A logo for college computing

Description automatically generated

**Assessment Cover Page**

|  |  |
| --- | --- |
| *Student Full Name* | Alina Marrulyn Lozano Flores |
| *Student Number* | 2024068 |
| *Module Title* | Data Visualization Techniques, Machine Learning for Business |
| *Assessment Title* | CA2 |
| *Assessment Due Date* | 24th November 2024 |
| *Date of Submission* | 29th November 2024 |

**Declaration**

By submitting this assessment, I confirm that I have read the CCT policy on academic misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source.

I declare it to be my own work and that all material from third parties has been appropriately referenced.

I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Contents

[Machine Learning for Business 1](#_Toc183820395)

[Recommendation Systems 1](#_Toc183820396)

[**Content Based Recommender System** 1](#_Toc183820397)

[Market Basket Analysis 1](#_Toc183820398)

[Conclusion 1](#_Toc183820399)

[Data Visualization 1](#_Toc183820400)

[Data preparation 4](#_Toc183820401)

[Conclusion 5](#_Toc183820402)

[References 5](#_Toc183820403)

Illustrations

[Figure 1 - TfidVectorizer code 2](#_Toc183820523)

[Figure 2 - Recommendation for Item 2747774 2](#_Toc183820524)

[Figure 3 - Dataset head 3](#_Toc183820525)

[Figure 4 - Data split, y is user\_id 3](#_Toc183820526)

[Figure 5 - Baseline function 4](#_Toc183820527)

[Figure 6 - Mean score 4](#_Toc183820528)

[Figure 7 - Score Weighted Mean 4](#_Toc183820529)

[Figure 8 - Body type score 5](#_Toc183820530)

[Figure 9 - - Data split, y is item\_id 5](#_Toc183820531)

[Figure 10 - Baseline score using with Item as dependable variable 5](#_Toc183820532)

[Figure 11 - The mean score for item 6](#_Toc183820533)

[Figure 12 - Frequent users 50 or more times 6](#_Toc183820534)

[Figure 13 - List of transactions for MBA. 7](#_Toc183820535)

[Figure 14 - MBA Table 8](#_Toc183820536)

[Figure 15 - Items and their respective support 8](#_Toc183820537)

[Figure 16 - Results of FP Growth Algorithm 9](#_Toc183820538)

[Figure 17 - FP Growth vs Apriori Model - Extracted from Medium.com 9](#_Toc183820539)

[Figure 18 - Interactive Scatter Plot 10](#_Toc183820540)

[Figure 19 - Interactive bar graph 11](#_Toc183820541)

[Figure 20 - Category rented for 11](#_Toc183820542)

[Figure 21 - Screenshot from the website: https://www.renttherunway.com/ 12](#_Toc183820543)

[Figure 22 - Dataset for interactive barchart 13](#_Toc183820544)

[Figure 23 - Dataser for barchart 13](#_Toc183820545)

# Machine Learning for Business

For this project the data used data about a store that rents clothes. The are many variables like size, year, item id and user id. The first model used was Content based recommender, based in the rating of the users.

Due the high volume of rows and hardware limitations, tha data has been reduced in the frequency of users, on this case the number of users that appear more 50 or more times are going to be in our analysis, this number is high, but due the hardware limitations it was not possible to low the number.

Anyway, it is possible to adjust this number to 5 and we can analyse the customers that appear more than five times, that it makes more sense.

If we apply this change and also we adjust our index, we can run the code normally.

The second part for Market Basket Analysis is has a dataset with transactions and the list of items bought in a store. The dataset includes the part of the day of the transaction (morning-afternoon) and if the transaction was made on weekend or weekday, for this analysis those columns have been dropped.

With this clarification we can proceed with our report.

## Recommendation Systems

A recommendation system is basically a system build to recommend you different services or products based on reviews, features or users that consumed this service.

This system needs to predict the features that a user could needs and should recommend the best option available.

Mainly, there are two types of recommendation systems, one is the Collaborative filtering that is divided in user based filtering and item based filtering.

The second type of recommendation system is Content based systems.

One advantage of this system is that it does not need to know the past behaviour of the user, only on their profile and metadata of the product. One example of this systems is Netflix.

### Content Based Recommender System

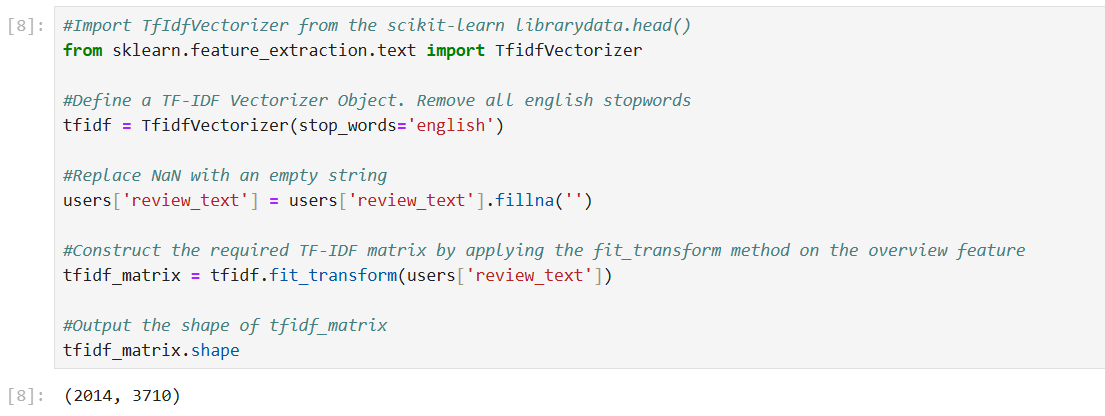
The first model applied in our dataset was the Content Based Recommender System. The variable used was user\_id, item\_id(that should be unique) and review\_text.   
  


Figure - TfidVectorizer code

To analyse the text is necessary to apply an algorithm that analyses word by word, on this case TFidVectorizer was used, the stop words were specified in English language and this algorithm weights the words and define which word has more weight. On this case we analysed the similar words that are between the comments and suggest the items to another users based on this.

The similarities are measure on the Cosine similarity, this is measure of similarity between two vectors, the range of this measure is between -1 to +1, 1 it means the vectors are very similar and -1 the vectors are totally opposite.

To measure this similarity the algorithm Linear Kernel was used.

Finally we created a function to get the item clothes as input and give us recommendations. We have to sort the items clothes based on the similarity scores. On this case we choose the 10 most similar items based on the item number that we assign.

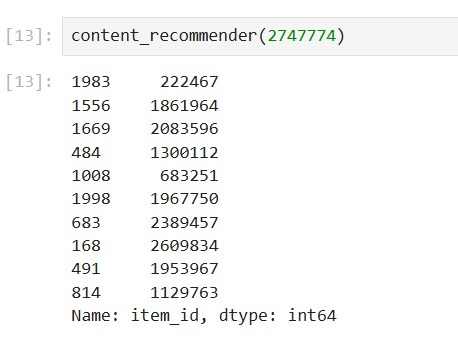


Figure - Recommendation for Item 2747774

**Collaborative Filtering**

This type of recommendation system needs the history of the user’s behaviour. The system can be built based on the similarities between users or the similarities between items that the user has bought.

For this analysis the data from Rent The Runway, a company that rents designer clothes. The dataset includes variables as size, measures and weight.

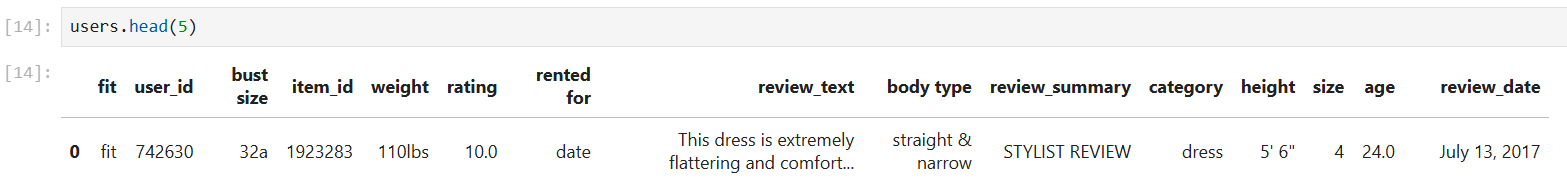


Figure - Dataset head

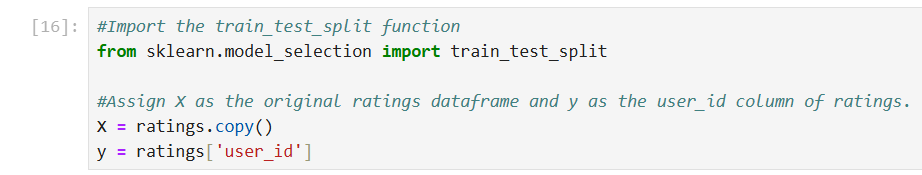
****

Figure - Data split, y is user\_id

First, we need to split the data and our dependable variable is going to be users.

To start we need to create a baseline model that serves as reference against other more complex models, it is important to have a baseline line model as reference to understand if other models are giving a better result.

In our baseline model we defined to always return a rating of 5, this is a benchmark for performance evaluation in rating prediction tasks. This approach makes sure that any advanced models offers a better prediction.

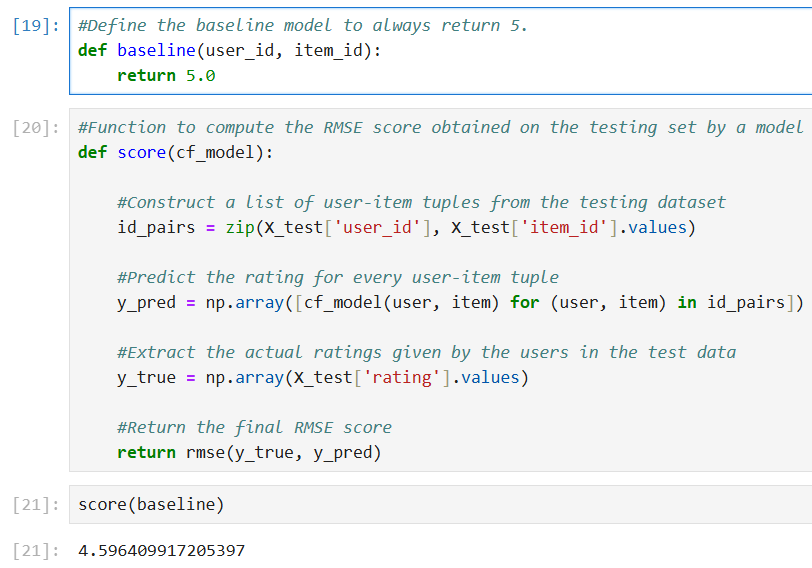


Figure - Baseline function

The baseline score is 4.5, therefore, any lower score is going to mean an improvement.

On this last example we used the ratings that each user gave to the item clothes.

The second approach is to use the mean rating of the items, however the score is the same, that means that our approach recommending items base in the mean score is not useful.

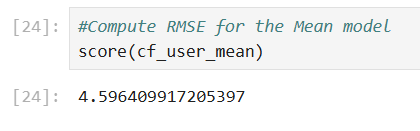


Figure - Mean score

We also used the weighted mean, on this case we fill with 0 the items that were not rated for certain users.

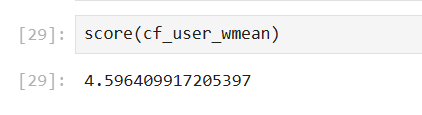


Figure - Score Weighted Mean

The score of weight mean is the same that means there is not improvement using a different approach.

#### Demographics

At this time we added other features as fit and body type of our customer.

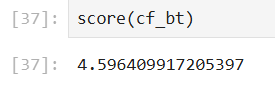


Figure - Body type score

Unfortunately there is not improvement using this features.

The second approach is , split the data and our dependable variable is going to be items.



Figure - - Data split, y is item\_id

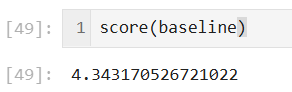


Figure - Baseline score using with Item as dependable variable

The baseline score 4.34, this is going to be the reference in our mean model.

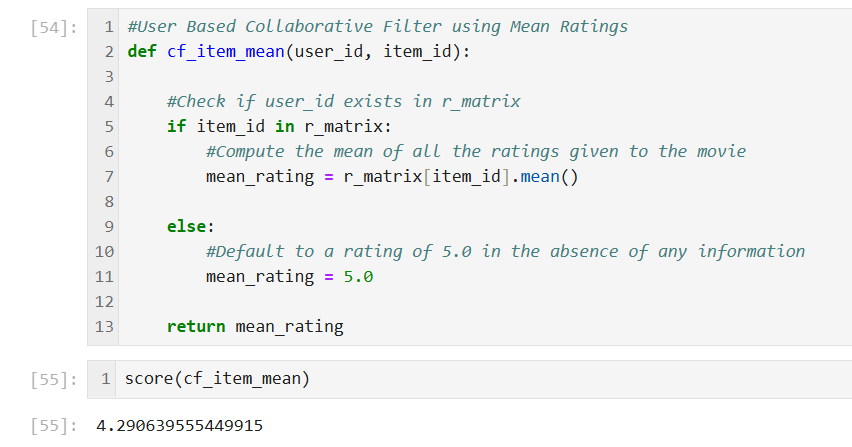


Figure - The mean score for item

At this time we obtained an score of 4.29, that means an improvement in our recommendation system.

**Conclusion**

It is better an approach using the item collaborative system. However is important to explore further our data. Also it is important highlight that our system can improve using more data, the data used was filtered based on recurrent customers.

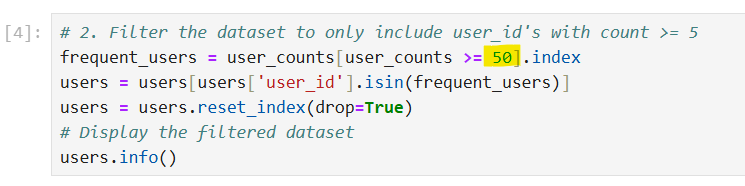


Figure - Frequent users 50 or more times

As you can see in the graph above the data was filtered based on this number to reduce the data and continue with the analysis, unfortunately it was not possible to apply a lower number due hardware limitations, we recommend to experiment with a lower value to improve the scores.

### Market Basket Analysis

Association analysis known also as market basket analysis, is a model algorithm usually used to find association between products, commonly used in supermarkets.

This is an unsupervised model, analyses the relation between independent variables to have a new output. The independent variables are categorical/ discrete instead numerical/continuous.

Association analysis starts finding combinations called frequent *itemsets*, this combination could be two or more items. After this you can find the support of each itemset, this number indicates how often these products appear together. Support is usually referred as ‘SUPP’ and itemset as ‘X’, and besides we have ‘T’ as the total of transactions or data points. For example if our itemset only appears two times and we have a total of 10 transactions, our formula is:

SUPP = X/T

SUPP = 2/10

SUPP= 0.2

Not all the transactions are useful for the analysis, thus we can discard the transactions with low support that are not relevant, to do this we need to set up a minimal support or minsup, for example to choose that only the items that appear in at least 40% of our transactions are relevant.

The next step is Rule Generation, with statements as *if* and *then,* and we can calculate the level of Confidence that measures how often the rule generated is true.

After corroborate that our data does not have null values, we created a list of transactions (rows) with the items bought.

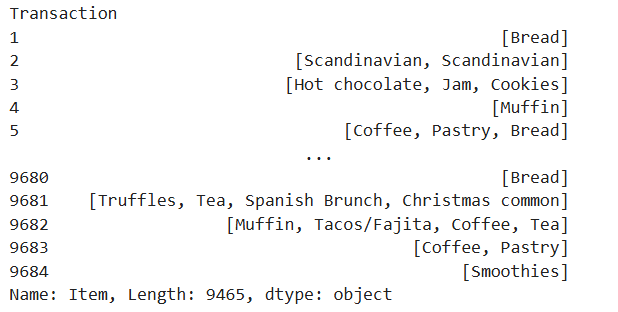


Figure - List of transactions for MBA.

From this dataset we did two different lists, one for the transactions made and other with the items that exist. With this information separated we construct another dataset, at this time each item is converted into a column and each row is stil the transaction (index). The dataset contains 94 columns which it means that we have a list of 94 items. The value True or False appears in our dataset and it means that the item exist in the row:

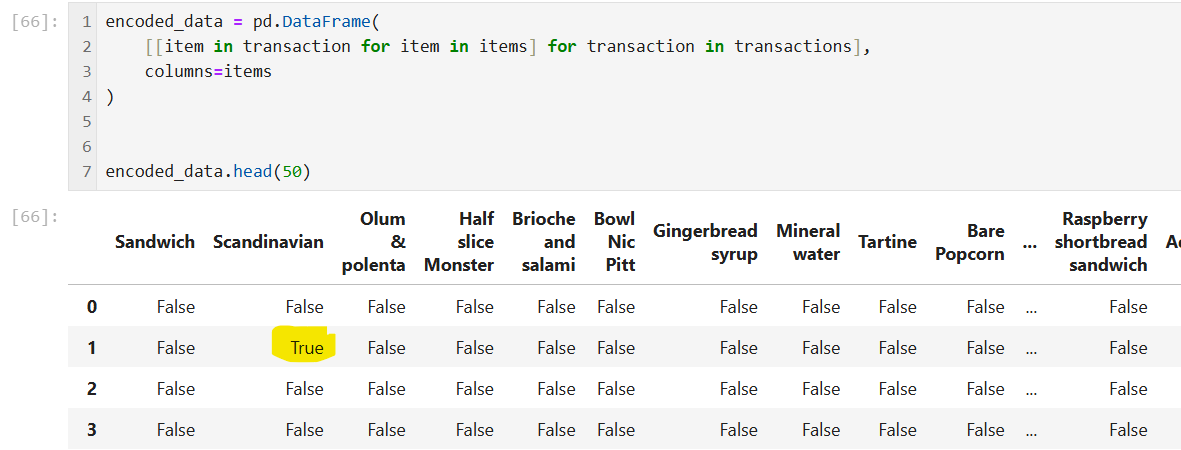


Figure - MBA Table

The transaction one shows True in the Scandinavian Column, this means that this item was sold. Due the transactions have between 2 or 3 items we have several False values compared to the True ones. This is common when we are using this system.

#### Apriori Algorithm

The first algorithm applied is Apriori, this model took 0.03 seconds to calculate our dataset. This algorithm gives priority to frequent items and subsets. The frequency of each individual item is computed and they have priority to build sets, assuming that a frequent item is in frequent sets. It is important to set the minimum support to run this association rule.

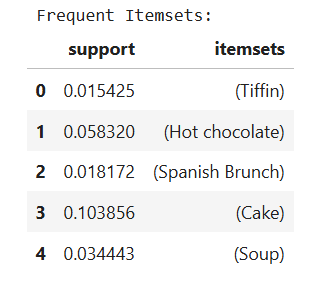


Figure - Items and their respective support

In the graph above we can see some items and their support, for example the cakes have a high demand compared with the other items, this information is useful for marketing campaigns or further analysis to improve our sales.

Apriori Algorithm can be slow if we have a dataset with a high volume of transactions,

#### FP Growth Algorithm - Frequent Pattern Growth Algorithm

This model is an improved version of the Apriori Algorithm, one drawback of the Apriori Model is that repeatedly scans the dataset, this models constructs a Frequent Pattern Tree (FP-tree) structure that help to use less computational resources. Each node represents an item.

In our analysis, the Frequent Itemsets were calculated in 0.75 seconds.

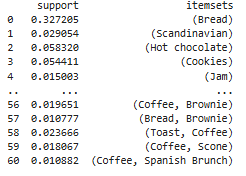


Figure - Results of FP Growth Algorithm

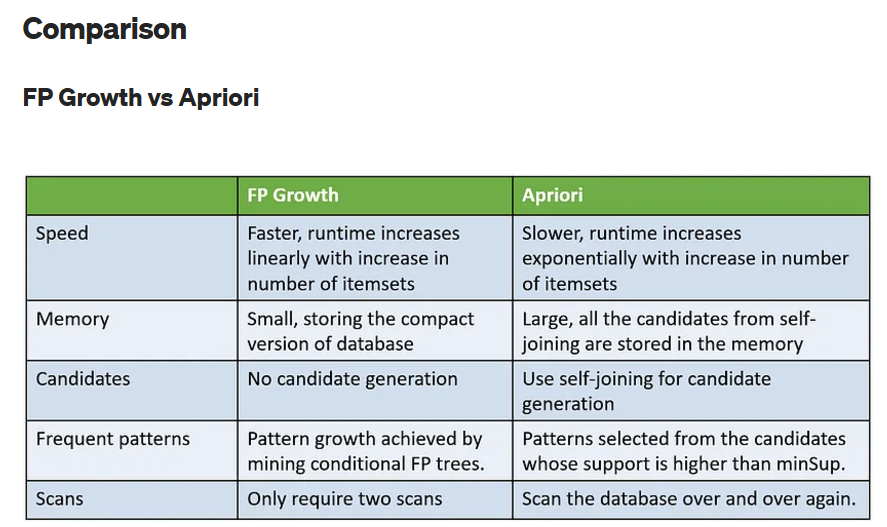


Figure - FP Growth vs Apriori Model - Extracted from Medium.com

## Conclusion

In recommendations systems is important to find a model that fits with our dataset contemplating the size of our data and also the hardware. Depending the quality of the data we can make better recommendations or predictions.

In some cases, it could be better to recommend and item based on the properties of the item instead the information about our users, like age or gender.

In the case of MBA, is important to have a high volume of transactions and this model in general that is simple it does not require information about our users or their past behaviour. It is really useful to follow our sales without to ask to the customer for further information because we can extract the data from the till.

# Data Visualization

The visualization of our data it was created with code extracted from plotly.com. This website contains code written in python with the option to create interactive applications.

The dashboard consists in one interactive scatter plot, one interactive bar chart and a simple bar chart.

The first graph gives the option to visualize the features weight, size and age for further analysis.

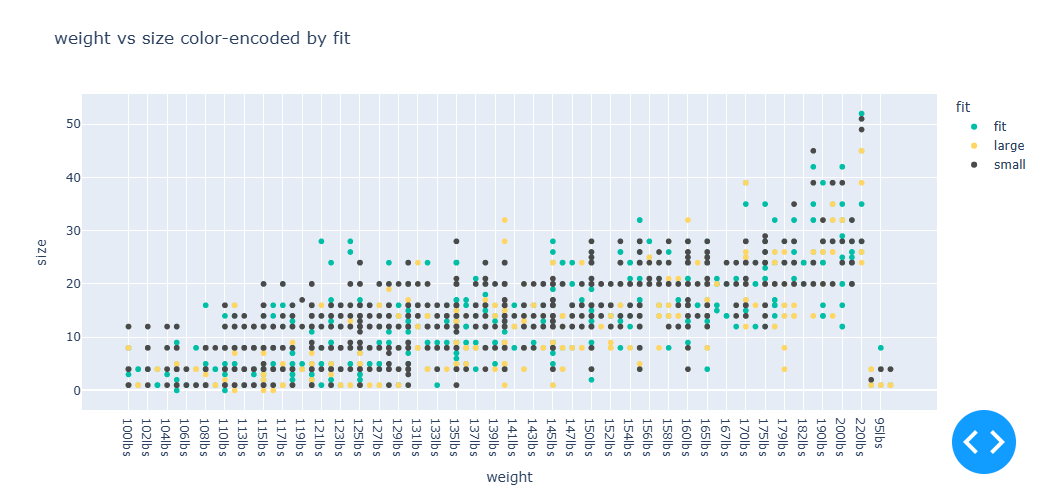


Figure - Interactive Scatter Plot

This features are continuous and numerical values and they are perfect for this type of visualization, with this graph it is easy to observe the relationship between size of the rented clothes and weight of our customer, this could help the store to enhance the type of clothes chosen for example.

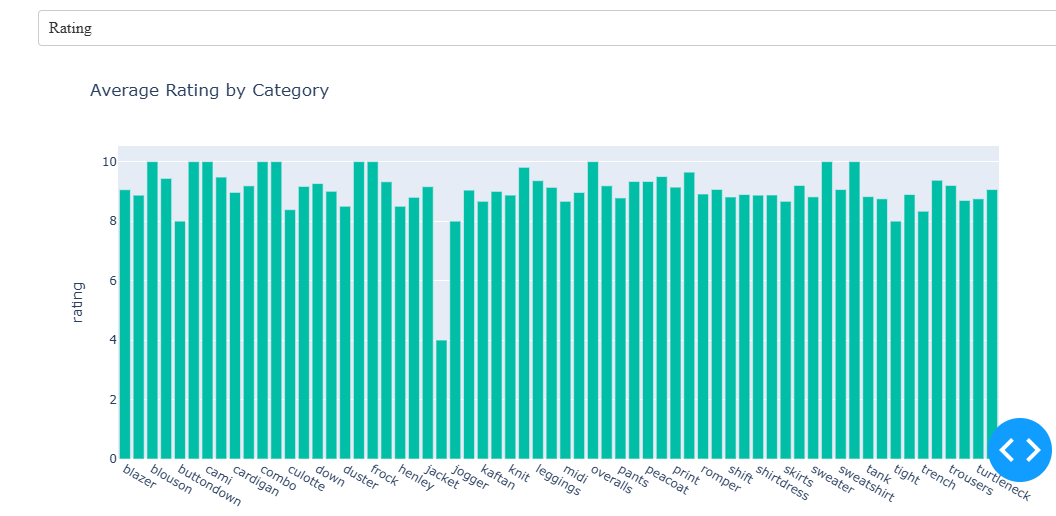
The second graph is an interactive graph bar that shows the average rating, age and size of each category of cloth.  
  


Figure - Interactive bar graph

This interactive bar chart help us to compare instantly the average of our clothes in different aspects, for example the jeans are the less liked type of item liked because has a lower average rating and the average age is 28 years old. This could help to the store to focus on this drawback and try to improve jean models for this age.

The third model is a barchart and is a count of the reason for the Rental, on this case the category “Everyday” is the most recurrent. Usually people does not rent clothes for vacations.

This could help the store to identify the items less rented based and improve the party category that is in third place.

It also gives more ideas for promotions or annual and montlhy plans.

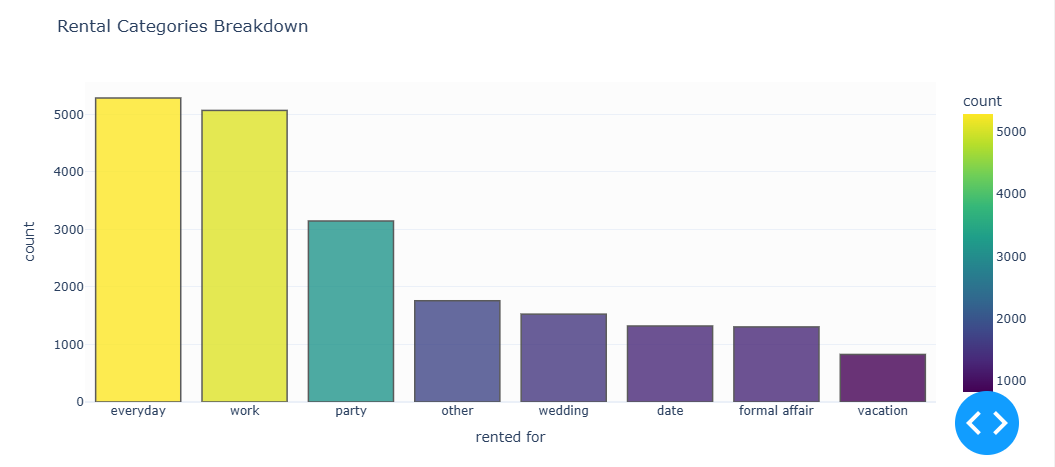


Figure - Category rented for

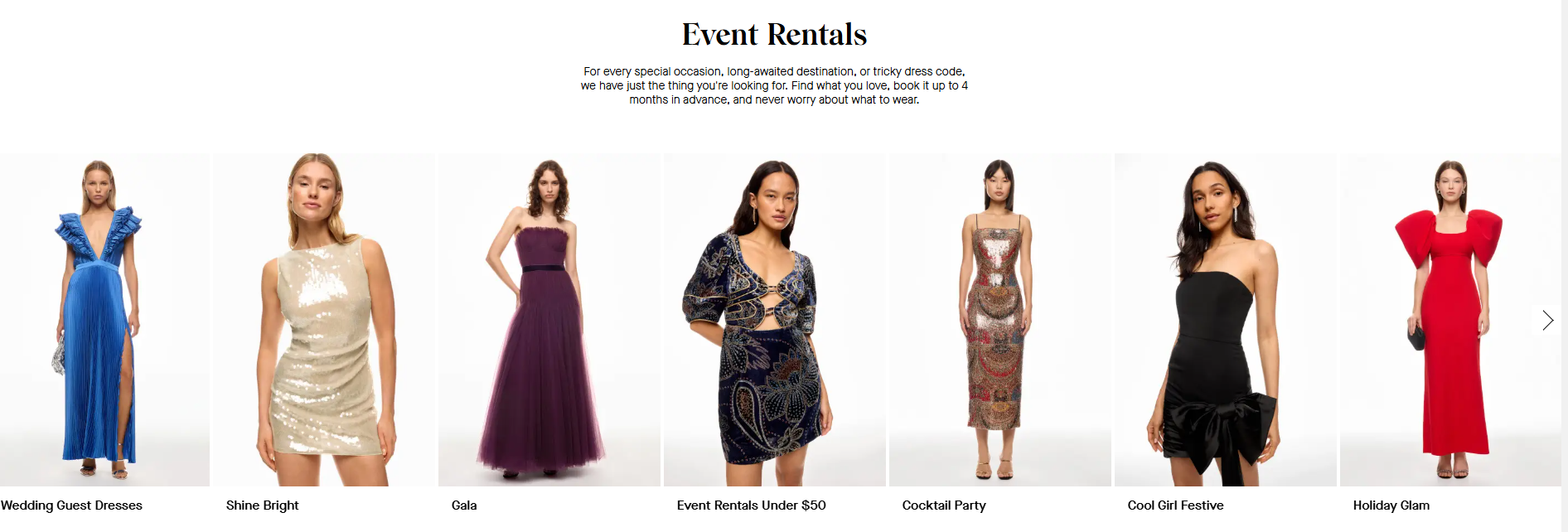


Figure - Screenshot from the website: https://www.renttherunway.com/

(70 marks)

The colors used for the visualization are considering a young audience between 18-35 years old, with vivid colors, orange for the title that is color related with energy and fun, the palette Viridis was used for the contrast and is a palette that contrast each category, for the scatter plot was used a version of color similar to Viridis but with less saturation. The objetive of this color is to keep the interest of the audience as well. The dashboard is very intuitive and is easy to explore.

This dataset is very interesting and has plenty of information and it is possible to apply several machine learning models, it is possible to apply clustering to create more categories based on the age, weight or size (as we can see in the scatterplot).

We can also use random forest to predict the items that could be choose based in the type of body, age and occasion when the cloth is used.

## Data preparation

The preparation of our dataset was first with the EDA eliminating the rows with null values. Also, a filter to show only the customers that rented 10 or more times the items was applied. This last one was based on the hardware limitation because our complete dataset was being slow to process.

After to apply an histogram to all the variables possible, we found some outliers in the age, with people registered with 116 and 117 years old, that is basically imposible. This rows were eliminated too.

For the bar charts it was created two datasets independent, for the interactive bar chart we did a dataset with four columns.

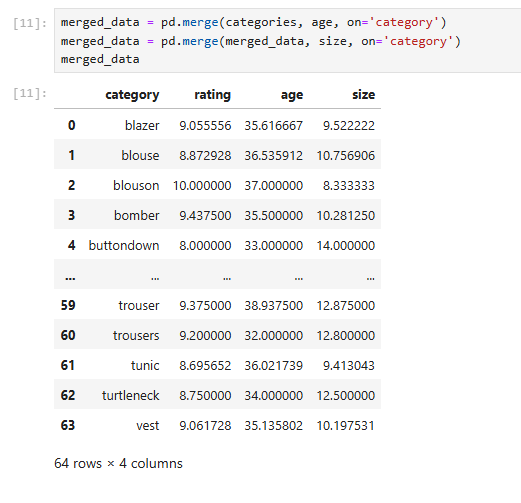


Figure - Dataset for interactive barchart

The second bar chart was created counting the category ‘rented for’

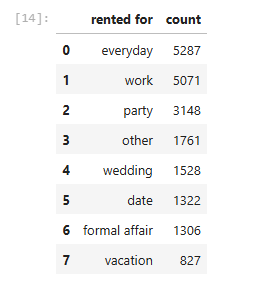


Figure - Dataser for barchart

The scatterplot was based on the same dataset and the features fit, weight,size and age were used.

To create the dashboar the module Dash was used, the first part of the code has individual charts, the second part is to create individual widgets, with dropdowns for the bar chart and scatter plot.

Finally, we created callbacks this helps us with the interaction of the graph and responds to any changes that we apply to the dahsboard, as dropdown’s selected values.

## Conclusion

The creation of a dashboard is an advanced level for data visualization, it is really useful for different business and can help to take important decisions. The difference is that static graphs are limited in observation and interaction with the user.

This dashboard created was very simple but it was useful to find correlations and also to compare different features and averages at the moment. Other advantage of this type of visualization is that can keep the attention of the users.

# References

Amberella Academy. “Recommender Systems and Baseline Models - Amberella Academy - Medium.” *Medium*, 27 June 2024, medium.com/@amberellaacademy/recommender-systems-and-baseline-models-e2f900b3d622. Accessed 29 Nov. 2024.

“Apriori Algorithm - Javatpoint.” *Www.javatpoint.com*, www.javatpoint.com/apriori-algorithm.

Chonyy. “FP Growth: Frequent Pattern Generation in Data Mining with Python Implementation.” *Medium*, 10 Nov. 2020, towardsdatascience.com/fp-growth-frequent-pattern-generation-in-data-mining-with-python-implementation-244e561ab1c3.

“Frequent Pattern Growth Algorithm.” *GeeksforGeeks*, 21 July 2019, www.geeksforgeeks.org/frequent-pattern-growth-algorithm/.

GeeksforGeeks. “Apriori Algorithm - GeeksforGeeks.” *GeeksforGeeks*, 4 Sept. 2018, www.geeksforgeeks.org/apriori-algorithm/.

*Https://Github.com/Nurlailiis/Association-Algorithm*.

“Modern Analytic Apps for the Enterprise - Plotly.” *Plotly.com*, plotly.com/.

Solanki, Sunny. “How to Create Basic Dashboard in Python with Widgets [Plotly & Dash]?” *Coderzcolumn.com*, CoderzColumn, 2020, coderzcolumn.com/tutorials/data-science/how-to-create-basic-dashboard-in-python-with-widgets-plotly-dash.

Suresh Kumar Mukhiya, and Usman Ahmed. *Hands-on Exploratory Data Analysis with Python : Perform EDA Techniques to Understand, Summarize, and Investigate Your Data*. Birmingham, Uk, Packt Publishing, 2020.

Theobald, Oliver. *Data Analytics for Absolut Beginners*. Second ed., Scatterplotpress.com, 2019.

**Github link: https://github.com/2024068/Integrated-CA2\_DVT\_MLB---Sept-2024**