysis, especially when banking domain knowledge is applied on this then it has been found that junior account is only applicable for age group which are less than or equal to 18 years hence only this age group is present in the data for junior accounts and if it presents for other age group then it could be a data entry issue which can be imputed accordingly.
On further analysis, it has been found that Current account, direct debit, Particular account and Pensions are more popular product than others in the data set which is used in this research.
This has been also observed that current account has more inactive account as compared to other products.
Figure 3 helps in visualizing the product popularity by customer activity index.

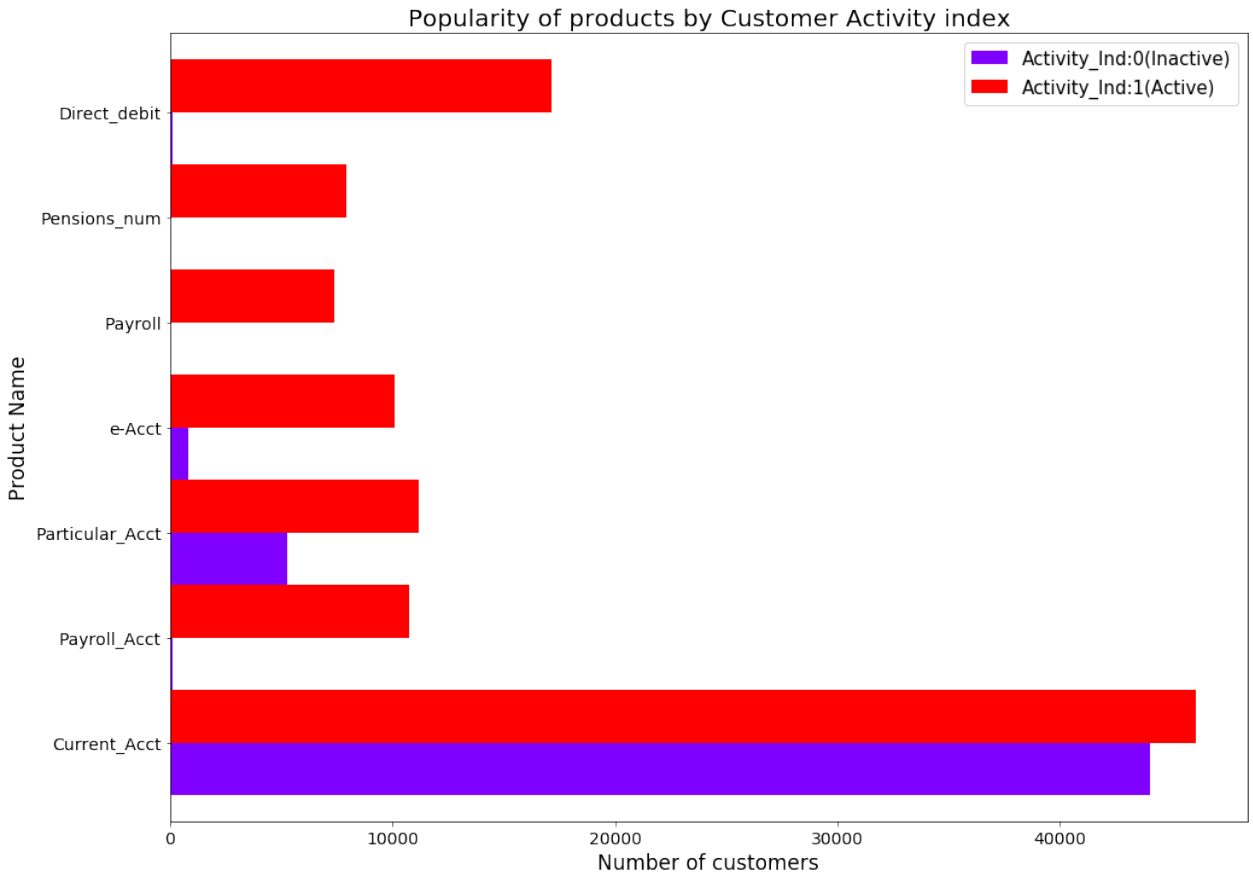

Figure 3 Popularity of Product by customer activity index

4.4.2 Gender wise Data distribution

In few countries like India gender wise dependency on product can be found e.g. Sukanya Samridhi Yojana. Under this scheme, only girls can purchase this account from bank/FI but in this dataset bias has not been found on gender basis hence it has been concluded that there is no bias in the distribution of the dataset gender wise. Following pictorial representation of the product on gender basis distribution depicts the same:

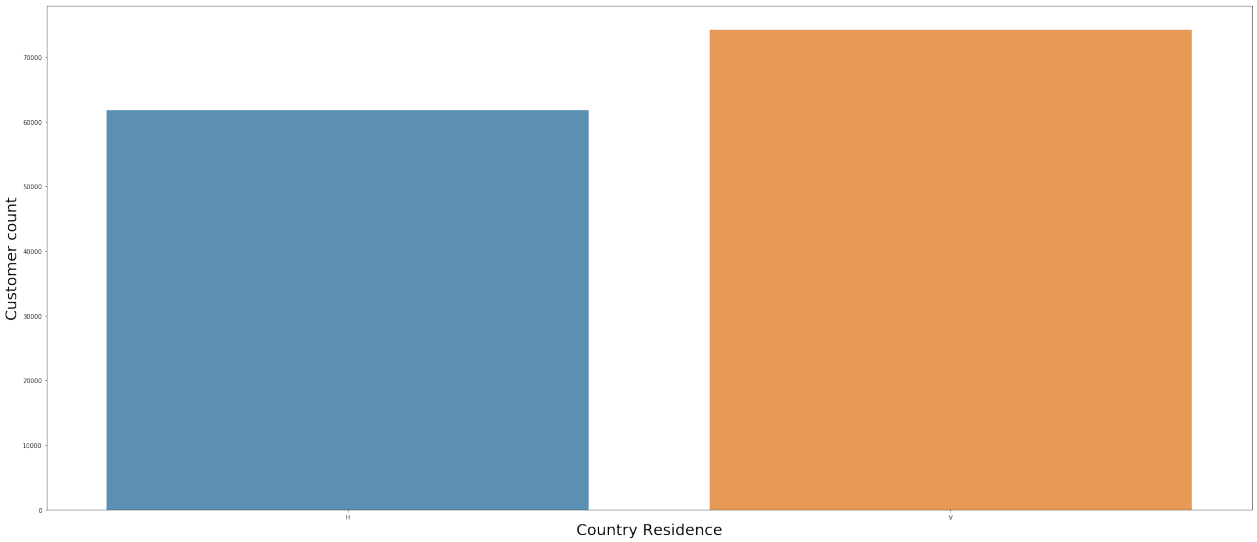

Figure 4 Gender wise data distribution

4.4.3 Country-wise Customer's Presence

In this analysis, it has been observed that Santander bank is doing their business globally but the more customer presence is only in Spain.99% Customer belongs to Spain.

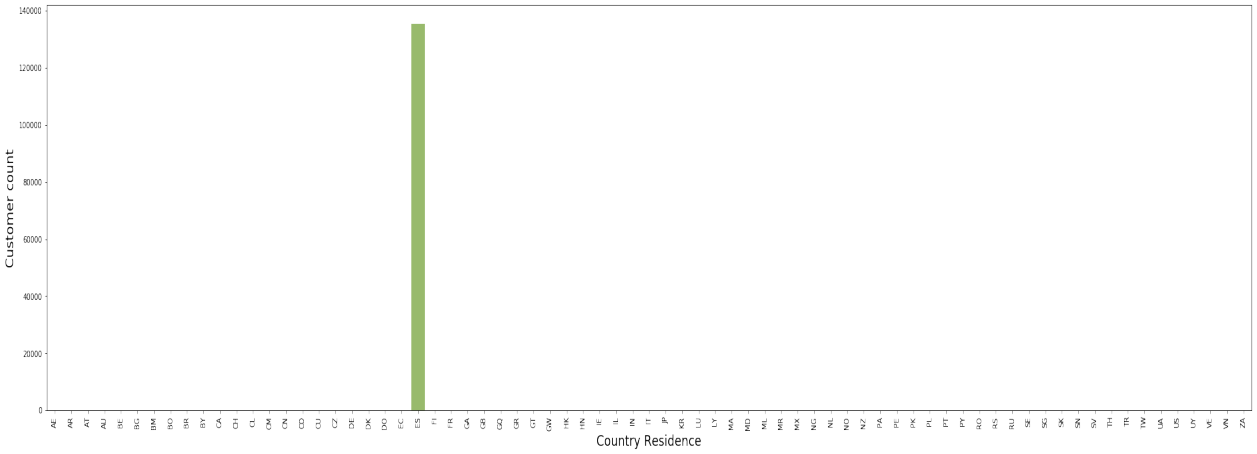

Figure 5 Customer's presence country-wise

4.4.4 Joining channel wise customer count

Following are the top 8 Joining channels customer count-wise:

KHE, KAT, KFC, KHQ, KFA, KHK, KHM, KHD

KHE and KAT channels are performing good in terms of customer enrollment and KHD is performing poor as compared to other channels.

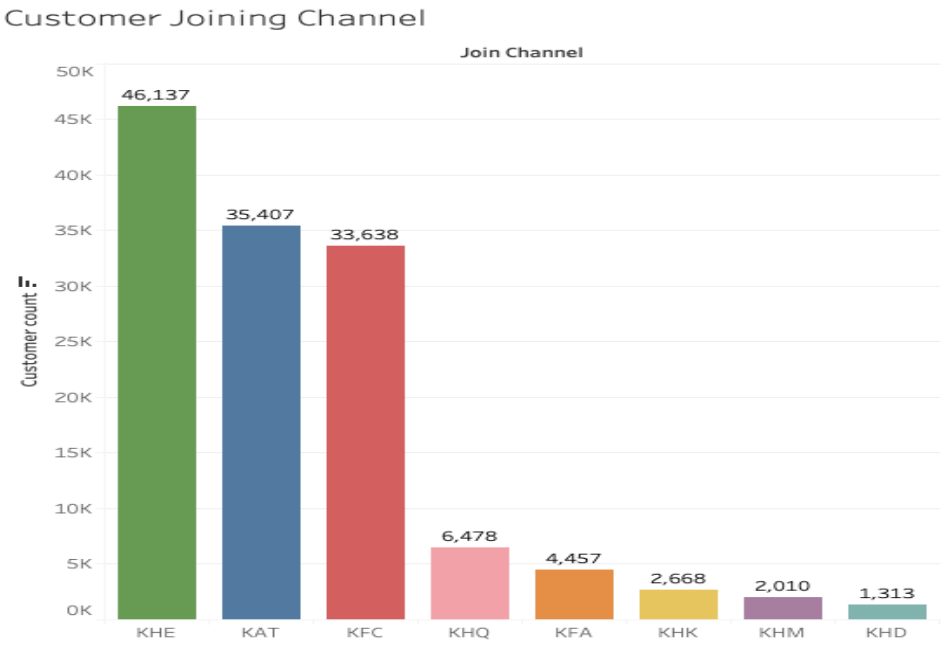

Figure 6 Joining channel popularity

4.4.5 Product popularity in different age group

As it can be seen in the figure 7 that Junior account is popular only in less than 18 age group While Current account is popular in all the age group except age less than 18 group.
Second most popular product among all groups are direct debit and particular account.

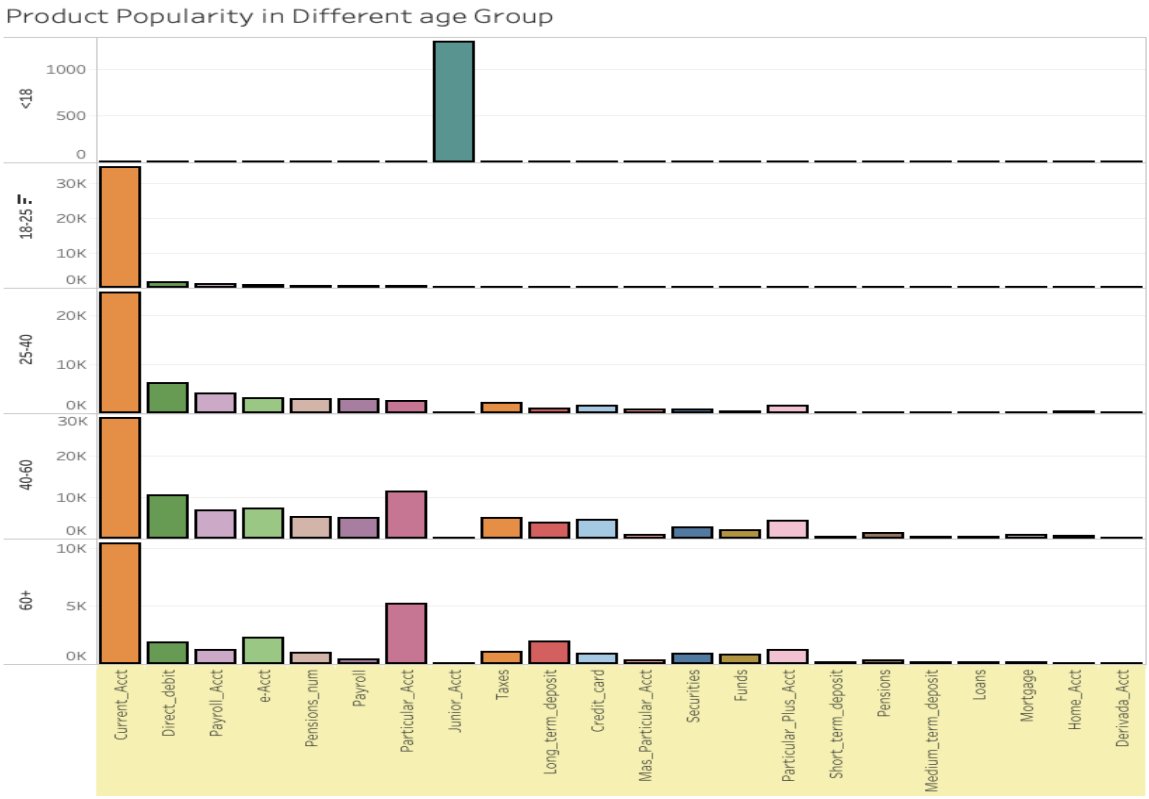

Figure 7 Product Popularity age group wise

4.4.6 Product popularity segment wise

As it has been seen in the analysis of this segment that current account is popular among all segments. While popularity of other products depends on segment of the customer.

Top three popular product among customer of university segment are:

- Current account
- Direct debit and
- Payroll account

Top three popular product among customer of particulars segment are:

- Current account
- Particular account and
- Direct debit

Top three popular product among customer of Top segment are:

- Current account
- Long term deposit account and
- e-account

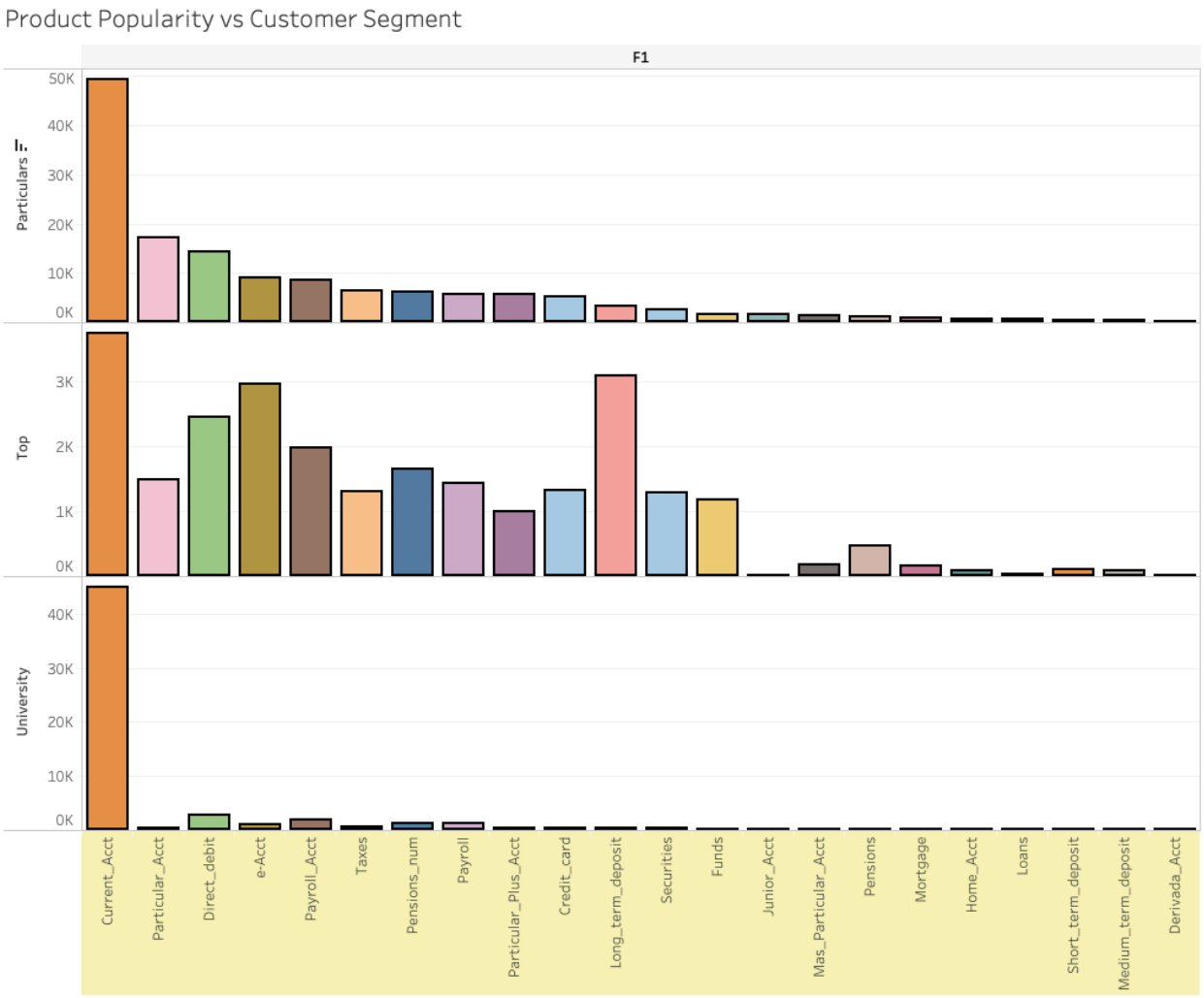

Figure 8 Product popularity segment wise

4.4.7 Product Popularity Gross income wise

As seen in the below figure, Current account is popular in every segment and second and third most popular product after current account are direct debit and particular accounts.

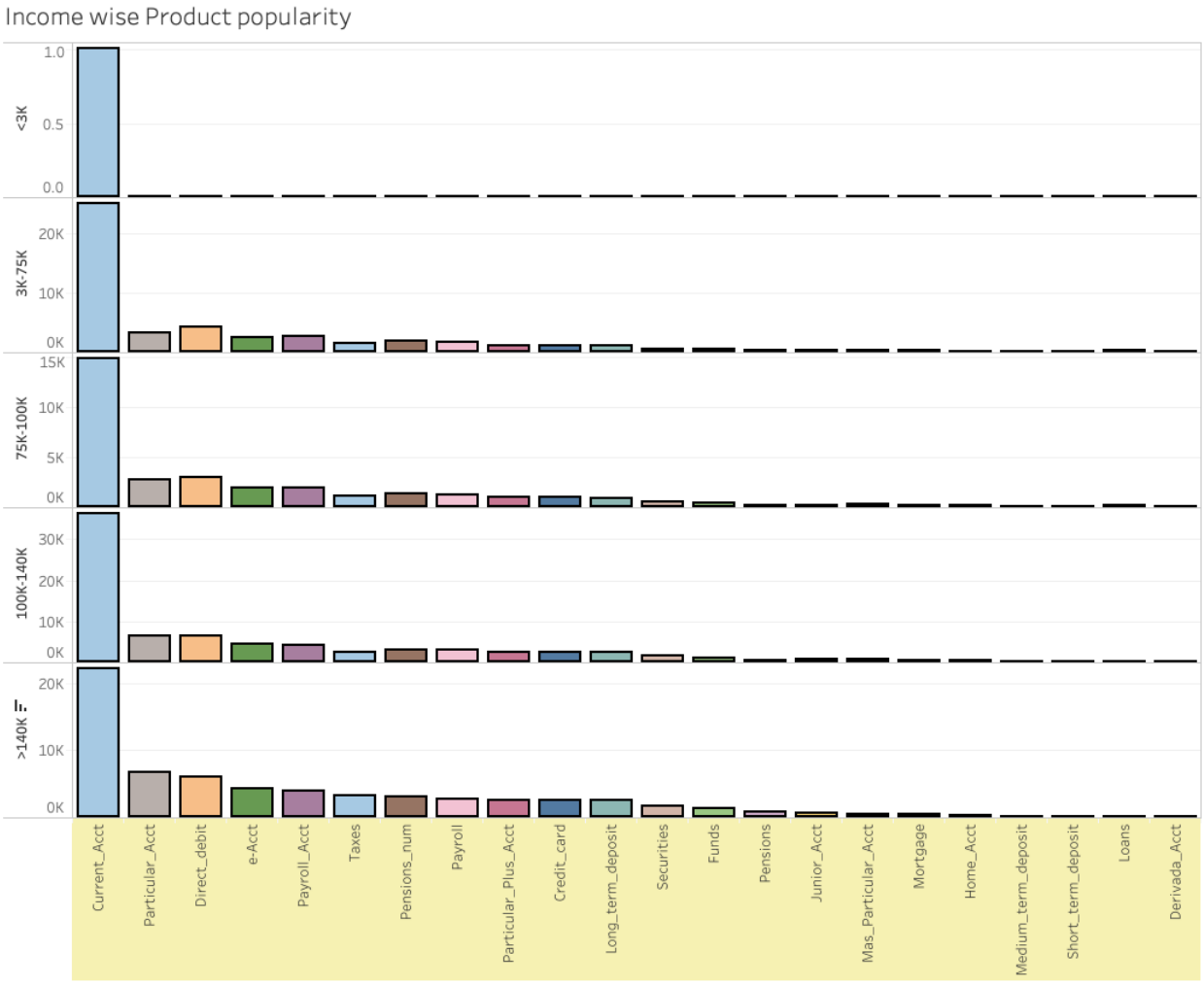

Figure 9 Product popularity Gross Income wise

4.5 Specific and general difficulties encountered

Following are the list of difficulties encountered during this research:

1. **Relevant Dataset**

   Very few data sets are available for banking domain and identifying the data for our problem statement was big challenge as this problem statement requires the purchasing history of the customer which is very difficult to get from the FIs/Bank.

2. **Relevant Literatures**

   As most of the literatures of recommendation system (especially for implicit dataset) belongs to either media streaming or ecommerce hence it was very difficult to find the literature which matches with this problem statement. Most of the research on recommendation system was done on rating-based data but the study of this research as per the problem statement of this thesis, research has to be done on implicit data that too on banking products which is very difficult part due to unavailability of literatures especially for banking domain and binary form of the data (implicit data set)

3. **Handling binary data**

   As this dataset is binary in nature hence it is difficult to understand that how strong is user's action on any of the purchased product. This can be understood by comparing this with other kind of implicit dataset eg: website click or any other media streaming app.

   In media streaming dataset, if user has watched any movie then that flag is one else it is recorded as zero (binary form) which is same as this research dataset but along with watched movies record, media streaming dataset also records how many times user has watched any movie. This feature wherein number of times user has watched any movie, helps in understanding how much user likes any movie.

   The dataset which is used in this research contains only binary data. With this dataset it is difficult to find out how much any user likes any product hence a scoring approach is used to understand user's confidence in purchasing any product.

4. **Model Evaluation**

It is one of the biggest hurdle to evaluate the recommendation system but we have used the concept of evaluation proposed in *Netflix prize competition* i.e. offline evaluation of recommendation system. This research has adopted offline model evaluation strategy to test efficiency of this model because it is difficult to get the access of banking/Financial infrastructure.

4.6 Summary

In this chapter, explanation about the data set is provided and preprocessing of the dataset including data transformation and data wrangling. Distribution of the data is explained feature wise and as well as segment-wise analysis is also performed on the data set.

Product popularity is analyzed on different-different parameters like age group, customer segment-wise, Gross income-wise and also analysis of the data distribution based on Gender and country are provided. Popularity of the joining channel based on joining channel of the customer is also analyzed and result presented to show the importance of the channel.

Chapter 5 Design

In this chapter, different aspect of design related to this research have been discussed right from model selection to model development.

5.1 Model Selection

As this dataset belongs to implicit data set hence selecting an algorithm which handles these kinds of datasets becomes an essential step.

In this study, user – item interaction is very important because at the end of the day recommendation system has to recommend only products to user and hence collaborative filtering will be a key idea here. Also, this dataset has good number of products and hence instead of interacting with each product, this study focuses on understanding taste space of the item.

One can understand the importance of taste space in different aspects like computational efficiency and in understanding the category of their product if any data set have large number of products.

This can be understood by following example: Suppose any person is in any shop and there are more than 500 products and only 5-10 products he/she has tried so far which means he/she understands only the taste of 5-10 products but when 500 products are presented then he/she would not be able to choose the products efficiently which means recommendation would be poor if products in this case are recommended with this large number.

In this case if super market understands the taste of user and bucket those 500 products in generic taste space then it would be easy for user to pick the product. Suppose these 500 products can be categorized as sweet, salty and so on. Now if user has to pick the product it would be easy for him/her. If he/she likes sweet taste then he/she would go to sweet bucket and pick the product of their choice. Advantage in this case is even if user has not tried any product so far then also he/she can pick the product because it suits his/her taste.

To understand the latent features, matrix factorization is used in this research.

In this study matrix factorization is used to reduce the dimensionality of the original matrix i.e. “all users by all items” into something much smaller that represents “all items by some taste dimensions” and “all users by some taste dimensions”.

Doing reduction in dimension makes the process computationally efficient.

Model based collaborative filtering is used in this research, along with matrix factorization approach in order to achieve good recommendations.
Alternating least squares (ALS) is used for best fit for this data set and find similarities between the data points and matrix factorization is used to find the latent features in the data set.

5.2 Principles of Model Selection

This section provides the details which were used for model selection. While selecting the model following parameters have been used:

- Occam's razor
- Overfitting
- Model complexity
- Bias-variance tradeoff
- Regularization

5.2.1 Occam's razor

Following are the highlights of Occam's razor:
A model should be as simple as necessary but no simpler. When you are not able to decide then choose the simpler model. Advantage of choosing simpler model is: less assumptions are required and thus less data for model training is required which makes the process computationally also.

5.2.2 Overfitting

Overfitting occurs when model performs well on training data but fails miserably on test data.
This happens when model memorize the data and don't learn the pattern present in the data set.

5.2.3 Model Complexity

Model complexity is an important parameter and simpler model is preferred over complex model in most of the cases.

Complexity can be understood by below example:

$Y = c + mx$ ..... (1)

$Y = v + kr^2$ ..... (2)

Equation 2 is complex and equation one is simpler as compared to the equation 1. Equation 2 is using 2-degree polynomial while equation one is 1-degree polynomial.

5.2.4 Regularization

Regularization helps in avoiding overfitting in the model. Usually a coefficient parameter is added in the cost function of the model to avoid the overfitting. This process of adding coefficient in the cost function is known as Regularization.

5.2.5 Bias-variance tradeoff

Bias measures the accuracy of the model which means how accurately the model can describe the task in hand. While variance measures the flexibility of the model which means how flexible is model when there is a change in the training data.

Bias and variance can't be achieved together because when bias increases then variance decreases and vice-versa.

Hence optimal value of both is required where model error is least.

5.3 System Design

Figure 10 system design, depicts the design used in this research.

System design is divided in the following five parts: Data, Pre-processing, Modelling, Parameter tuning and Recommendations (output).

Data is very important for training any machine learning algorithm. The data used in this research belongs to financial domain. This data contains the user's demographic details, product purchase history and their financial status as well as category (Bank's category eg: VIP, tier 2 and tier 3).

This data is collected as per the aims and objective of this study and passed to second step i.e. pre-processing. Pre-processing is most time-consuming stage in the design cycle of any machine learning algorithm development.

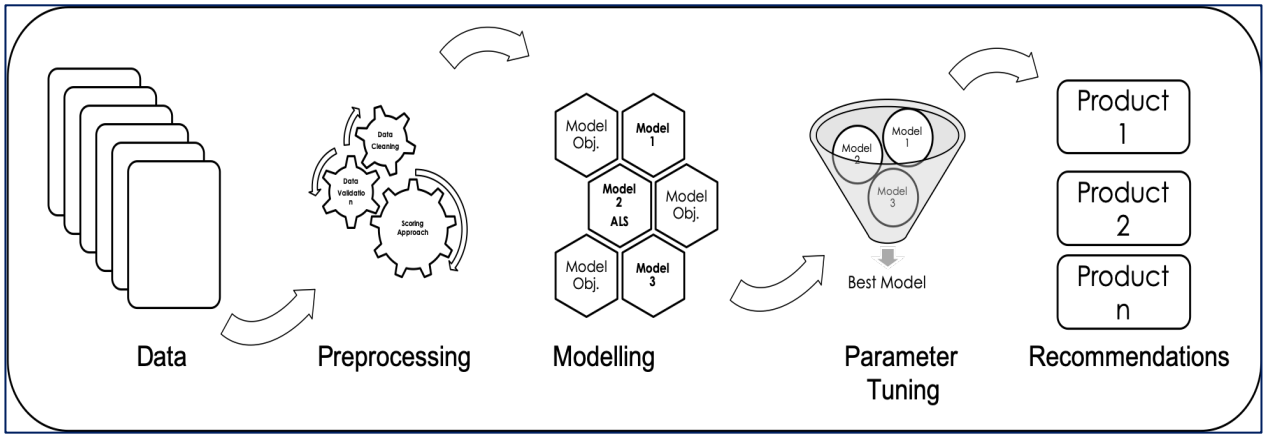

Figure 10 System Design

Pre-processing makes the data in the form which is acceptable by model (input of the model). It requires data cleaning, transformation, statistical test (to understand whether data meets the prerequisite of the model), deriving new feature (if needed), dimensionality reduction (if any), class balancing (if data is biased for any one class), sampling (if needed).

In this study, along with generic pre-processing process one more approach is used to meet the pre-requisite and that is scoring approach.

As the model which is used in this research takes numerical input (base 10) but most of the data in this data set belongs to binary form hence a scoring approach is used to convert binary form of purchase (product purchase) data into a number which reflects the confidence of user for purchased product. Once pre-processing of the data is done then process move to next step i.e. modelling.

In this study, different-different model is used to meet the business objective (Aims and objective of this research) and model based collaborative filtering was the one whose result was very promising among all other models. In model based collaborative filtering, this study finds Alternating least squares (ALS) model very satisfactory for this study. Alternating least squares (ALS) with collaborative filtering approach was able to address aims and objective of this research.

Post the model development stage, process moves to performance tuning stage. Performance tuning is done based on the output received by different-different evaluation metric e.g.: mean average precision, recall and Area under the curve – receiver operating characteristic (AUC-ROC). Hit and trial method is also used in this research to achieve the best value of alpha and in this experiment best value of alpha for our model is 0.0005.

Once this model is able to produce the result which meets the aim and objective of this research then only this process moves to next stage.

Final stage of this research is recommendation of the products and evaluation. Evaluation is very important stage in order to understand the quality of recommendation. Due to some constraint (access of financial environment or infrastructure), this research relies on offline evaluation and online evaluation was not done on this research.

Historical data is used to perform offline evaluation of the model output.

5.4 Summary

In this chapter, design aspects of this research are discussed. In this chapter, in this chapter principles of model selection like Occam's razor, Complexity, bias- variance tradeoff etc. are discussed. In this chapter end to end process flow of design of this research is also discussed.

Chapter 6 Results and Evaluation

6.1 Introduction

In this chapter, results and evaluation of this research has been covered. As most of the recommendation system is developed on explicit data set and as this research is done on implicit data set hence comparing the result of this research with the result of explicit data set based recommendation system is almost not feasible. However, we have used many metrices to validate and verify the research output.

This research is evaluated using offline evaluation methodology. Data set of this research is divided into following two parts: Train and Test set.

Model is trained on train set and evaluated on test set. Model has not seen the test set hence this set is treated as unseen data for the model evaluation.

This research is evaluated on the different evaluation matrices such as recall, Area under the curve – Receiver operating characteristic (AUC-ROC) and Mean average precision (MAP).

6.2 Model Output

Model is designed to recommend top-n products and in this research value of n is taken as 10.

Model internally generates a score for each product for each user and based on this score, model provides top-10 products for each user as its output.

6.3 Results

This research is evaluated on following metrics: Mean AUC vs Alpha, mean average precision, Mean recall, Area under the curve.

6.3.1 Mean AUC vs Alpha

As this thesis is performed on the data for more than one user hence it is essential to understand that what is the model performance for all users i.e. average performance of the model for all users hence mean is selected.

The rate of increase of cui is controlled by a constant alpha.

$$c_{ui} = 1 + \alpha\, r_{ui}$$

alpha for this research is found as 0.0005 where mean AUC was maximum i.e. 0.699. Figure 11 mean AUC vs alpha depicts the mean AUC and alpha relationship.

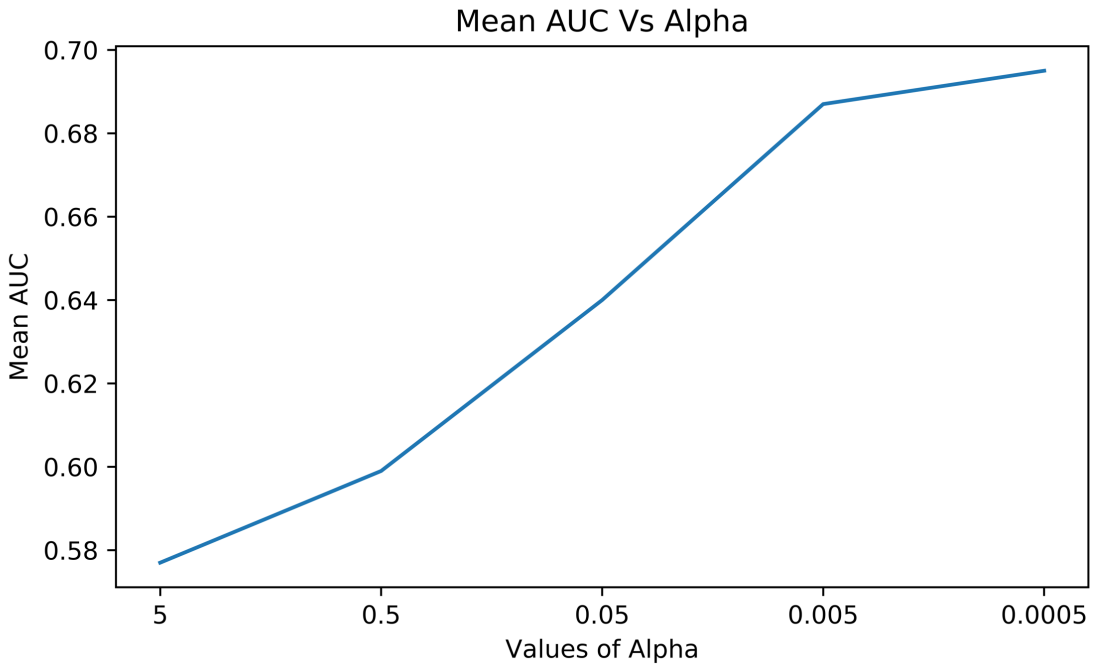

Figure 11 Mean AUC vs Alpha

6.3.2 User-wise AUC vs Popularity-wise AUC

Following figure highlights the user-wise AUC and popularity-wise AUC.

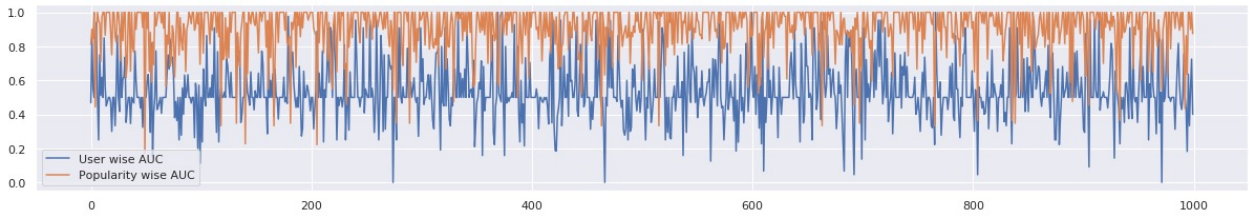

Figure 12 User wise AUC vs Popularity wise AUC

6.3.3 Mean Average Precision at k (MAP@k)

Mean average precision at k is nothing but mean of average precision of each user for k recommendations.
In this research, mean average precision at 10 was calculated and value of the same for this research is 0.052.

6.3.4 Mean Recall

Recall in machine learning is also known as sensitivity that is true positive rate of the model.
Mean recall value for this research is equal to 0.42.

6.4 Summary

In this chapter, different aspect of evaluation is discussed and also result of this thesis is presented. In this chapter values of offline evaluation metrices are presented and explained as well. This chapter also talked about model output.

Chapter 7 Conclusion and Future work

7.1 Introduction

In this chapter, conclusion, contribution of this research and future work will be discussed.

7.2 Discussion and Conclusion

Recommending top-n banking products to customer based on implicit dataset (binary form) has been studied in this research. A scoring approach is developed using implicit dataset to understand the user's confidence- preference on the purchased product which was difficult to understand using implicit dataset, that too if it just contains list of product customer has purchased.

This research has also elaborated how cross selling of the product can be increased using this approach.

This approach is computationally efficient and also provide the list of features which are playing a key role in generating the recommendation of the product. Implementation of proposed scoring approach can be applied to other implicit dataset as well and it is not limited to only this research.

This research also elaborated how implicit dataset is cheaper and accurate in identifying the user's buying/purchasing pattern as compare to the explicit dataset.

Confidence-Preference paradigm is discussed and implemented in this research and also highlighted the importance of the same in banking and FI use cases.

In this research, utilization of implicit data set for recommendation system has been studies and also, it has been explained in this research that recommendation system built using implicit data set is better than explicit data set.

This research proposed an approach to transform binary implicit data set to a score which can be used to understand the importance of the product for that user. This research has also proposed how cross selling can be done using the model based collaborative filtering (on implicit data set). Finally, this research also elaborated the approach of top-n recommendation system for banking products on implicit data set and also explained the list of features which are playing important role in recommending any product (along with its weightage) for the user.

7.3 Contributions to knowledge

Previous studies on recommendation system for banking products mostly focused on explicit data set but after Y. Hu, Y. Koren and C. Volinsky [1] paper on "Collaborative Filtering for Implicit Feedback Datasets," many researches had been done on implicit data set and results were very promising. Netflix prize challenge is one of the most famous use case for this implementation where researchers were able to increase the accuracy more than 10% but this implementation was not tried much in banking domain.

Advantage of implicit data set in other domain especially in media streaming is that it records how many times any user has watched any particular movie hence it is easy to understand the user's interest in the movie even with implicit data set.

But in the case of banking domain it only records which product customer has purchased or which product customer is using which means it records the purchase data of the product in binary form. So, understanding user's interest in implicit data set of binary form is very difficult.

To overcome above problem, concept of scoring approach is proposed to transform binary data into a score. This score reflects users interest in the purchased product.

Also, this research provides the list of features which are playing a key role in recommending the product.

7.4 Future work

This research is only validated on the offline data. This can be also evaluated online (in terms of purchase based on recommended items) and can be taken as part of the future research. This research focused only one form of scoring approach for handling binary form of implicit dataset. Some other form can be also used for developing scoring approach using advance algorithms and it can be an interesting area of research in the future.

Matrix factorization is used in this research for identifying latent features and some other advance algorithm can also be considered to perform this task as part of the future research.

REFERENCES

1. Hu, Y., Volinsky, C. & Koren, Y. (2008). Collaborative filtering for implicit feedback datasets. *Proceedings - IEEE International Conference on Data Mining, ICDM*. p.pp. 263–272.
2. *Francesco Ricci and Lior Rokach and Bracha Shapira, Introduction to Recommender Systems Handbook, Recommender Systems Handbook, Springer, 2011, pp. 1-35*
3. Adebayo A, Agbola I, Ayangbade A & Obajimi O (2015). Bank Products Recommender. *The International Journal Of Engineering And Science (IJES)* ||. [Online]. p.pp. 2319–1805. Available from: www.theijes.com.
4. Köhler Victor (n.d.). *ALS Implicit Collaborative Filtering – Rn Engineering – Medium*. [Online]. Available from: https://medium.com/radon-dev/als-implicit-collaborative-filtering-5ed653ba39fe.
5. Bogdan, M. (2018). Presenting Bank Service Recommendation for Bon Card Customers. *2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*. p.pp. 145–150.
6. C. Aggarwal, "Neighborhood-Based Collaborative Filtering", In Recommender Systems, Springer, PP .29-70,2016.
7. Sharifi, Z., Rezghi, M. & Nasiri, M. (2013). New algorithm for recommender systems based on singular value decomposition method. *Proceedings of the 3rd International Conference on Computer and Knowledge Engineering, ICCKE 2013*. p.pp. 86–91.
8. Shafiei Gol 1, E., Ahmadi 2, A. & Mohebi 3, A. (2016). Intelligent Approach for Attracting Churning Customers in Banking Industry Based on Collaborative Filtering. *Journal of Industerial and Systems Engineering*. [Online]. 9 (4). Available from: http://jise.ir/article_16171.html.
9. Abdollahpouri, H. & Abdollahpouri, A. (2013). An approach for personalization of banking services in multi-channel environment using memory-based collaborative filtering. *IKT 2013 - 2013 5th Conference on Information and Knowledge Technology*. p.pp. 208–213.

10. Shah, K., Salunke, A., Dongare, S. & Antala, K. (2018). Recommender systems: An overview of different approaches to recommendations. *Proceedings of 2017 International Conference on Innovations in Information, Embedded and Communication Systems, ICIIECS 2017*. 2018-Janua. p.pp. 1–4.
11. Linden, G., Smith, B. & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*. 7 (1). p.pp. 76–80.
12. Oard, D.W. & Kim, J. (1998). Implicit Feedback for Recommender Systems. *Proceedings of the AAAI workshop on recommender systems*. p.pp. 81–83.
13. Takács, G., Pilászy, I., Németh, B. & Tikk, D. (2008). Matrix factorization and neighbor-based algorithms for the netflix prize problem. *RecSys'08: Proceedings of the 2008 ACM Conference on Recommender Systems*. p.pp. 267–274.
14. Gigli, A., Lillo, F. & Regoli, D. (2017). Recommender systems for banking and financial services. *CEUR Workshop Proceedings*. 1905. p.pp. 5–6.
15. Lyu, B., Xie, K. & Sun, W. (2017). A deep orthogonal non-negative matrix factorization method for learning attribute representations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 10639 LNCS (3). p.pp. 443–452.
16. Shen, M. & Wang, R. (2018). A New Singular Value Decomposition Algorithm for Octonion Signal. *Proceedings - International Conference on Pattern Recognition*. 2018-August. p.pp. 3233–3237.
17. Khan, S.A. & Ali Rana, Z. (2019). Evaluating Performance of Software Defect Prediction Models Using Area under Precision-Recall Curve (AUC-PR). *2019 2nd International Conference on Advancements in Computational Sciences, ICACS 2019*. p.pp. 1–6.
18. Li, K., Huang, Z., Cheng, Y.C. & Lee, C.H. (2014). A maximal figure-of-merit learning approach to maximizing mean average precision with deep neural network-based classifiers. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. p.pp. 4503–4507.
19. Lancieri, L., Manguin, M. & Mangon, S. (2008). Evaluation of a recommendation system for musical contents. *2008 IEEE International Conference on Multimedia and Expo, ICME 2008 - Proceedings*. 5. p.pp. 1213–1216.

20. Karim, J. (2014). Hybrid system for personalized recommendations. *Proceedings - International Conference on Research Challenges in Information Science*. p.pp. 1–6.
21. Mai, J., Fan, Y. & Shen, Y. (2009). A neural networks-based clustering collaborative filtering algorithm in E-commerce recommendation system. *2009 International Conference on Web Information Systems and Mining, WISM 2009*. p.pp. 616–619.