**1. Business Understanding**

**1.1 Background**

H&M is a fast fashion brand that was first established in Sweden. The brand offers the latest style and the best prices to customers. H&M consists of inexpensive clothing necessities, perfect appearance accessories, and sportswear for women, men, teenagers, and children. It also includes living items from H&M Home. H&M operates 4420 stores, 72 store markets, and 49 online markets worldwide. H&M was considered as the second most valuable clothing brand in the world after Nike in 2018.

**1.2 Situation Assessment**

Every day, H&M handles large amounts of orders from its multiple sales channels, and product volume is tremendous. By giving accurate predictions on what items are best sellers and offering personalized recommendations, H&M increases sales and save inventory costs by optimizing supply chain management. It is a waste of money and resources to have inventories piled up in stores or warehouses and became outdated or deadstock. Providing satisfactory personalized recommendations is also important for H&M to retain and attract more customers and improve benefits.

**1.3 Models Objectives**

Here, we want to first analyze the datasets to develop models in helping H&M identify valuable customers and then use text analysis technique and association rules to provide personalized product recommendation which customers are most likely to buy in the future.

**1.4 Project Plan**

In general, our project will address two problems.

Problem1: Segment customers and define high-value customers; use machine learning models to find out important features for identifying potential high-value customers. Here various classification models will bed used.

Problem2: Develop algorithms to provide personalized article recommendations based on association rules and text analysis.

**1.4.1 Classification Models**

In first problem, we will define high-value customers to do customer segmentation and build classification models to help H&M to identify important predictors.

We calculate customers’ annual consumption in 2020 and define high-value customers as those with top 25% consumption, medium-value customers as those with middle 50% consumption and low-value customers as bottom 25%. (This segmentation is based on long-tail theory -- 20% of the customers contribute to 80% of total sales). Then we will use these customers’ transactions history data in 2019 to get attributes such as age, buying channels, and different amounts of items a person bought, and other customers features and behaviors as predictors. By building a classification model on these, we could use the customer’s previous purchase characteristics and buying behaviors to predict his/her value in the future.

H&M can use this list to do customers analysis. By analyzing attributes of customers, such as age, buying channel, etc., H&M can find out common characteristics of high-value customers and better target them in future marketing and strategy. For example, it could assign more marketing budget for high-value customers’ favorable buying channel; it can replenish its stock with more clothes high-value customers prefer; it can also get a high-value customers’ portrait.

**1.4.2 Association Models**

In personalized recommendation part, due to the large amount of data and our limited computer resources, we will use a subset of data in training our model, but models in this part are capable of being expanded to the whole dataset.

We will choose high-value customers’ transaction data in May 2020 and recommend similar products to what they bought in May as predictions of possible items they will buy in the near future. We use their real purchases in the next three months to test our predictions’ accuracy.

Association Rules Model treats customer’s purchases as the articles bundles and find out highly-related articles. By applying association rules, we sort pairs of items that are usually bought together for all the customers in HM and give top 10 highly-related articles as recommendations for each customer in May 2020 cohort.

**1.4.3 Text analysis**

The next model we use for giving predictions is text mining techniques TF-IDF. Article attributes in articles table tend to give a detailed description of articles and the information combined together can be viewed as a “document” describing this article. TF-IDF stands for term-frequency-inverse document frequency and quantifies the importance of relevance of string representations in the “document”. We use a document’s TF-IDF values for all words to represent this document and calculate similarity between document pairs. In this way, we can find out similar products to what customers already bought and recommend these to them.

**2. Data Understanding**

H&M provides us with 3 datasets, the articles.csv, the customers.csv, and the transaction.csv[[1]](#footnote-1).

**The Articles Dataset**

The articles Dataset includes all the items that H&M currently has. It consists of 105, 542 articles with 25 attributes.

图片包含 应用程序

描述已自动生成

表格

描述已自动生成 手机屏幕截图

描述已自动生成

Apart from some common attributes such as **article\_id**, **prod\_name** and **product\_code**, the dataset also provides us with some useful attributes such as the color, the perceived color, and the description of the article.

Among these attributes, most of them are categorical features. Their categories are divided based on H&M’s methodology, for example types, departments, sections, garment groups, and so on.

For example, **product\_type\_no** and **product\_type\_name** are two attributes that H&M uses to categorize the articles. H&M put the articles into different types and assign them a type series number. From the chart below, we can see that most of the articles are trousers and dress.

图表, 条形图

描述已自动生成

For another attribute, **index\_name**, is also similar.

图表, 条形图

描述已自动生成

**The Customers Dataset**

The Customers Dataset includes all the customer information, consisting of 1,371,980 customers with 7 attributes.

图片包含 图形用户界面

描述已自动生成

文本

描述已自动生成 文本

描述已自动生成

Also, there are some missing values for some attributes. Note that over 50% of values in FN and Active columns are missing here, which implies bad data quality, and we decide not to use them in our models.

文本

描述已自动生成

For **club\_member\_status**, this is an attribute to show the status of the customers. It has 3 unique values, ACTIVE, LEFT CLUB, and PRE-CREATED. We can see most of these customers are active members and a few of them are at other statuses. We believe this attribute has little effect on the model, so we decide to drop it.

图表, 瀑布图

描述已自动生成

The **fashion\_news\_frequency** describes the frequency of the customers receiving the fashion news. We can see the distribution as the following:

图表, 瀑布图

描述已自动生成

Most of customers have no habit of receiving fashion news.

Moreover, the dataset also provides the age of the customers. Most of these customers are between 20 and 30 years old. The maximum age could reach over 80. The median age is 32 years old.

图表, 直方图

描述已自动生成

**The Transaction Dataset**

The Transaction Dataset includes the transaction history from 2018-09-20 to 2020-09-20. It consists of 31,788,324 transactions with 5 attributes. Each instance represents a transaction of a customer for a single article.

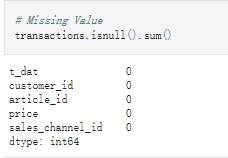
文本

描述已自动生成

图形用户界面, 文本

描述已自动生成

There is no missing value for this dataset, which is good.



图表, 直方图

描述已自动生成

We can see some patterns in the line chart. We have a huge volume of transactions happening in October 2019.

图表, 条形图

描述已自动生成

If we just look at the monthly distribution, most transactions occurred in June and July and few in winter and spring.

图表, 条形图

描述已自动生成

We have the highest volume of transactions on Saturday. For Monday, Tuesday and Sunday, the sales are lower. We can arrange our employee schedule based on this. Also, for Sunday, the management team should go to discover the reason why Sunday, a non-working day, has this few numbers of transactions.

**3. Data Preparation**

**3.1 Divide the Transaction into 2 period**

We are going to build a model to predict the future importance of a customer. If we want to use many X to predict Y. We need to collect our X, the customers’ buying behavior, and feature attributes from 2019. We get the Y, our target variable, from the data for 2020. Thus, first, we need to divide the transaction dataset into two time periods.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

We split the transaction in the point of 2019-9-20.

**3.2 Merge the Transactions and Articles Dataset**

We are going to combine the transaction data with articles data and keep the attributes that are useful for our models.

There are many attributes describing the category of the article, eventually, we pick the most distinct and meaningful attributes as shown below.

图片包含 表格

描述已自动生成

**3.3 Modify the fashinon\_news\_frequency NONE value**

To improve the classification models’ performance, we decide to turn the string description into numbers. 0 represents NONE, 1 represents Monthly, and 2 represents the other circumstances.

图片包含 文本

描述已自动生成

**3.4 Drop useless attributes**

Some attributes, because they have many null values and some of them are hard to analyze and interpret, such as the encrypted postal code. We decided to drop them.

图形用户界面, 文本, 应用程序

描述已自动生成

After dropping, the Customers Dataset is shown below:

表格

中度可信度描述已自动生成

**3.5 Fill null value in Age with the median**

Age has many null values, we decided to fill it with the median.

图片包含 图示

描述已自动生成

**3.6 Merge Customers Dataset with the merged Transactions\_Artibles Dataset**

图形用户界面

低可信度描述已自动生成

**3.7 Turn nominal attributes into dummy variables.**

As mentioned above, we have some attributes about the type and the color and some other nominal attributes in the current dataset. We decide to make them into dummy variables for our next step.

图形用户界面, 文本, 应用程序

描述已自动生成

文本

描述已自动生成

**3.8 Select the customers with more than 5 transactions**

In order to get meaningful insight, we decided to concentrate on the customers who had more than 5 transactions. We don’t want some outlier or noise in our dataset.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

**3.9 Group the dataset**

In order to calculate the difference and preference of each customer, we can sum up the 1 in a column. If a customer often buys a certain type or a certain color of a product, we will know.

图形用户界面, 文本

中度可信度描述已自动生成

Group some attributes that do not need to sum up.

文本

描述已自动生成

Group some attributes that need to sum up.

图形用户界面, 文本

中度可信度描述已自动生成

表格

描述已自动生成

Combine the sum and the non-sum-up data frame.

表格

中度可信度描述已自动生成

**3.10 Create the target variable based on total consumption.**

We are going to create our target variable, the importance of a customer based on his/her next year’s total consumption at H&M.

First, we sum the price of all the transactions for customers as total consumption.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Then, we could see the total consumption (the sum of price in 2020) distribution.

图形用户界面, 文本, 应用程序

描述已自动生成



We decide to divide them into 3 groups by using the 25% and the 75% percentile as our split points. Therefore, the top 25% of customers will be labeled as 2 - the valuable customer. The last 25% will be labeled as 0 – less valuable, and the rest will be 1 – normal customer,

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Finally, we merge this column into our previous dataset.

表格

描述已自动生成

**4. Data mining and Evaluation**

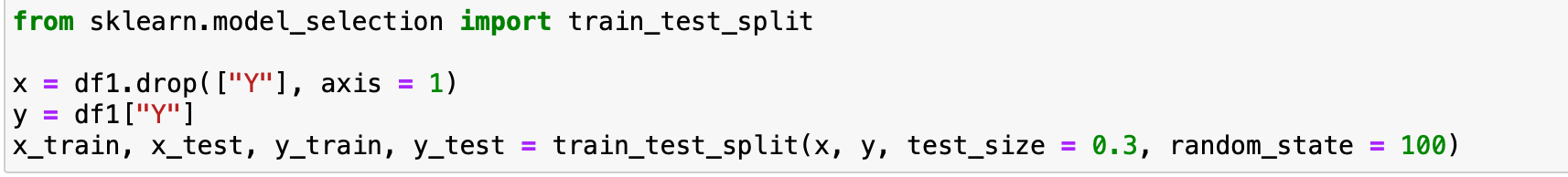
**4.1 Classification Modeling**

After the data cleaning and exploration, we have finalized the dataset and firstly we decide to use the classification modeling to find the most and least valuable customers. In this dataset, we have used the data of 2020 to classify our customers into 3 groups in which the value 2 corresponds to the most valuable customers and the value 2 corresponds to the least valuable ones. And we use the data of 2019 to do the classification modeling.

Considering that we have a large number of independent variables and a huge dataset, we decide not to use the support vector machine and KNN algorithms. Instead, we decide to try 3 classification algorithms: naïve bayes, decision tree and random forest, and select the best result one for our client. And before coding the algorithms in Python, we decide to drop the customer\_id column out of the independent variables and preprocess both the dependent and independent variables. Then we split the data into 70% train and 30% test sets.

表格

描述已自动生成



* + 1. **Gaussian Naive Bayes**

We first try the Gaussian Naïve Bayes model using scikit-learn package. After fitting the model and generating the prediction, we then calculate its accuracy and generate the confusion matrix for the model.

文本

描述已自动生成

文本

描述已自动生成

图表, 树状图

描述已自动生成

As the result showing above, we only get around 31% accuracy for the gaussion naïve bayes models.

* + 1. **Decision Tree**

Next, we try the decision tree model. After fitting the model and generating the prediction, we then calculate its accuracy and generate the confusion matrix for the model.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

As the result showing above, we get a better accuracy than naïve bayes model which is around 51%.

图表, 树状图

描述已自动生成

* + 1. **Random Forest**

Finally, we try the random forest model. We find the max\_depth = 5 yields to the great result. Like always, after fitting the model and generating the prediction, we then calculate its accuracy and generate the confusion matrix for the model.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

图表, 条形图

描述已自动生成

It turns out that the random forest model performs the best among the 3 models. It has around 63% overall accuracy.

Because the main goal of the classification modeling is to help the client find the most valuable customers, compared to the rest 2 groups, the value 2 group is the “important” group that we care about most.

To focus on the value 2 group, we deeply analyze the confusion matrices of the 3 models.

**For Gaussian Naïve Bayes:**

Sensitivity = 22978 / (26876 + 12553 +22978) = 37%

Specificity = (8732+15533) / (15533+1447+1202+56536+8732+7284) = 27%

**For decision tree model:**

Sensitivity = 35326 / (4276 + 22805 + 35326) = 57%

Specificity = (39577+3279) / (3279+10592+4311+10359+39577+22616) = 47%

**For random forest model:**

Sensitivity = 36127 / (36127+26280) = 58%

Specificity = 60296 / (16762 + 60296 + 1420 + 12256) = 67%

In this perspective, we find it that decision tree model and the random forest has the similar sensitivity to detect the important group members. But the random forest model is more likely to rule out the other 2 group members.

Another aspect for model evaluation is to check how fast the model could capture the essential information, in our case is to find the most valuable customers. We decide to use the lift chart to compare among the 3 models.

图表, 折线图

描述已自动生成**For Gaussian Naïve Bayes:**

**For decision tree model:**

图表, 折线图

描述已自动生成

**For random forest model:**

图片包含 图表

描述已自动生成

Based on the results of lift charts, we find it that the random forest model is the one capture the most valuable customers quickest.

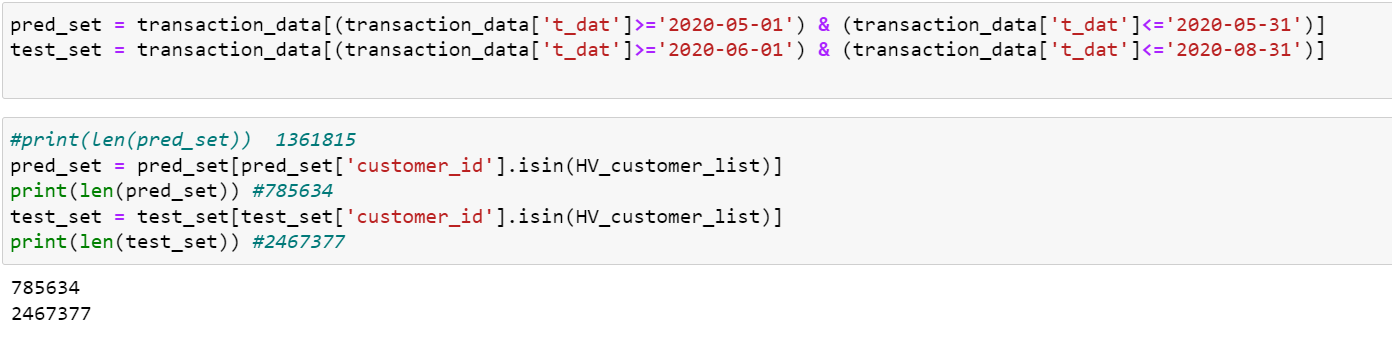
To conclude, after the model evaluation, we decide to choose the random forest model to classify the customers to find the most valuable ones.

**4.2 Text Mining Technique – TF-IDF**

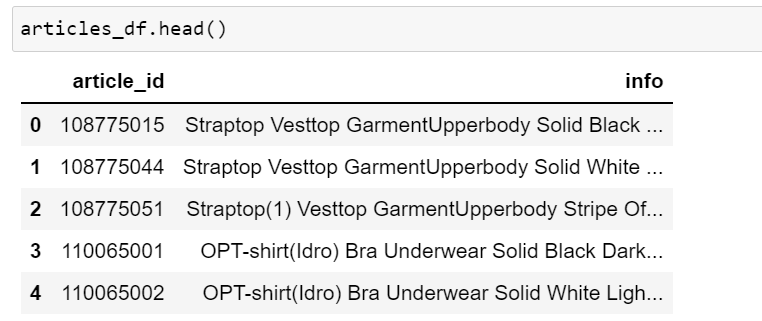
To give recommendations, we have two approaches: first is to recommend similar products to what a customer already bought, based on the assumption that the customer has a certain purchase preference pattern; second is to recommend what similar customers bought to a customer, based on the assumption that similar customers share common purchase preference pattern. TF-IDF here is the first approach. We consider articles’ attributes as the “document” representing articles, and get TF-IDF values for all words in article “documents” to calculate similarity between articles. In this way, we successfully recommend similar products to customers.

**4.2.1 Data Preparation**

Due to large volume of articles and transaction data and limited computer resource, we select customers identified as high-value in 2020 as our customer list and select their transaction data in 2020-05 to make predictions and test the prediction accuracy from June to August.



As for articles table, there are 105542 articles in total with 25 attributes. It is quite difficult for our computer to calculate similarity between all articles pair. Here we will only select 8000 of them and treat these 8000 articles as the article basket. To create the document for these 8000 articles, we combine its prod\_name, product\_type\_name, product\_group\_name, graphical\_appearance\_name, colour\_group\_name, perceived\_colour\_value\_name, perceived\_colour\_master\_name, department\_name, index\_name, index\_group\_name, section\_name, garment\_group\_name and detail\_desc as a string. The result is as below.



**4.2.2 Model Training**

Then we calculate TF-IDF values for these articles and get the similarities for each article pairs. We find out the five most similar articles for each article and define this as a function to give recommendations. To check if we successfully get the similar articles for each article, we used images in the H&M datasets and check some articles.

We selected three articles in the list and used the algorithm to find its most similar 5 products and the results are as below. We can see that it works pretty well.



Based on the results below, we recommend similar products in this 8000-article basket to what they bought in May and form a prediction list of articles.

**4.2.3 Model Accuracy**

By comparing our predictions to what customers actually bought in the next three months, the accuracy in the articles side: we give 123322 recommendations in total. Among them, 815 are correct, 122507 are incorrect. The accuracy rate is 0.66%.

From the customer side: we give recommendations for 19477 customers. Among them, 719 are correct and 18758 are incorrect. The accuracy rate is 3.69%.

**4.3 Association Rules**

We applied association rules to the dataset and give recommendations for customers.

**4.3.1 Reason for applying association rules**

By exploring the data, we found that many customers bought more than 1 articles at a time and we believe that there may be some relations between articles.

The simplest example is the association between a holiday dress and a straw hat. We may find that when a customer buys a holiday dress, he/she is likely to buy a straw hat through the purchase records of a large number of customers in the past. Therefore, it would be a meaningful recommendation to recommend a straw hat to a customer who just bought a white holiday dress recently.

By applying association rules, we can figure out the relation between articles and recommend articles to customers that they are most likely to buy.

**4.3.2 Choice of parameters**

Customer id and article id are the main parameters to analyze the problem.

One traditional case for the association rule is in 1990s, Walmart found that beer and diapers, two completely insignificant items, have a high probability of being purchased together. Therefore, to analyze the problem, we cannot presuppose possible outcomes from experience, sometimes unexpected results occur.

In this case, we choose customer id to help filter out some low-quality users and only keep the customers who have high potentials to buy articles in HM. Articles ids are grouped by customer id and date parameters to find out what articles are bought together by a customer at a time.

**4.3.3 Model Accuracy**

Analyze the problem from the total recommended articles side: we give 542155 recommendations in total. Among them, 9669 are correct, 532486 are incorrect. The accuracy rate is 1.8%.

Analyze the problem from the customer side: we give recommendations for 72515 people, among them, 7228 have bought at least one articles from the recommendations we gave them, the accuracy rate is 10.0%.

For a recommender system, we have a considerable prediction accuracy by applying association rule.

**5. Analysis of Models:**

**5.1 Classification Models**

Based on the modeling results, we finally decide to use the decision tree model to classify our customers into 3 groups with different purchasing powers. The results of the model indicate that around 60% customers follow the same pattern on spending money for H&M products.

**图表, 条形图

描述已自动生成**

And around 40% customers are classified as the most valuable customers (labeled with value 2). For the group of the most valuable customers, H&M decision makers may consider make the marketing team to push ads of top gear products, send new products emails in the first places and may design customized products or gifts for certain shopping seasons. For those customers, it’s important to keep their loyalty and generate more profits from them. On the other hand, for those one-time or new customers (labeled with value 0), they should consider attracting them to place at least a new order through sending promotion codes for instance.

And another thought is to expand our modeling by make models to recommend the “right” products for “right” customers.

**5.2 Text Mining Technique – TF-IDF**

**5.2.1 General Description**

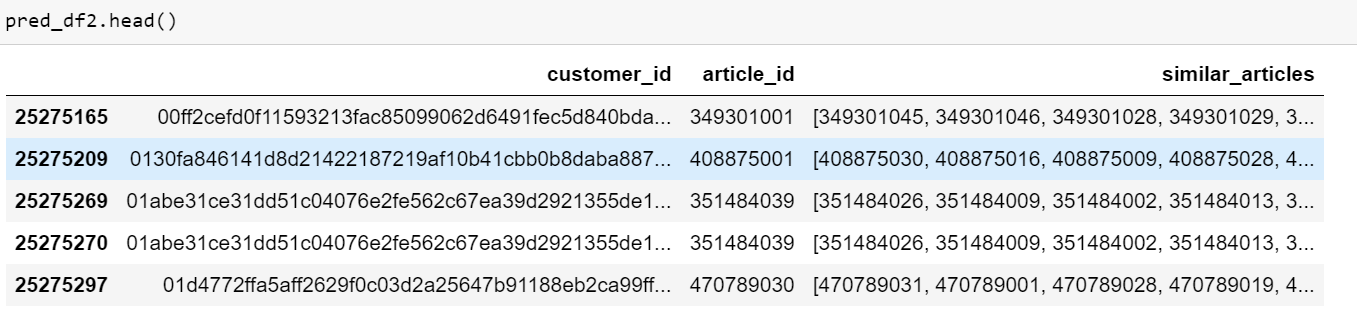
We use a subset of 8000 articles as the article basket to calculate article pair’s similarity. As for transaction data, we select transactions data in May 2020 as the training data and provide similar articles to what they bought in May. Here we only select the high-value customers, those tagged as 2. By sorting the similarity score for each article, top 5 similar products become recommendations to customers. The testing dataset are the same group of people who have transactions in the following 3 months (June, July, August 2020).

**5.2.2 Results in Detail**

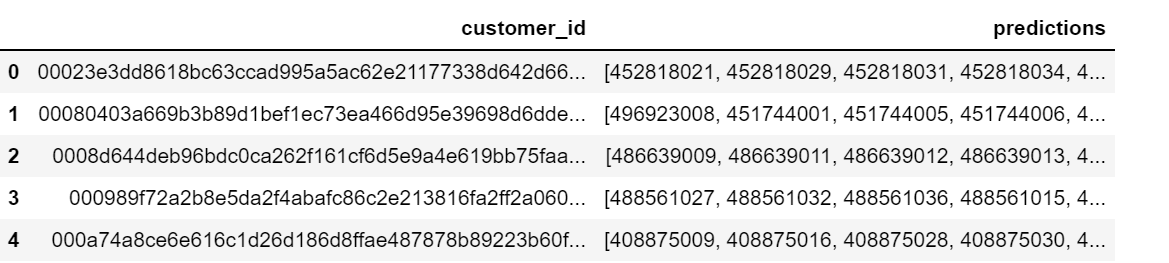
The articles and their “documents” are like below.



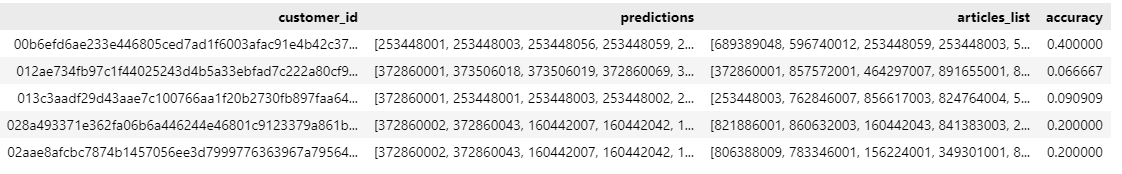
The recommendations we provided for each customer are as below.



Since our recommendation is provided per customer, then we aggregate the result by customer.



After that, we compare our predictions with actual articles they bought and calculate the accuracy for each customer.



**5.2.3 Future Improvements**

We can see from above results that the accuracy is not quite satisfying. From my perspective, I think this could due to the following reasons:

- Here we only select a subset of 8000 articles in our basket to calculate article pair similarity. If we select the whole basket, the top 5 similar articles may be different; In addition, these customers did not buy articles in this basket for the next three months, but they may have bought other similar products to what they bought in May and we did not cover those in our basket. If the similarity algorithm is expanded to the entire dataset, the accuracy will be much likely to lift.

- Maybe one-month data is not enough to capture the purchase preference and three months is not enough for test the accuracy. In real life, H&M may use the past several months purchase history for giving recommendations and we need to try different test period to find out the most proper test period.

**5.3 Association Rules**

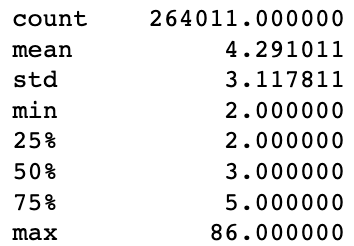
**5.3.1 General Description**

We use transactions data in May 2020 as the training data to figure out the association rules between pairs of articles. By sorting the confidence score by pairs of articles, recommendations are given for the customers. The testing dataset are the people who have transaction history in May 2020 and in either the following 3 months (June, July, August 2020). People who have low intends tag in HM are also filtered out.

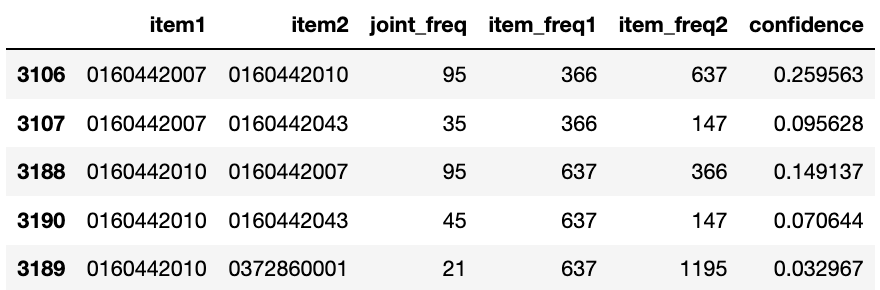
**5.3.2 Results in Detail**

Use data from “transactions\_train.csv” where date range from 2020-05-01 to 2020-05-31 as training data for the association rules.

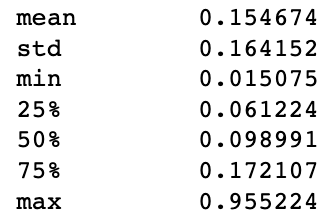
Extract all transactions where people bought at least two items a time as the original table. Descriptive statistics for origin table：



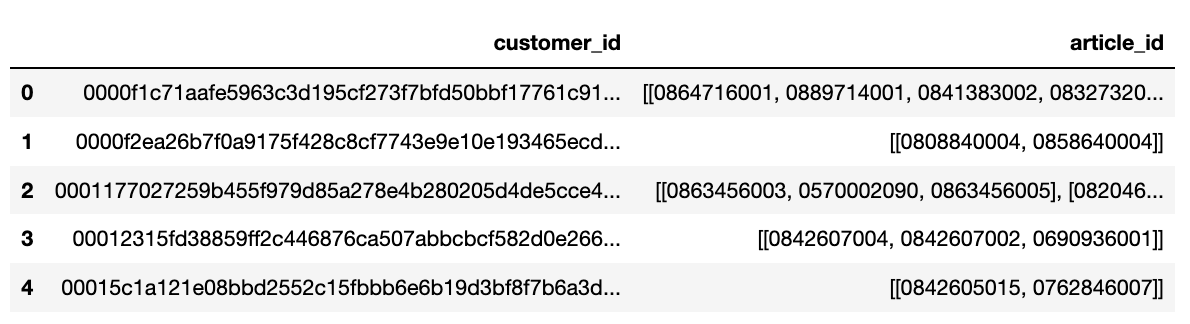
Construct a frequency table for pairs of articles bought in May 2020. Calculate the joint frequency and confidence to evaluate the association between articles. The frequency table is as following:



Descriptive statistics for confidence in origin table：



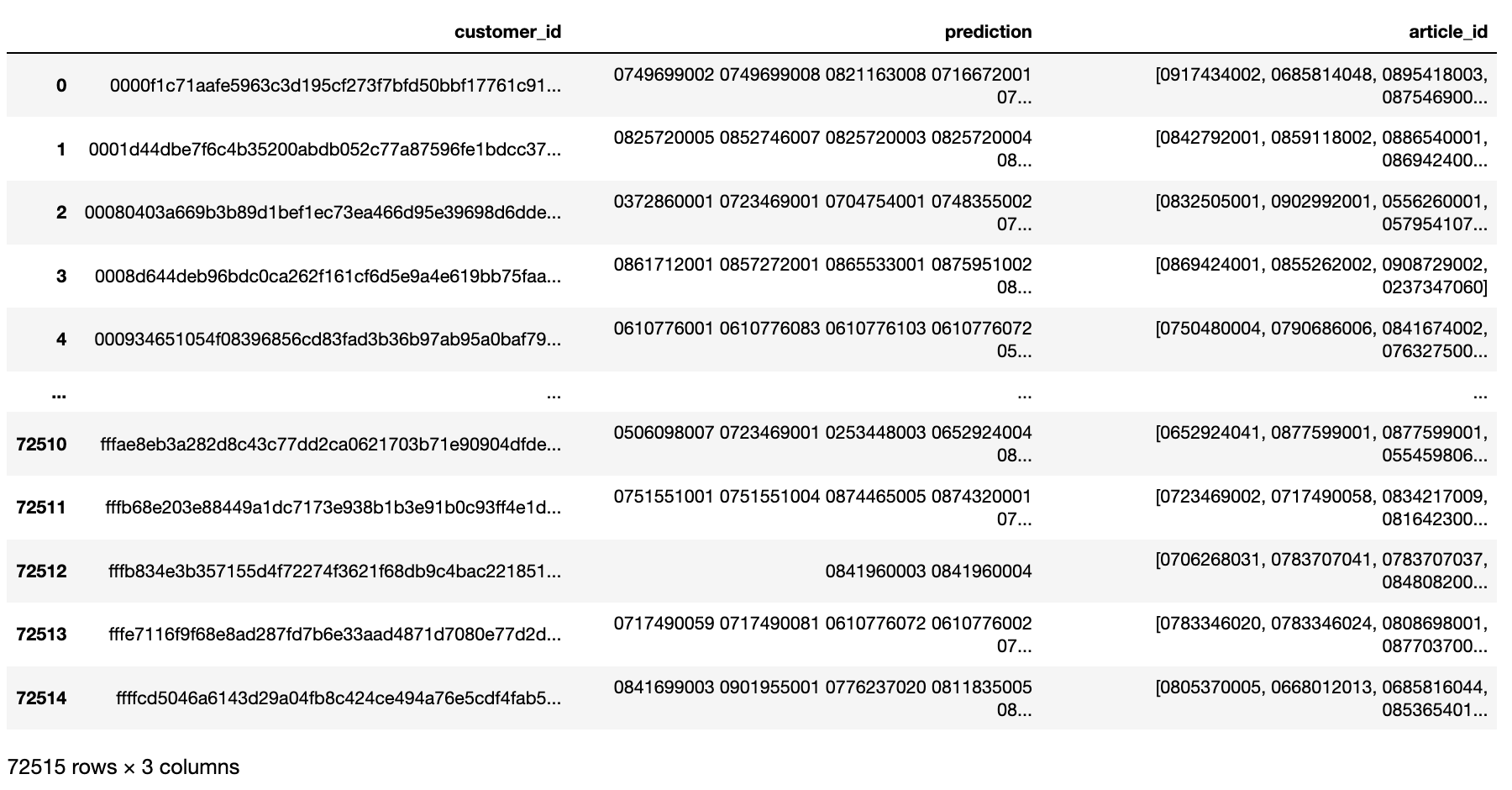
For the prediction part, we first construct a transaction table where date range from 2020-06-01 to 2020-08-31.



Then we inner join the transaction table with original table to get the customers who bought article in May 2020 and also in either the following 3 months of 2020. We also use the tags we get when clustering the people to filter out the low potential customers. Thus, we get a new customer list for prediction.

For these customers, we get their articles bought in May and extract all the association articles from the frequency table. In this step, we limit the association articles by setting the joint frequency > 20 (items1 and items2 are bought more than 20 times together by a same customer in May 2020). Finally, we sort the related articles by sorting confidence value from the highest to the lowest. For each customer, we give at most 10 recommendations.

We then get a prediction table as following:



For each customer, we give at most 10 predictions and extract what they actually buy in June, July, August 2020 for comparison (article\_id in the table). There are 72515 people in total that we give recommendations.

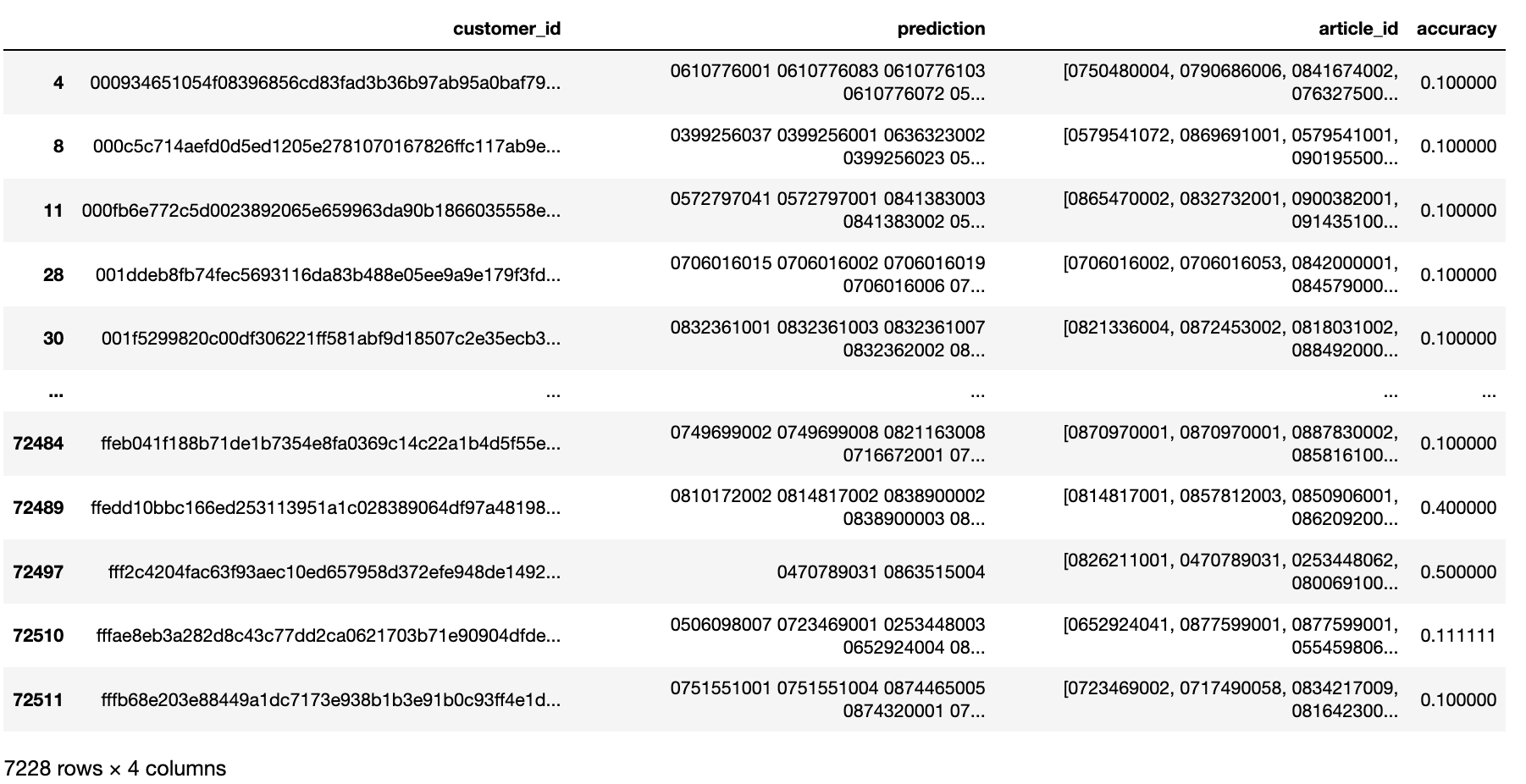
To evaluate the model, we calculate the accuracy of each predictions we give. In total, we give 542155 predictions and 1.8% of them are correct.

We also calculate the accuracy for each customer:

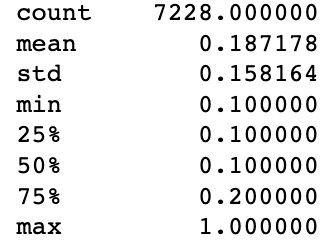


For all the 72515 customers whom we give recommendations, 7228 (10%) of them bought at least 1 article we recommended to them.

Construct a predict correct table for all the customers whom we have at least 1 correct prediction.



Descriptive statistics for accuracy in predict correct table:



**5.3.3 Business meaning of the result**

The business meaning behind the result is if we predict 10 items to each high-potential customers, around 10% of the customers will actually buy at least 1 item from the recommendations, which is a really high prediction accuracy rate. The result can give HM the confidence that reasonable personalized recommendations are useful Therefore, in reality, if we actually push the 10 items’ product information to the customers by front page or advertisements, we may probably get a more than 10% accuracy rate. In this case, HM customers can benefit from reducing searching time to get what they want and have a better customer experience. HM can also make front pages meaningful and earn profits from directly giving good recommendations to customer.

**6. Recommendations:**

For the group of the most valuable customers, H&M decision makers may consider make the marketing team to push ads of top gear products, send new products emails in the first places and may design customized products or gifts for certain shopping seasons. For those customers, it’s important to keep their loyalty and generate more profits from them. On the other hand, for those one-time or new customers (labeled with value 0), they should consider attracting them to place at least a new order through sending promotion codes for instance.

As for personalized recommendation, according to our previous results, association rule performs better. In fact, there are several methods for us to provide personalized recommendations and here we only provide two of them. To offer more accurate predictions, we can ensemble these algorithms using majority voting and find out articles that appear frequently in our predictions. By dynamically scrolling every week or month, we get a dynamic personalized recommendation for each customer based on their previous purchases.

After getting accurate predictions, the recommendation presenting methods may include：

1. Pushing the predicted product on the homepage；

2. Using the predicted product as an opening advertisement on the website;

3. Display our recommendations after customers finished a transaction and made the payment.

At the same time, by analyzing the most related products, HM can also use business strategies such as bundling sales to promote customer purchases (For example, if customers purchase certain types of goods at the same time, they can have discounts).

1. <https://www.kaggle.com/competitions/h-and-m-personalized-fashion-recommendations/data> [↑](#footnote-ref-1)