WEB BASED CLICKSTREAM ANALYSIS

Enroll No. (s) **-14103290, 14103291**

Name of Students **-Anandvardhan, Anubhav Singh**

Name of supervisor **-Dr. Neetu Sardana**

May-2018

Submitted in partial fulfillment of the Degree of

Bachelor of Technology

In

Computer Science Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

TABLE OF CONTENTS

DECLARATION

We hereby declare that this submission is our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by any other person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Place: Signature:

Date: Name:

Enrollment No

CERTIFICATE

This is to certify that the work titled "Web based clickstream analysis" submitted by "Anandvardhan, Anubhav Singh" in partial fulfillment for the award of degree of B.Tech. Of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor ……..

Name of Supervisor ……..

Designation ……..

Date ……..

ACKNOWLEDGEMENT

This project is the result of our effort and the time we spent in making it possible. But it won't be possible without the guidance and instruction of the institution and faculty involved in making this project to come into realization. We will like to extend our thanks to them.

Dr. Neetu Sardana, our project supervisor and mentor, who is supporting us from start through her rigorous introspection and presenting us different aspect of this project to work on it. We would like to extend our gratitude towards her help in making this project.

We are thankful to our institution and its administration for providing us the formats of report and helping us in this project.

Signature of the Student ……..

Name of Student ……..

Enrollment Number ……..

Date ……..

SUMMARY

When we buy a product online, we generally makes comparison to find out which product is better. Clickstream is the log of all the activity we perform on a site and it is maintained by the host of the website or the company which handles the website. So to improve the user experience, these companies analyze and churn the data to a meaningful log and recommend the product on the basis of our search.

What we are doing is that we are making a recommendation system based on the log file or web clickstream data to find out the similar users corresponding to a user and then recommending the products from the similar user basket and also we are trying to maximize the sales by collecting top products that these users buy and then recommending it, so that the experience of a client on site visiting gets enhanced and it will be easier for him to make his choice rather than comparing all the products he sees belonging to a particular category.

_____	_____
Signature of Student	Signature of Supervisor
Name	Name
Date	Date

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS & ACRONYMS

- PCA — Principal component analysis
- KNN — K nearest neighbor
- *P* — Rho, Pearson's coefficient
- CF — Collaborative Filtering

Chapter 1

Introduction

1.1. General Introduction

In the start of online marketing on product based company, they used to provide users to let them compare their products by selecting them and compare their specification one by one. But now with the evolution of machine learning, these company have started to study the online behavior of a user on the basis of how he chooses a product, what are in his ordered list or what products they can recommend him on the basis of his near neighbor users purchase whose similarity matches with the similarity of the user who they are studying.

Market basket analysis is the name of the product we are working on. What is the purpose is to make or recommend a user the similar items so that it could be purchased in his next purchase. To be more exact what we are doing is "to predict among all the products in the user previously ordered list which product he is going to re-order in his next order."

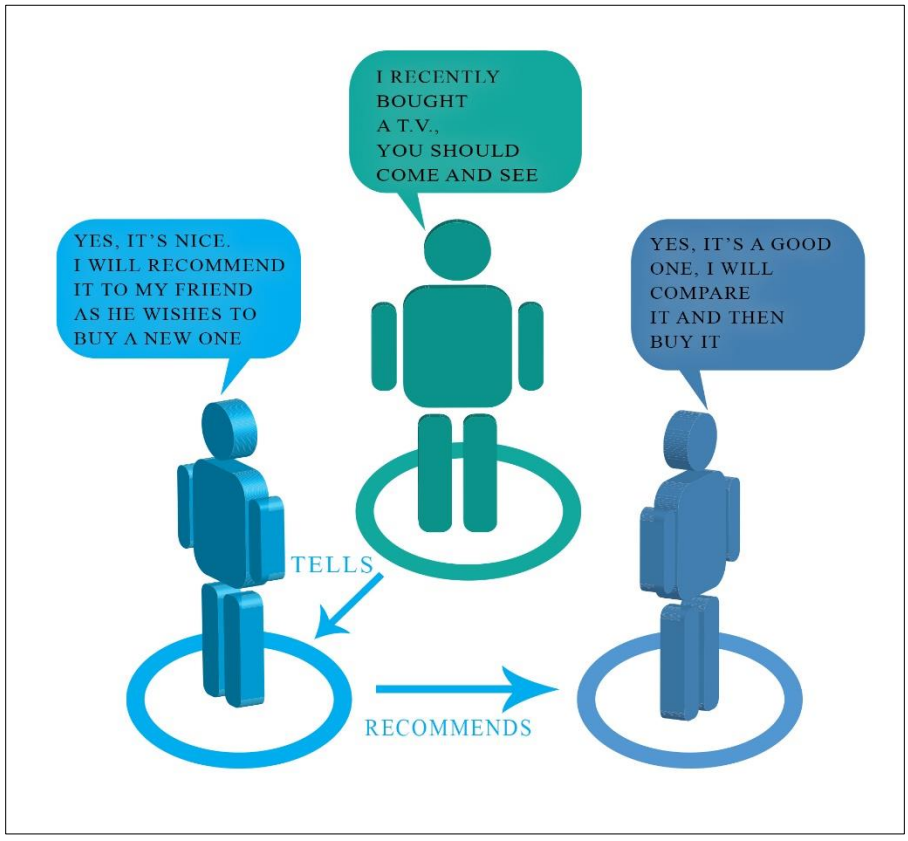

Fig 1. Showing how a recommender system works

For this we are trying to make a recommender system that will help us to determine our desired result. We preprocessed the data to use it further and make it further to be passed through PCA algorithm which is reducing it to dimensions and we have created the 6 physical axis which have no physical significance but is helping to reduce the dimension of our data by making the aisles as a point on a six dimensional figure so that we are able to apply K-mean algorithm on it to make it into clusters according to the users purchasing items belonging to a particular aisle and then selecting the top 10 aisles from 136 aisles in that cluster and recommend its products to the users to buy.

There are two methods of recommender system:

1. Content-based Filtering: Here we study the log of a user to find out the pattern of how he/she buys a product and then recommend the products from the same category.
2. Collaborative Filtering: Here we study the pattern by identifying similar users to a user and recommending the product s of similar users to the user.

For our purpose, we are using **collaborative filtering**.

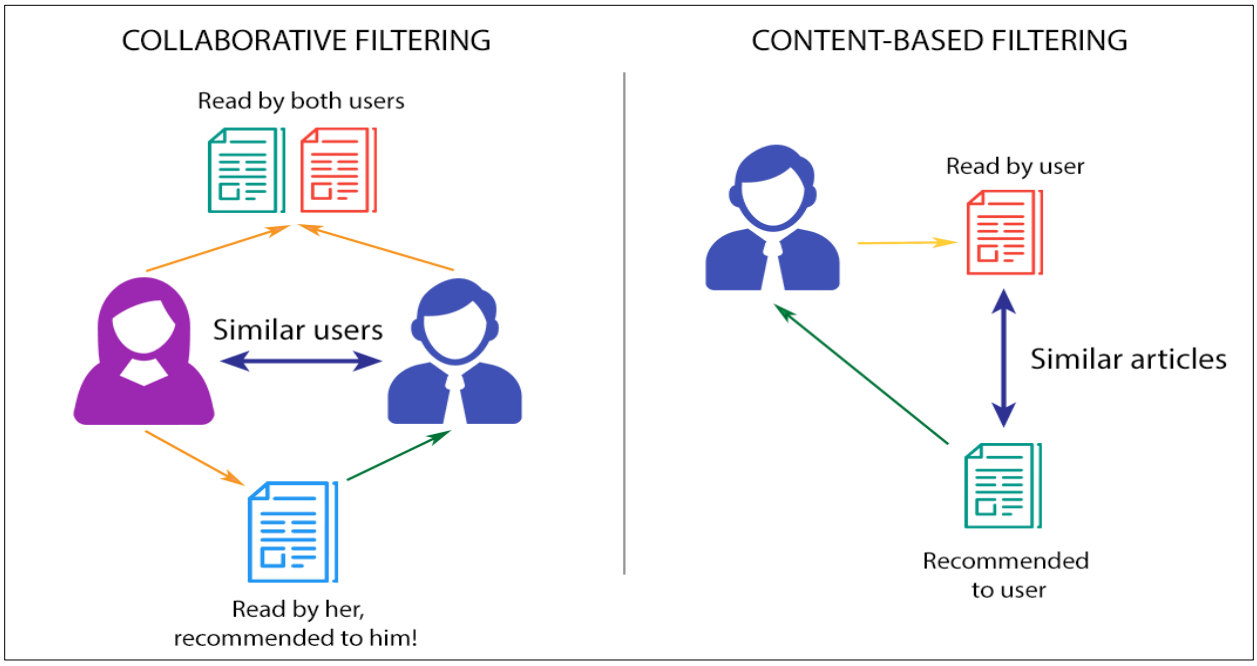

Fig 2. Difference between collaborative and content based filtering

1.2. Problem Statement

We have a set of data file with the extension **.csv.** The problem statement is to determine which product will be in user's next purchase. So simplifying it in technical term will mean that if a user has **k** similar users related to it and in this data we grouped the users in a community based on the similar Aisles (Category) purchased. What we need to do is to find is determining top aisles of each community (as here we have 4 communities) and then we need to find similar users to that user and then we need to suggest the products from those aisles which are in the similar users bucket list.

1.3. Empirical study

We are using the data of Instacart Market Basket Analysis for our recommendation system. Currently to make a recommender system, the company is using **WORD TO VEC** method and **Naïve Bayes classifier**.

WORD TO VEC is a kind of analysis where our system recommends a word most closely related to it. For example, when we type a word in Google search engine, it already recommends a series of words or possible combination of the word we need to write next.
It basically works by capturing the meanings of the word, their semantic relationship and the context they are most widely used. **Fig 3**. Showing the word to vector relationship English keywords and the most closely related word to it. For example **Punctuation** has comma in it, so it has mapped it to **comma**.

NAÏVE BAYES CLASSIFIERS are probabilistic classifiers which use Bayesian Probabilistic method to predict the result which uses strong independence between the features. In the recent years, it has proved to be most legitimate competitors of the method Word to vector. Fig 4. Showing the method of using Naïve Bayesian Classifier whether a product made belongs to **Machine A** or **Machine B**.

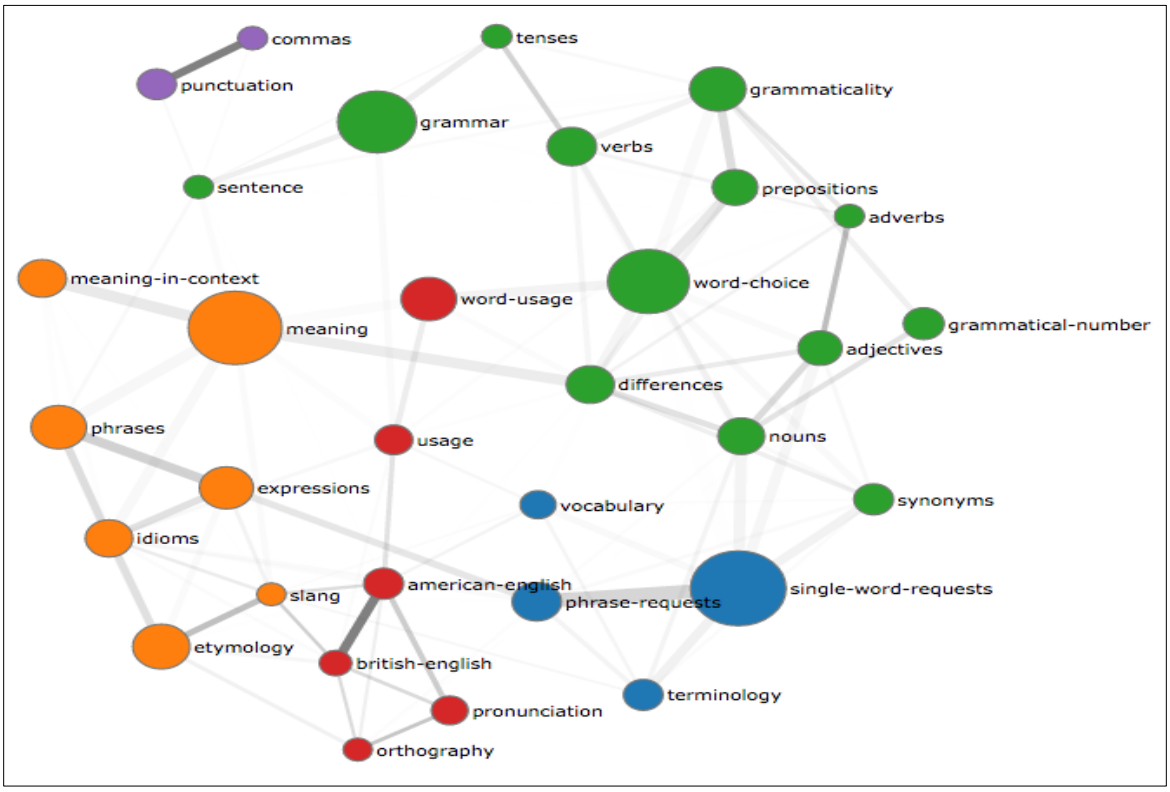

Fig 3. Word to Vector Example

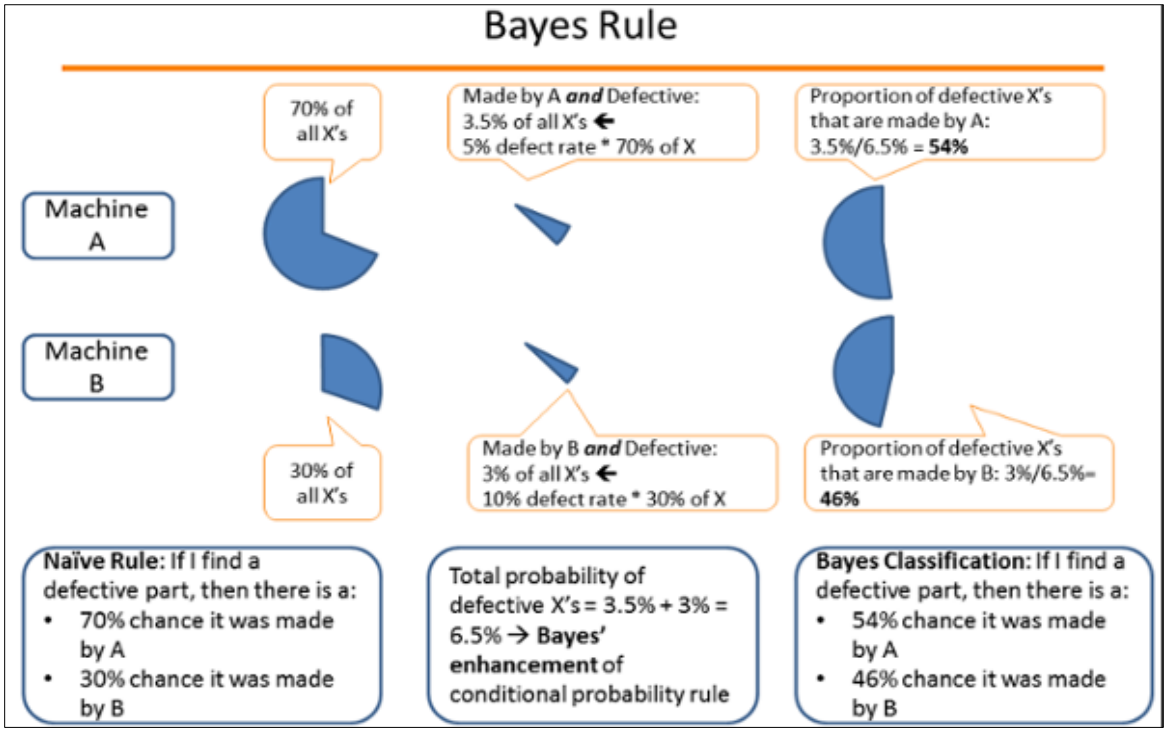

Fig 4. Example of Naive Bayes Classifier

1.4. Approach to problem in terms of technology /platform to be used

We have used Python version 3.6 for our purpose and idle as Spyder. Algorithms used are of machine learning and Data Science. Basically we are making a recommender system for a Grocery store data.

Approach to the problem:

- Fetching of data.
- Refining of data for usage.
- Final pre-processing the data by joining the fields of different datasets.
- Use Principal Component Analysis (PCA) algorithm to find the better representation of data.
- Applying k-mean algorithm to divide the pre-processed data into 4 communities by separating the users on the basis of Aisles (categories).
- Finding Top Aisles from each of the 4 communities.
- Applying Pearson co-relation coefficient to get similar users.
- From the aisles of the similar user, we are selecting only those aisles which are in top aisles.
- Now in those aisle, we are trying to find out what is the most recently n product of the similar users.
- Suggesting those recently bought products to the user.

It is good to have a high end machine to run the project as the data is quite large and needs high memory usage for processing. We have used Tableau Desktop for initial combining of data and to see what are the categories and the content of data.

1.5. Significance of problem

In today's world, it is need for product based company which offers online marketing system to see what its users want and recommend them those products. Already there are many research going on in this field to maximize users experience and increase the sales of that company.

But most of these company use content based filtering as it is easy for them to study an individual user activity rather than studying the pattern of similar users. So we are providing a method of collaborative filtering.

1.6. Give tabular comparison of other existing approaches/ solution to the problem framed

Serial no.	Our method	Existing method
1	It is based on collaborative filtering.	They were based on content based filtering.
2	Studied pattern of users on the basis of similar users.	Studied pattern of users on the basis of interest of that user.
3	Algorithm used are k-mean, knn, Pearson coefficient	Algorithm used are Word to vector, Naïve Bayesian model

Table 1. Comparison of the existing and our method.

Chapter-2

Literature Survey

2.1. Summary of papers studied

Research Paper 1: A Clickstream-Based Collaborative Filtering Personalization Model: Towards a Better Performance, **ISBN**: 1-58113-978-0, 2004 Article

Author's names: Dong-Ho Kim, Vijayalakshmi Atluri, Michael Bieber, Nabil Adam, Yelena Yesha

To be in long run, it has become necessary for the company to continuously monitor its user's search records and to suggest the users the most likely product they can buy, so as to impede the buyer's personalization and buying experience. This research paper concentrates on types of filtering we use while analyzing the results of clickstream.

There are two types of filtering based on what we want to focus on:

- Collaborative filtering: Collaborative Filtering (CF) makes use of peers' evaluations or behaviors for a personalized prediction. Depending on data type in use for CF, it can be classified into either user-based CF or item-based CF.
    - User-based CF employs user data such as user profiles to find the most similar users or a reference group for a user, and based on their preferences it recommends items for the user.
    - Item-based CF employs item data such as products or Web pages to find item relationships based on which it recommends items for a user. User-based CF has been extensively used in research and practice.
- Clickstream-based CF: Clickstream-based CF is a kind of item based CF and it is very efficient in predicting the result by analyzing the data of web. It trains the models offline and uses them in online recommendation. Different from other item-based CF recommendations (e.g., online shopping user activity data), the way data is arranged is important for increasing the quality of recommendation because of the fact the serialized structure is embedded through hyperlinks in Web pages.

Research Paper 2. COMMUNICATIONS OF THE ACM Visualizing online activity
Author's name Stephen G. Eick, doi>10.1145/381641.381710

This research paper is concentrated on user online search pattern and how to apply algorithm to find a pattern in searches and the parameters we can categorize a website with. The more popular website has the more number of visitors each day. This tells us what improvement we need to make in our business model in the order to attract more number of users. We are using these features integration in our project:

Scale: There are a number of Web sites which are quite complex in structure but the best site among them is which gets maximum hits on its pages also called Page View which shows it has millions of user visit each day.

Site usage patterns: The page has its own parameters to represent it. For example it is represented by its color, shape, height of the page, 3D verticals, contents which it incorporates, page error, hits on the page per day and even number of visitors currently visiting that page.

Path visualization: Every user has his/her own way of surfing a page. The idea is to monitor all of his page searching pattern so as to find a path. Further we analyze that path and recommend it to him/her for helping them with a better page visiting experience. It's more like the way we go to a shop at a regular interval, the manager or the seller keeps in mind the category or specialties we want and when we visit the shop again, he/she makes us the suggestion to buy that product as we will like them.

Flow analysis: It is more like setting up some points for making the recommendation to the user, unlike path visualization. It concentrates on most frequently clicked links, bad links, entry and exit points and other problems of the site

2.2. Integrated summary of the literature studied

The point that these research paper concentrates on the type of filtering we are using while analyzing the pattern of the website visit and enhancing the experience of the user and thus enhancing number of user visits per day and the sales.

They also have set parameters on which we visualize the position of the website in the long run and thereby making changes to it. Now-a-days, understanding user's site visit pattern is very important because the site which will give its user a better experience, only it will survive in long run.

To try to study user's site visit pattern, we have two ways, either study them in real time or studying their pattern offline by generated clickstream. The most popular filtering are Collaborative filtering and Clickstream based collaborative filtering. Since we are working on clickstream or offline website data, we are using clickstream based collaborative filtering.

The parameters that the second research paper is talking about are used to visualize user's online activity. How often the visit a website, which page category they browse on for maximum number of times, which page they spend their maximum time of browsing. By defining these parameters, it becomes easy for the developers to keep the track of the likes and dislikes of users and where they can improve.

Chapter 3

Analysis, Design and Modeling

3.1. Overall description of the project

By this project we aim at developing a recommender system by using Clickstream based collaborative filtering. We are making a recommender system which try to find out similar user to a particular user and recommending him/her the products of similar users.

Market basket analysis is the name of the product we are working on. What is the purpose is to make or recommend a user the similar items so that it could be purchased in his next purchase. To be more exact what we are doing is "to predict among all the products in the user previously ordered list which product he is going to re-order in his next order."

We have a set of data file with the extension **.csv.** The problem statement is to determine which product will be in user's next purchase. So simplifying it in technical term will mean that if a user has **k** similar users related to it and in this data we grouped the users in a community based on the similar Aisles (Category) purchased. What we need to do is to find is determining top aisles of each community (as here we have 4 communities) and then we need to find similar users to that user and then we need to suggest the products from those aisles which are in the similar users bucket list.

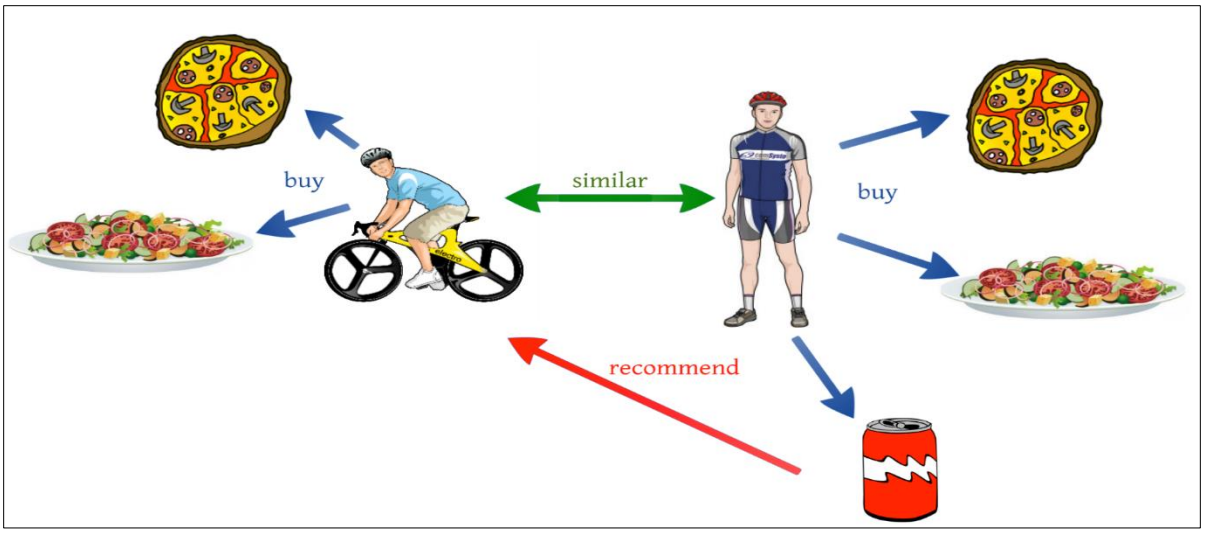

Fig 5. Showing how our recommendation system will work

3.2. Functional requirements

- Uploading of data file
- Merging of data for pre-processing

3.3. Non-Functional requirements

- Representing aisles on 6 dimensional axis (reducing it into dimensions)
- Making of 4 clusters by applying K-Mean algorithm

3.4. Design Diagrams

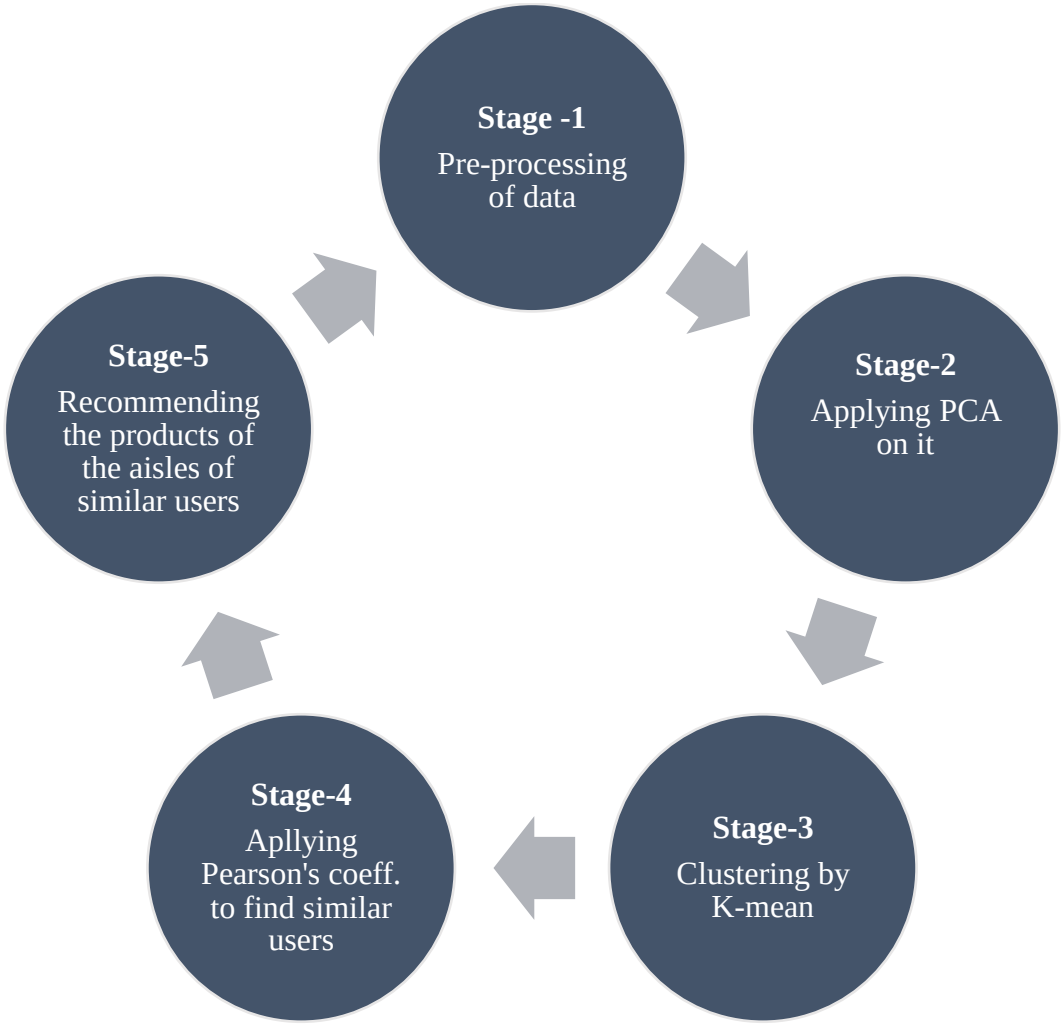

Fig. 6. Different Stages of this project

3.5. Use Case diagrams

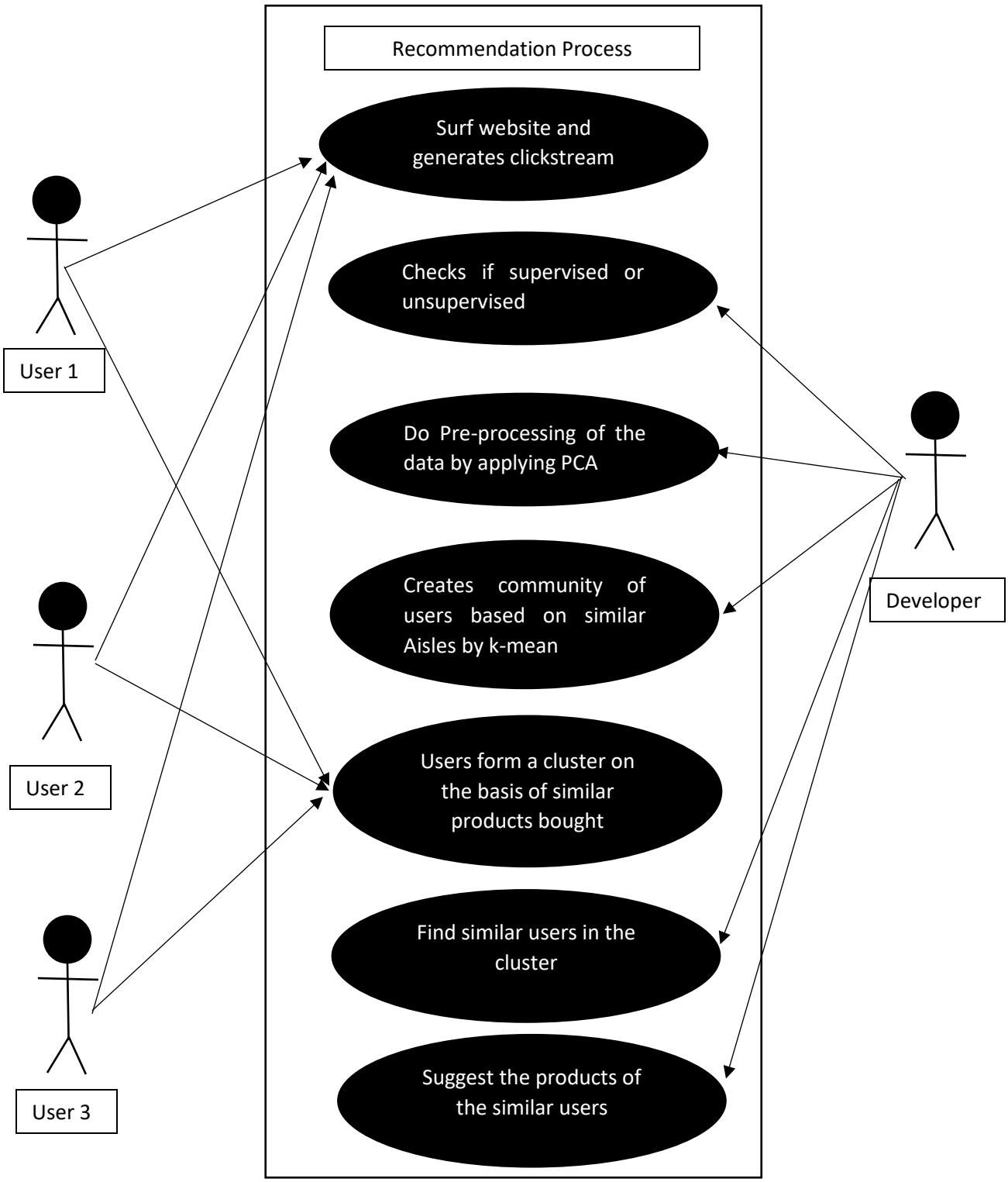

Fig 7. UML use case diagram

Chapter-4

Implementation details and issues

4.1. Implementation details and issues

We are using the data of Instacart market basket analysis. Basically data is unsupervised meaning it is not a trained or trimmed data.

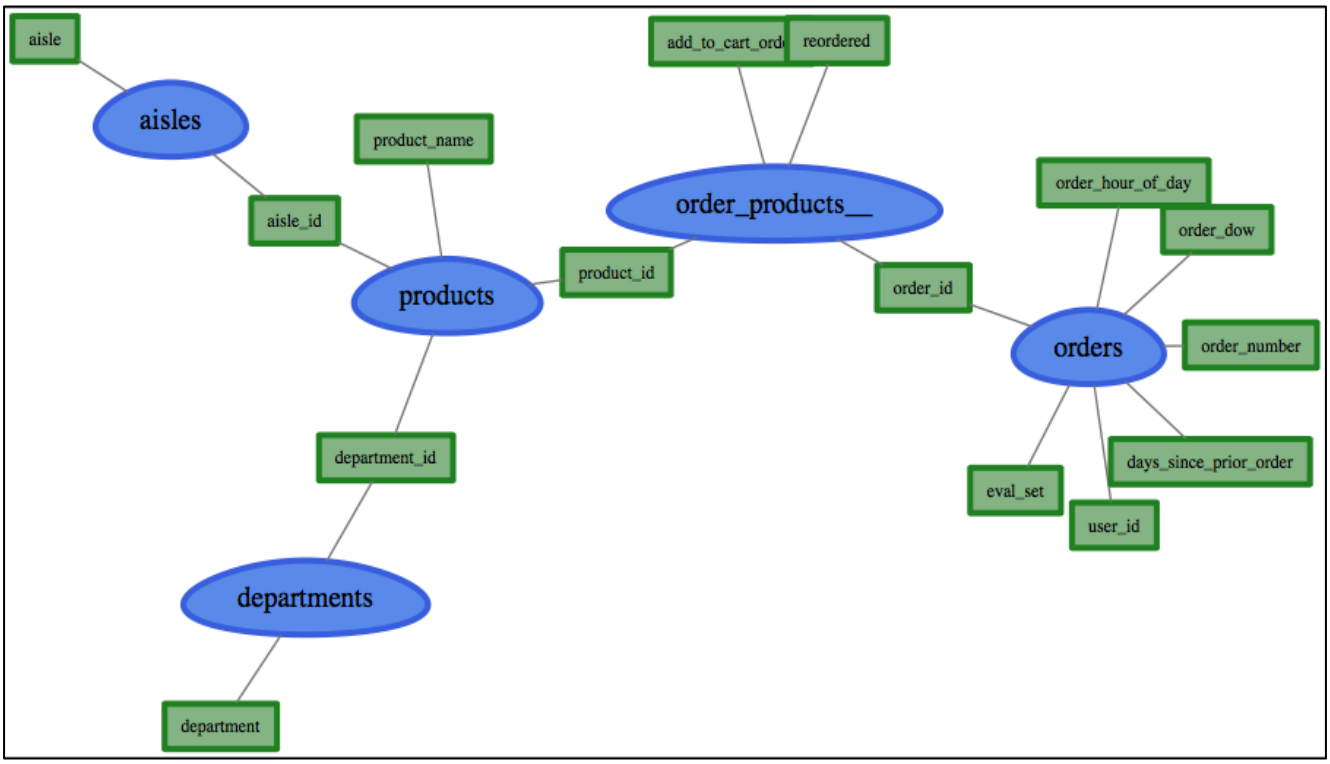

Fig 8. Showing the format and parameters in data

So to use it we are first pre-processing the data first by joining the related fields of the data files to get a complete dataset. Then we applied Principal component analysis algorithm to reduce the dimension of the data meaning find out the most uniform distribution of data. We applied k-mean algorithm to divide it into 4 clusters, each of which comprises of users belonging to similar aisles (purchasing products from similar category). Then we extracted top 10 aisles from each of the clusters. We generated an aisle-user matrix for each cluster and then applied Pearson's correlation coefficient to find "n" similar users who has same interest as to a particular user by passing correlation matrix into recommender system. Since we are trying to suggest the user products belonging to the top 10 aisles only, we first merged the aisles of similar users and find out which of them belongs to the top aisles and then recommending the product which were

bought by them recently. For getting recently bought product of the user, we have used add_to_cart order value and its reorder value.

For example if we need to find the recently bought product of user 1 belonging to the aisle id 21, we will check maximum value of add_to_cart order which is 5 in this case and reorder value is also 1, so it is the recently bought product of the user 1. Table 2 shows these values for user 1.

user id 1	Aisle Id	Order Id	Add To Cart..	Reordered	
True	21	431534	4	1	Abc
		473747	4	0	Abc
		550135	4	1	Abc
		2254736	4	1	Abc
		2295261	3	1	Abc
		2550362	5	1	Abc
		3108588	4	1	Abc
		3367565	4	1	Abc
	23	431534	2	1	Abc
		473747	2	1	Abc
		550135	3	1	Abc
		2254736	2	1	Abc
		2295261	6	1	Abc
		2398795	3	1	Abc
			5	1	Abc
		2539329	3	0	Abc
			4	0	Abc
		2550362	9	1	Abc
		3108588	1	1	Abc
		3367565	2	1	Abc

Table 2. Showing Tableau representation of data for User 1 and aisle ids 21 and 23

4.1.1 Implementation Issues

Since the recommender system uses the algorithms of machine learning and our data is quite large (about 30 millions), so at the time of implementation RAM goes full and the system starts to crash. So for our purpose, we have to work on restrict our data to a particular limit so that we can make a recommender system.

This data is unsupervised data, so pre-processing makes it a quite large file. To see the combined table, we used Tableau Desktop software.

4.1.2 Algorithms (Module wise- with respect to design)

We have used the following algorithms:

- Basic establishment of relationship by joining two csv files with their common field.

- **Principal component analysis algorithm**: Let us take a cubical box and insert about 100 particles in it. All the particles are independently suspended and every particle is in equilibrium unhinged from its place. Now we start seeing the particles from different edges and see from which edge we are getting more uniform distribution of the particle. Now take two of the side of that edge to represent the position of the particles in two dimensional picture. This is what PCA algorithm does to our data, it finds out the more proper distribution of our data points (in our case data points are aisles) from a 6D spread to 2D spread.

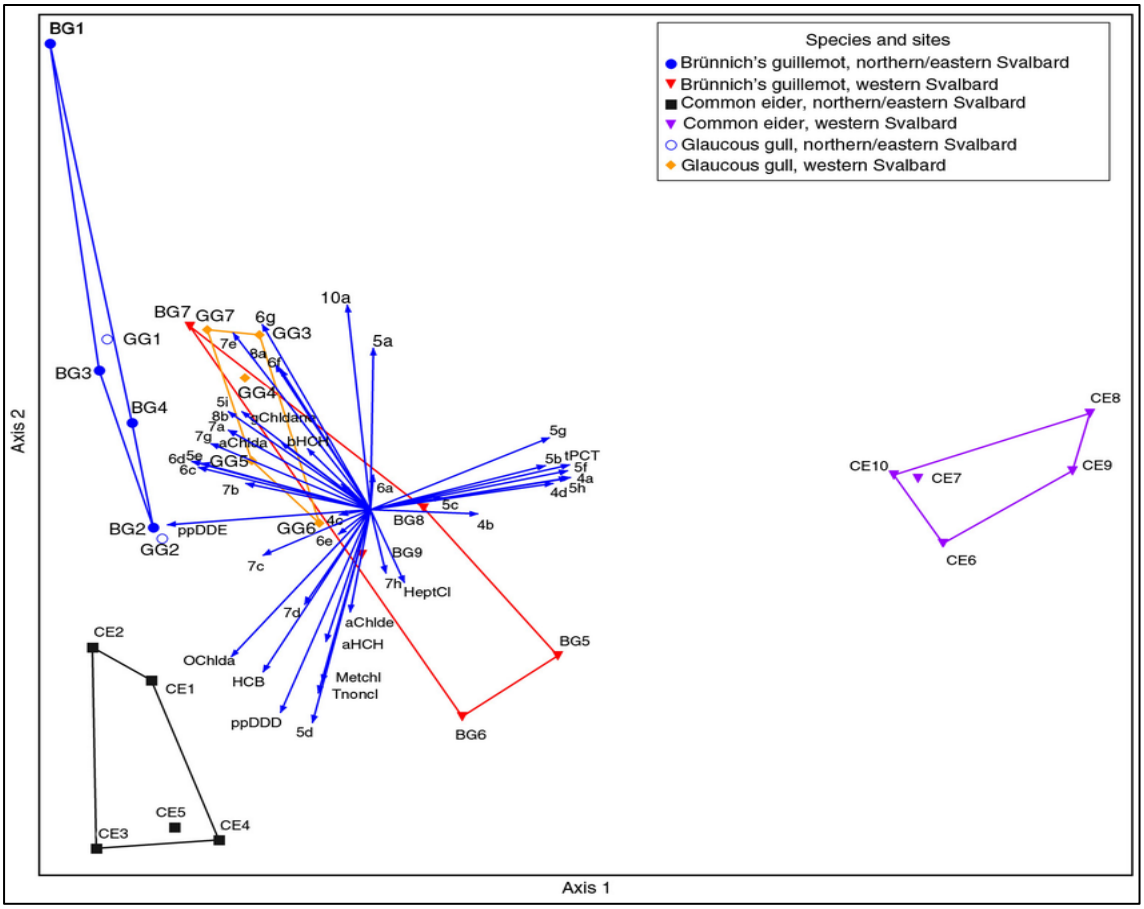

Fig 9. Shows 6D PCA analysis of relativized contaminant concentrations in birds

- **K-mean algorithm**: It is a kind of algorithm that is used on unsupervised data to form k clusters, where k represents the centroid of each cluster. These centroid should be placed in a smart or basically a clever way because of different location causes different result. So for the better result we place them as far as possible from each other, so that they form a better cluster.

  Then we pass these points to a objective function defined J (V) as below:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\left\| x_i - v_j \right\| \right)^2$$

  Where $\mathbf{c_i}$ is the number of data points in cluster I and **c** is the number of cluster centers $||x_i - v_j||$ is the Euclidean distance between the points.

New cluster center is calculated by using the formula

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

Where $\mathbf{c_i}$ is the number of data points in i^{th} cluster and x_i is the i^{th} point in that cluster.

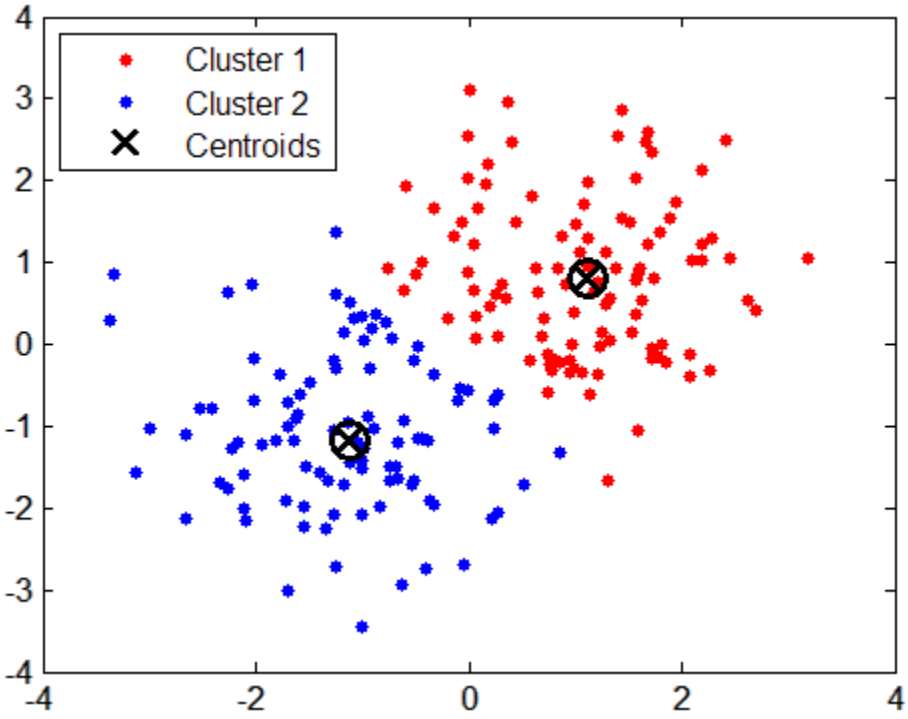

Fig 10. Showing the example of 2 clusters formed by k-mean

- Pearson's correlation coefficient: It is defined as the covariance of the two values divided by the multiplication of their standard deviations. It involves a "product moment", which is the mean about the center of the product of the mean-adjusted random variables.

It is denoted by the Greek symbol ρ (rho). For a population Pearson's coefficient $\boldsymbol{\rho}$ is determined as

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

where:

- cov is the covariance
- σ_X is the standard deviation of X
- σ_Y is the standard deviation of Y

The formula for ρ can be expressed in terms of mean and expectation. Since

$$\operatorname{cov}(X,Y) = \operatorname{E}[(X-\mu_X)(Y-\mu_Y)],$$ [5]

the formula for ρ can also be written as

$$\rho_{X,Y} = \frac{\operatorname{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

where:

- cov and σ_X are defined as above
- μ_X is the mean of X
- μ_Y is the mean of Y
- E is the expectation.

The formula for ρ can be expressed in terms of uncentered moments. Since

- $\mu_X = \operatorname{E}[X]$
- $\mu_Y = \operatorname{E}[Y]$
- $\sigma_X^2 = \operatorname{E}[(X - \operatorname{E}[X])^2] = \operatorname{E}[X^2] - [\operatorname{E}[X]]^2$
- $\sigma_Y^2 = \operatorname{E}[(Y - \operatorname{E}[Y])^2] = \operatorname{E}[Y^2] - [\operatorname{E}[Y]]^2$
- $\operatorname{E}[(X-\mu_X)(Y-\mu_Y)] = \operatorname{E}[(X-\operatorname{E}[X])(Y-\operatorname{E}[Y])] = \operatorname{E}[XY] - \operatorname{E}[X]\operatorname{E}[Y],$

the formula for ρ can also be written as

$$\rho_{X,Y} = \frac{\operatorname{E}[XY] - \operatorname{E}[X]\operatorname{E}[Y]}{\sqrt{\operatorname{E}[X^2] - [\operatorname{E}[X]]^2}\,\sqrt{\operatorname{E}[Y^2] - [\operatorname{E}[Y]]^2}}.$$

- **Recommender System:** With the evolution of machine learning, it has been made possible to do a clickstream based collaborative filtering, which is offline user activity monitoring.

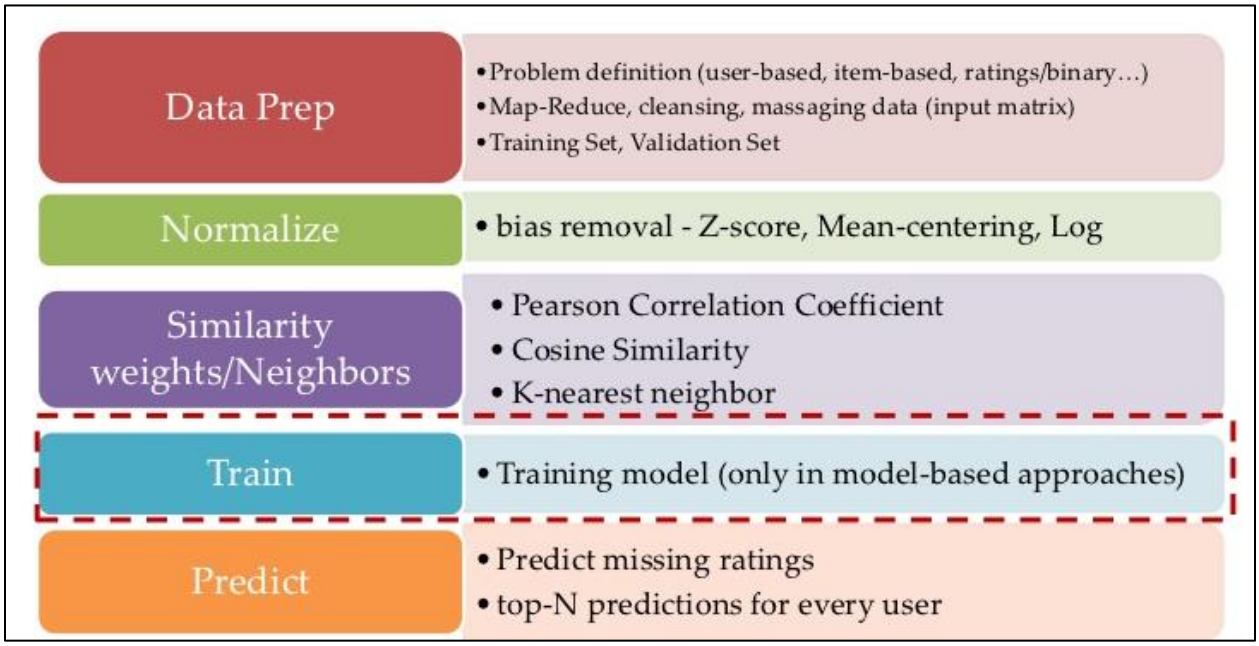

Fig 11. Major Steps in making a recommendation system

Chapter 5

Testing

5.1. Testing plans

To test our model, we limited our data to 10,000 and performed step-by-step analysis on it. Since the data is very large, so seeing the size of RAM, it was the need to limit the data.

5.2 List all test cases in prescribed format for each algorithm

Sample Data used:

aisles - DataFrame

Index	aisle_id	aisle
0	1	prepared soups salads
1	2	specialty cheeses
2	3	energy granola bars
3	4	instant foods
4	5	marinades meat preparation
5	6	other
6	7	packaged meat
7	8	bakery desserts
8	9	pasta sauce
9	10	kitchen supplies
10	11	cold flu allergy
11	12	fresh pasta
12	13	prepared meals
13	14	tofu meat alternatives
14	15	packaged seafood
15	16	fresh herbs
16	17	baking ingredients
17	18	bulk dried fruits vegeta…
18	19	oils vinegars

Fig 12. Aisles Data frame

prior - DataFrame

Index	order_id	product_id	add_to_cart_order	reordered
0	2	33120	1	1
9	3	33754	1	1
17	4	46842	1	0
30	5	13176	1	1
56	6	40462	1	0
59	7	34050	1	0
61	8	23423	1	1
62	9	21405	1	0
77	10	24852	1	1
92	11	30162	1	1
97	12	30597	1	1
112	13	17330	1	0
125	14	20392	1	1
136	15	19660	1	1
141	16	9755	1	1
144	18	8021	1	0
172	19	42265	1	1
175	20	35430	1	0
183	21	5009	1	1

Fig 13. Order product prior Data Frame

product - DataFrame

Index	product_id	product_name	aisle_id	department_id
0	1	Chocolate Sandwich Cook…	61	19
1	2	All-Seasons Salt	104	13
2	3	Robust Golden Unsweetened O…	94	7
3	4	Smart Ones Classic Favor…	38	1
4	5	Green Chile Anytime Sauce	5	13
5	6	Dry Nose Oil	11	11
6	7	Pure Coconut Water With Or…	98	7
7	8	Cut Russet Potatoes Stea…	116	1
8	9	Light Strawberry Bl…	120	16
9	10	Sparkling Orange Juice …	115	7
10	11	Peach Mango Juice	31	7
11	12	Chocolate Fudge Layer C…	119	1
12	13	Saline Nasal Mist	11	11
13	14	Fresh Scent Dishwasher Cl…	74	17
14	15	Overnight Diapers Size 6	56	18
15	16	Mint Chocolate Flavored Syrup	103	19
16	17	Rendered Duck Fat	35	12
17	18	Pizza for One Suprema Froz…	79	1
18	19	Gluten Free Quinoa Three …	63	9

Fig 14. Product Data frame

We are merging these Data frames to a merged table which will have user id, aisle id, product id, add_to_cart order and their reorder value set as 1 if the product is being purchased else 0.

Result of merged table

Index	order_id	product_id	add_to_cart_order	reordered	product_name	aisle_id	department_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order	aisle
0	2	33120	1	1	Organic Egg Whites	86	16	202279	prior	3	5	9	8	eggs
1	26	33120	5	0	Organic Egg Whites	86	16	153404	prior	2	0	16	7	eggs
2	120	33120	13	0	Organic Egg Whites	86	16	23750	prior	11	6	8	10	eggs
3	327	33120	5	1	Organic Egg Whites	86	16	58707	prior	21	6	9	8	eggs
4	390	33120	28	1	Organic Egg Whites	86	16	166654	prior	48	0	12	9	eggs
5	537	33120	2	1	Organic Egg Whites	86	16	180135	prior	15	2	8	3	eggs
6	582	33120	7	1	Organic Egg Whites	86	16	193223	prior	6	2	19	10	eggs
7	608	33120	5	1	Organic Egg Whites	86	16	91030	prior	11	3	21	12	eggs
8	623	33120	1	1	Organic Egg Whites	86	16	37804	prior	63	3	12	3	eggs
9	689	33120	4	1	Organic Egg Whites	86	16	108932	prior	16	1	13	3	eggs

Fig 15. Merged Table

Relationships in merged table:

Orders.csv join with order_product_prior.csv on order_id.

This table is then joined with products.csv on product_id.

The newly formed table is then joined with aisles.csv on aisle_id.

Now we are applying Principal Component Analysis to reduce the dimension of the data.

Result of PCA

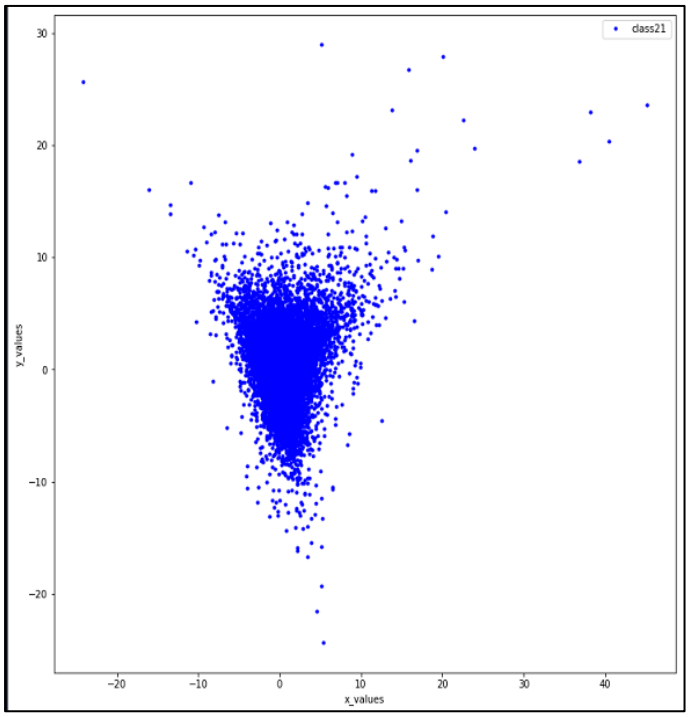

Fig 16. Results of PCA on axis 2, 1

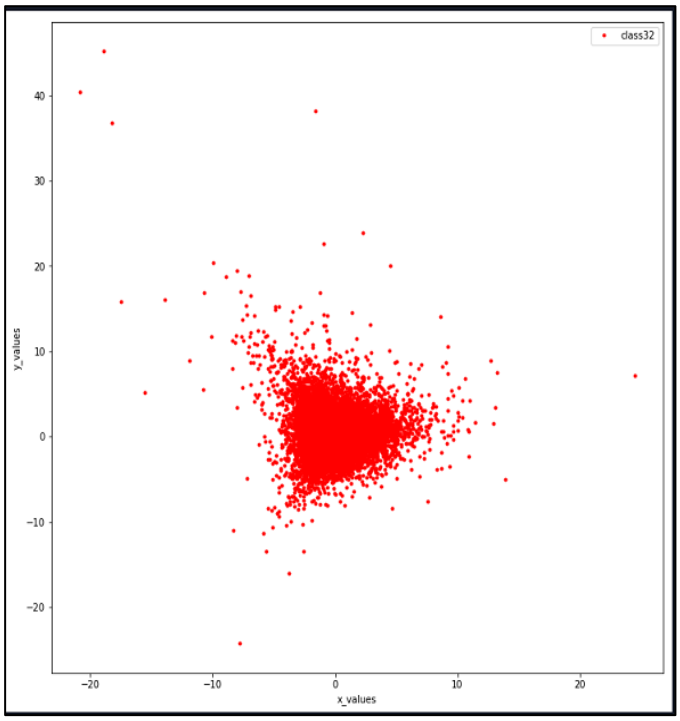

Fig 17. Results of PCA on axis 3, 2

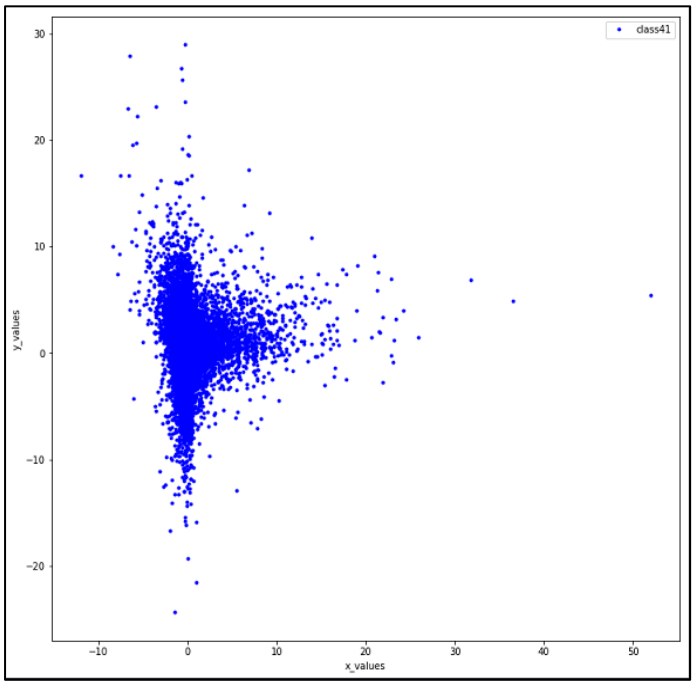

Fig 18. Results of PCA on axis 4, 1

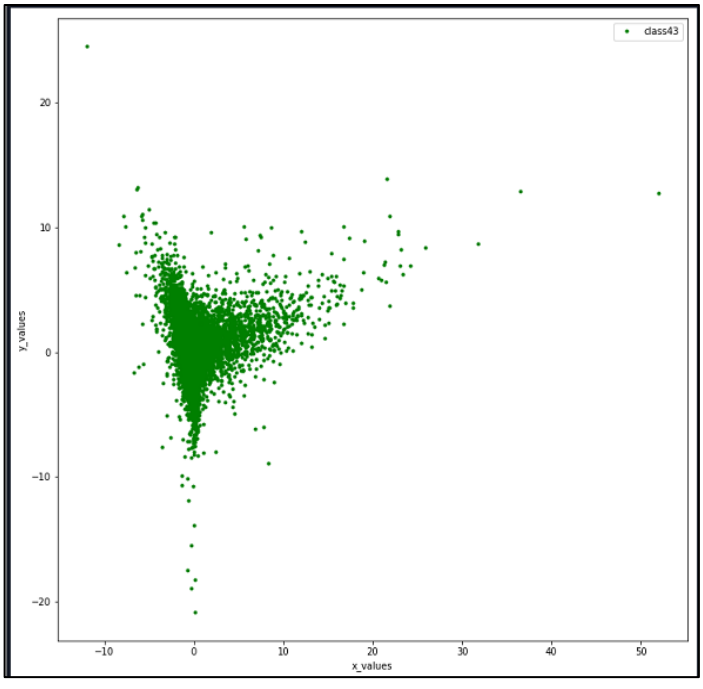

Fig 19. Results of PCA on axis 4, 3

pca_ - DataFrame

Index	0	1	2	3	4	5
0	-2.70897	-0.392952	-0.156048	-0.267602	0.0490034	-0.0340914
1	-0.839879	0.89138	-1.15074	0.511655	-0.320113	-1.29067
2	13.0392	-6.70465	2.14408	3.21587	-1.07647	-1.95815
3	-2.62677	-0.464926	-0.157908	-0.244025	0.115529	-0.135962
4	-1.31339	0.168963	-0.432722	0.729473	-0.0958182	-0.741344
5	2.00645	2.96183	-0.656181	-0.75472	-0.577263	0.615854
6	-1.71447	1.33469	3.00366	-1.15235	-0.217589	-1.0324
7	-1.33984	1.94116	2.62878	-1.54625	-0.271593	0.405963
8	1.25955	2.03876	4.13361	-2.46294	0.98276	-0.271945
9	-2.72655	-0.38039	-0.134925	-0.287762	0.0556487	0.0112049
10	-0.594921	-1.29651	-0.380833	-0.616974	0.224918	0.0355177
11	-2.49724	-0.0897381	0.25325	0.581762	-0.106266	0.373664
12	-1.69342	0.528867	-0.274075	0.487605	-0.131592	0.498674
13	-0.483489	1.46177	-1.55895	-0.144237	-0.13311	-0.0711344
14	5.42123	-2.2518	-1.39509	-1.61757	0.168803	1.72471
15	-1.30837	-0.521004	-0.442455	-0.434949	0.202205	0.073112
16	3.29586	2.8257	-0.798623	-1.41207	0.136839	1.01905
17	-2.75741	-0.466461	-0.234051	-0.405523	0.154351	-0.195037

Fig 20. PCA Segmentation result

Now we use these PCA results to make the clusters by k-mean, we have restricted the number of cluster in k-mean by 4.

Result of k-mean clustering

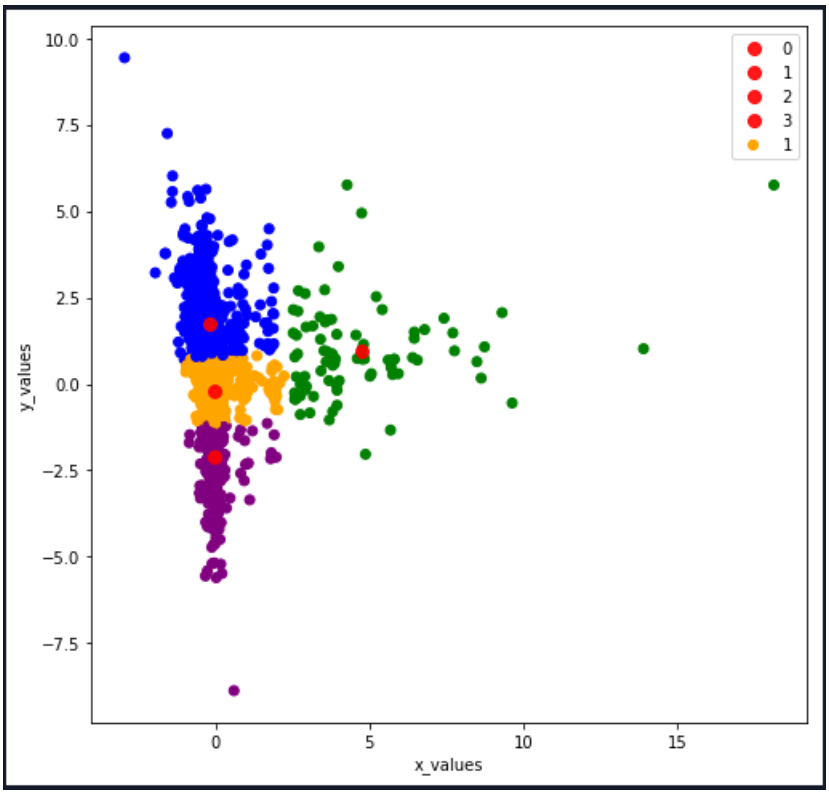

Fig 21. 4 clusters formed by k-mean

Since we have here 4 clusters, we will now plot Aisles purchase verses user graph to get which are the top aisles in the cluster.

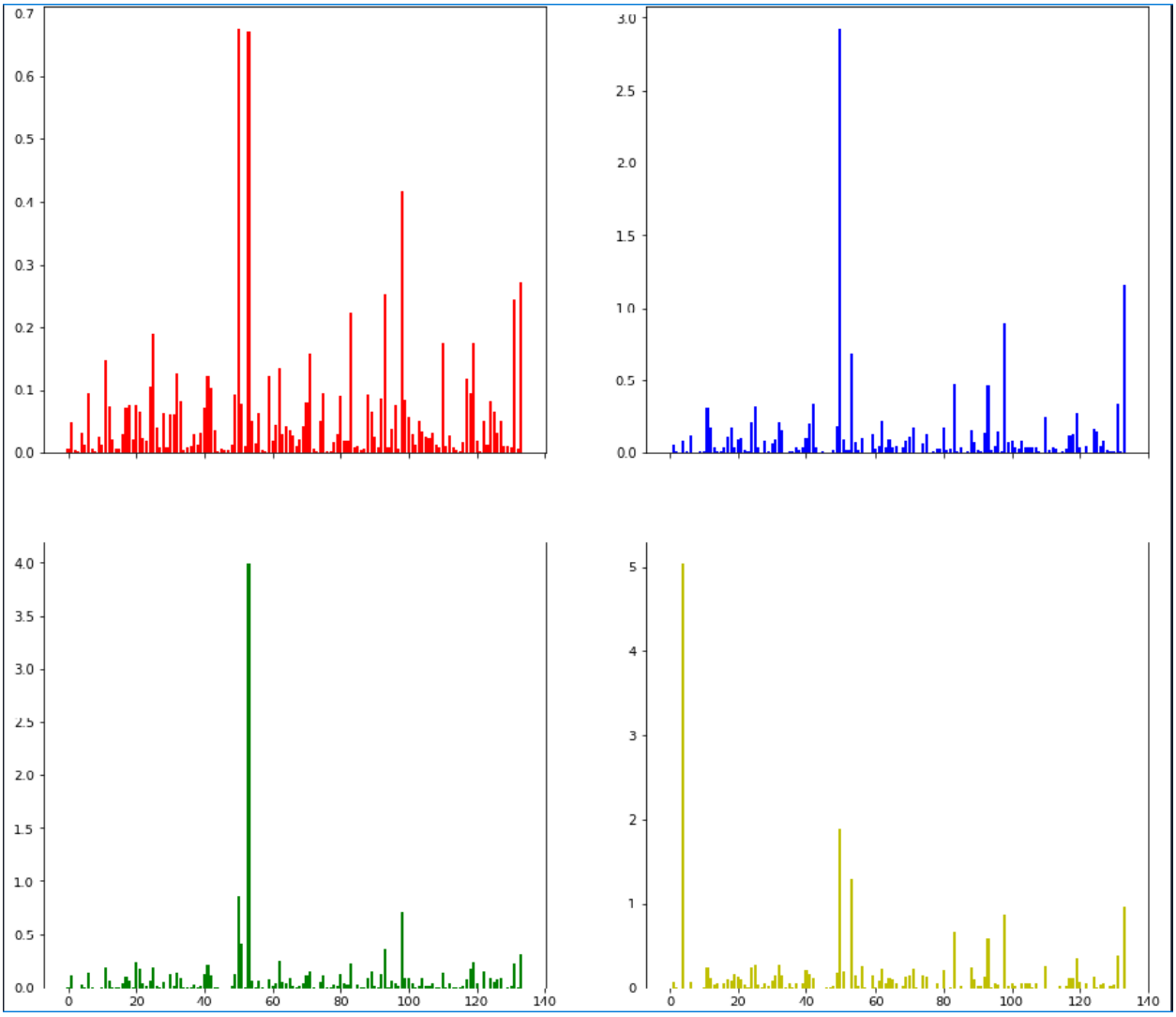

Fig 22. Aisles purchase verses user plot

Finding Top 10 Aisles from each cluster

clust0_top_prods - Series

Index	0
fresh fruits	1.12757
fresh vegetables	0.885313
packaged vegetables fr…	0.599984
yogurt	0.40275
packaged cheese	0.346492
water seltzer sparkling wat…	0.318815
milk	0.316865
chips pretzels	0.260025
soy lactosefree	0.227745
bread	0.211947

clust1_top_prods - Series

Index	0
fresh vegetables	5.62282
fresh fruits	1.78197
packaged vegetables fr…	1.24828
fresh herbs	0.542278
packaged cheese	0.536091
yogurt	0.410747
soy lactosefree	0.399633
milk	0.387489
frozen produce	0.365605
canned jarred vegetables	0.34269

clust2_top_prods - Series

Index	0
fresh fruits	4.85167
yogurt	2.52472
packaged vegetables fr…	1.65325
fresh vegetables	1.24631
milk	0.955087
packaged cheese	0.947321
water seltzer sparkling wat…	0.860601
chips pretzels	0.837173
energy granola bars	0.731426
refrigerated	0.665933

clust3_top_prods - Series

Index	0
baby food formula	8.17612
fresh fruits	3.94692
fresh vegetables	2.7684
yogurt	1.86369
packaged vegetables fr…	1.52232
packaged cheese	1.14234
milk	1.08806
chips pretzels	0.569361
water seltzer sparkling wat…	0.563329
bread	0.558504

Fig 23. Showing top 10 products from each cluster

Result of Correlation matrix that is to be passed in finding similar user by Pearson's correlation coefficient

clust_prod - DataFrame

Index	hs gelatinses	specialty wines champagnes	spices seasonings	spirits	spreads	tea	tofu meat alternatives	tortillas flat bread	trail mix snack mix	trash bags liners	vitamins supplements	water seltzer sparkling water	white wines	yogurt	user_id	cluster
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	4	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0
2	0	0	6	0	0	0	0	0	0	0	0	0	0	1	10	1
3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	13	0
4	0	0	4	0	0	0	0	0	0	0	0	0	0	0	19	0
5	0	0	0	1	1	1	0	0	0	0	0	1	0	2	21	2
6	0	0	0	0	0	0	0	0	0	0	0	1	0	4	23	0
7	0	0	0	0	0	0	0	0	0	0	0	2	0	4	27	2
8	0	0	0	0	1	1	0	0	0	0	0	0	0	6	32	2
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	36	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	37	0
11	0	0	0	0	0	1	0	0	0	0	0	0	0	0	42	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	46	0
13	0	0	0	0	0	0	0	0	0	0	0	1	0	0	50	0
14	0	0	0	0	0	0	0	0	0	0	0	1	0	0	54	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	57	0
16	2	0	0	0	0	0	0	0	0	0	0	0	0	2	63	2

Fig 24. Matrix containing user id and number of products bought from an aisle

Result of Pearson's correlation coefficient

```
5 most similar users for User 7:

1: User 45636, with similarity of 0.862561143966309
2: User 12292, with similarity of 0.7920292986081121
3: User 4311, with similarity of 0.7780233874386261
4: User 16566, with similarity of 0.7721990645865422
5: User 5652, with similarity of 0.7721990645865422

In [6]:
```

Fig 25. Finding similar users to the user id 7

```
In [6]: user_clust
Out[6]:
1    0
Name: cluster, dtype: int32

In [7]: similarities
Out[7]:
array([1.        , 0.86256114, 0.7920293 , 0.77802339, 0.77219906,
       0.77219906])

In [8]: indices
Out[8]: array([[    7, 45636, 12292,  4311, 16566,  5652]], dtype=int64)

In [9]: aisle_bucket
Out[9]:
['eggs',
 'water seltzer sparkling water',
 'milk',
 'spreads',
 'meat counter',
 'oils vinegars',
 'fresh fruits',
 'soy lactosefree',
 'refrigerated pudding desserts',
 'frozen dessert',
 'candy chocolate',
 'packaged vegetables fruits',
 'refrigerated',
 'packaged cheese',
 'cookies cakes',
 'crackers',
 'coffee',
 'fruit vegetable snacks',
 'lunch meat',
 'yogurt']

In [10]:
```

Fig 26. Showing the results of products in similar user list from top aisles

5.3 Limitations

This data is quite large which require a system of higher processing power and bigger RAM. We have currently run this project on Intel i5 and Intel i7 processor, both the system having a RAM of 8GB. One of the system having inbuilt Intel graphics card while other has NVIDIA 4GB dedicated graphics memory. So running this project require GPU, so much of the time one of the system with inbuilt graphics memory get hanged. Data being large, we are not able to see all of its rows, we can only get data of discrete values rather than continuous value.

Processing of data will be slow as it requires the file to be read one-by-one, since our data is about 30 millions in range, so it runs slow.

Result of k-mean can vary by increasing or decreasing the number of clusters. For our purpose, we have used the number of clusters to be 4.

Result of PCA is subjected to which axis we are taking. To be more precise for getting the most uniform distribution we have used the axis 4, 1.

Pearson's correlation coefficient is calculated by passing correlation matrix to it, we have made correlation matrix on the basis of users belonging to a particular aisles (since there are only 136-138 aisles approx.), you can make the matrix on user buying an item (since the item are about more than 20,000 in numbers, the matrix will have large dimensions).

Chapter-6

Findings & Conclusion

6.1. Findings

- Top aisles are different for each of the clusters
- More the number of clusters, more will be the number of centroids. But the major constraint is on choosing what should be that number. Since we choose the centroids to apart from each other, so that all the clusters are divided properly. Basically choosing 4 to be the number of clusters is the point of taking even number of centroids which will evenly lead to the distribution of all the aisles in it.
- We have subjected our aisles value to be 10 only because as a product based company, it will be more preferable to give its client a hassle less experience, so suggesting all the products from the aisles will create ambiguity and will further in confuse the user.
- PCA creates a 6D representation of aisles. The principal axis in PCA don't have any physical significance, they are just for the representation of data points.

6.2 Conclusion

This project is all about creating a Recommendation System for Grocery Store by using Instacart Market basket analysis. The Project uses all the datasets to merge up on the coinciding field to establish a relationship, which is further used in implementing PCA on it. PCA is reducing the dimension of data from 136 points of aisles on it by representing it in a 6 dimensional plot. We can use any side of the 6D figure to represent the data but the more appropriate side will be that which has more uniform distribution of data. By k-mean we can find k center points in the data but we have to restrict it to minimum numbers of centers and also we have to choose our center points cunningly. We could have used KNN algorithm to find k nearest neighbor to the user and then finding cosine similarity between the users, but with the Pearson's correlation coefficient everything is done own its own. This project belongs to the field of Data Analytics and Machine Learning, so it is recommended to use a high end system having dedicated graphics memory either of NVIDIA or AMD and CPU to be I7 processor with RAM of 8GB or above.

6.3 Future Work

- **Use rule mining to find the complementary products**
- **Make recommendation system based UI designs**
- **Using it on supervised datasets**

References

[1] **Dong-Ho Kim, Vijayalakshmi Atluri, Michael Bieber, Nabil Adam, Yelena Yesha, "A Clickstream-Based Collaborative Filtering Personalization Model: Towards a Better Performance", ISBN: 1-58113-978-0, 2004 Article**

[2] **Stephen G. Eick, "COMMUNICATIONS OF THE ACM Visualizing online activity", doi>10.1145/381641.381710**

[3] **Instacart Market Basket Analysis data, (n.d.),** "**https://tech.instacart.com/3-million-instacart-orders-open-sourced-d40d29ead6f2**"

[4] **Pearson's correlation coefficient, (n.d.),** "**http://www.statstutor.ac.uk/resources/uploaded/perasons.pdf**"