**Capstone Project**

**Create a Customer Segmentation Report for Arvato Financial Services**

**Sarah Dalton**

**Project Definition**

In this project, demographics data for customers of a mail-order sales company in Germany will be analysed, comparing it against demographics information for the general population. Unsupervised learning techniques will be used to perform customer segmentation, identifying the parts of the population that best describe the core customer base of the company. Then, a third dataset with demographics information for targets of a marketing campaign for the company will be used, and a model to predict which individuals are most likely to convert into becoming customers for the company. The used has been provided by Udacity’s partners at Bertelsmann Arvato Analytics and represents a real-life data science task.

The data sets used in this project are:

* Udacity\_AZDIAS\_052018.csv: Demographics data for the general population of Germany; 891 211 persons (rows) x 366 features (columns).
* Udacity\_CUSTOMERS\_052018.csv: Demographics data for customers of a mail-order company; 191 652 persons (rows) x 369 features (columns).
* Udacity\_MAILOUT\_052018\_TRAIN.csv: Demographics data for individuals who were targets of a marketing campaign; 42 982 persons (rows) x 367 (columns).
* Udacity\_MAILOUT\_052018\_TEST.csv: Demographics data for individuals who were targets of a marketing campaign; 42 833 persons (rows) x 366 (columns)

The problem which needs to be solved is a customer segmentation which classifies customers into groups of those likely to be converted into becoming customers and those who are not likely. This will help the company decide on which customers to target in their marketing campaign.

To do this I plan to:

1. Assess the customer and azdias datasets to identify what cleaning needs to be done to the datasets to prepare them for analysis.
2. Clean the datasets in the way identified in step 1.
3. Analyse the datasets using PCA and k-means clustering to identify the parts of the population that best describe the core customer base of the company.
4. Pre-process the training dataset by cleaning it and standardising the values in it.
5. Use classifiers to model the data.
6. Test the classifiers and compare MAE and accuracy scores to decide the best model.
7. Find best model parameters through tuning.
8. Apply best model to test data and generate predictions.
9. Find the set of customers that are likely to become customers.

I will be using the accuracy as the metrics to measure my model’s performance as it measures how often the classifier correctly predicts, which is an important factor of a classification model.

**Exploring the data**

I looked at the Excel documents (./DIAS Information Levels - Attributes 2017.xlsx and (./DIAS Attributes - Values 2017.xlsx) to get an idea of the data that was in the datasets. I noticed that -1 is used where the value is unknown, essentially that means that it’s a missing value. The same with 0 in other columns, although 0 shouldn’t have an impact on any aggregations that I may do on the columns, whereas the -1 will. There are also some columns that use 9 as an unknown value, but others that use 9 as a value. I had to consider what I was going to with these values so I decided to start by looking at the counts of these to find out how many were in each column of the datasets.

Looking at the data, there were different data types for the columns: a mixture of floats, integers, and objects. From looking at the Excel file, there did not appear to be any reason why the columns were a mixture of floats and integers. I could see that there was a column that had values which were a mixture of numbers and letters so I could understand why this column could not be a float or an integer type. However, there were two columns that only contained numbers but was saved as an object type, so I would have to convert this column to make it easier in the analysis part of the project.

When having a look at the missing values in the datasets, there appeared to be some columns where the majority of data was missing. This is something that I would need to consider, how would I deal with these missing values. I had a look at the Excel files to find what the columns were and the ALTER\_KIND column was not in these documents. Therefore, I decided to explore these more to see what values were in them and what percentage of the column was missing. I discovered that alter kind in German means child age in English. Therefore, I assumed that these columns were the ages of the children in the household, which explained why there were missing values as not everybody would have children living in the household. As this data may have provided use for the model, I decided to keep these columns for now.

As the dataset was so large. I decided to break it up into smaller dataframes based on the information levels in the DIAS Information Levels - Attributes 2017 file so I could explore them in more detail.

With the dataframes separated, I decided to look at histograms of the new dataframes so that I could look at how the values were distributed in each column. This would allow me to see if I could drop any of the columns if there was a lot of just one value, as this would mean that column of values would have little effect on the model.

Looking at the histograms of the person\_azdias dataframe, AGER\_TYP has a lot of -1 values which is 'unknown'. Therefore, I feel like this column would not contribute much to the model so I decided to drop it. The LNR is the person id so was expected to have a uniform distribution. The SOHO\_KZ and TITEL\_KZ have very little 1 values so, again, I decided to drop these as they wouldn't contribute much to the model.

From all the histograms of the dataframes, I chose which columns to drop based on the distribution of values.

A group of blue bars

Description automatically generated

Example of a set of histograms produced in the data exploration.

I also looked at the statistics from each dataframe to identify whether there were any obvious outliers. The only 2 that I thought may have contained outliers were ANZ\_PERSONEN (number of people in the household) and ANZ\_HAUSHALTE\_AKTIV (number of households in the building). When looking at these columns, although some of the numbers were high, it is reasonable to assume that number of people in the household could be high as it may be apartment blocks. The same for number of households in the building. Therefore, I decided to leave these values in.

**Cleaning**

When cleaning the dataframes, I found that the set was too big to perform most of the standard functions and the kernel kept dying. Therefore, I decided to complete the project using a sample of the azdias dataset. Although this may mean that the results are not as accurate as they would be with the whole dataset, using a random sample should not affect it too much and it allowed me to complete the project with very little issue.

I created functions that would clean each of the data sets by doing the following:

* change the object columns in status to integers.
* drop the columns that I found were not needed.
* remove any duplicate rows.
* change the columns to integers.
* fill missing values with 0.
* change the -1 to 0.
* identify columns that use 9 as 'unknown' and change them to 0.

I then did some further cleaning by changing the object columns to categories to encode them to numeric values.

**Analysis**

I decided to use PCA and k-means clustering to put my customers into clusters. I chose to use these together as it is believed to improve the clustering results.

First, I need to standardise the data in the data set otherwise, the weights of different values/columns will be treated differently.

I then looked at how many components I should include in my PCA by producing the graph below.

A blue dotted line on a white background

Description automatically generated

Although keeping 80% variance is the rule of thumb, I decided to keep 90%. This meant that I would be keeping 100 components. However, for further investigation, I could try the PCA with 50 components too and see how that affected the results.

A line graph with blue dots

Description automatically generatedI then completed k-means clustering using the results from the PCA. To decide how many clusters I should use, I created an elbow plot, as shown below.

From this plot, I identified that there was a clear ‘elbow’ at 2. Therefore, I decided to use 2 clusters for my k-means clustering.

I repeated the PCA and k-means clustering for the azdias dataset and found that it produced the same results: 90% variance kept by 100 components and 2 clusters in the k-means clustering.

To look at the similarities between the two datasets, I grouped the customers by clusters and see the average values for each variable. I tried to use the compare() function but it did not work so I looked at the output of the groupby function to see how the datasets differed. I also looked at the numbers allocated to each cluster and found them to be very similar in each group. This is expected as I used a sample from the azdias dataset which was the same size as the customer dataset.

**Supervised learning model**

A blue rectangle with white background

Description automatically generatedI brought in the training data and looked at the counts of the response column. I found that there was a vast difference in these counts with the number of responses being extremely low. This will potentially affect the predictions.

Next, I created the supervised learning model. To do this, I needed a classifier as I needed to classify the customers into those who were likely to become customers and those that were not. To decide which was the best classifier model to use, I decided to test out 3 different ones, these were:

1. Random Tree Classifier
2. Random Tree Regressor
3. K-Nearest Neighbors

I decided to use accuracy as the metric to measure the model’s performance as this is an important factor of classification: making accurate predictions. I also looked at the MAE of each model as this would help me to see the average number of errors the model made.

I started by finding the baseline accuracy to see which model/s did better than this. The baseline accuracy is 98.37%. This is very high but considering the number of 0's in the response column, it is not surprising. This implies that I could accurate predict 98.37% of responses by just entering the same number in all. However, I want to see if there is a model that can do better than the baseline.

I found that the results were as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Classifier | MAE | Accuracy on training data | Accuracy on testing data |
| Random Forest Classifier | 0.01 | 99.55% | 98.86% |
| Random Forest Regressor | 0.03 | 75.82% | -5.15% |
| K-Nearest Neighbors | 0.01 | 98.76% | 98.78% |

The Random Forest Classifier was the best model based on accuracy. I found the variables that were the most important in this model. All of them are 0.01 so don’t carry much importance.

Important variables in Random Forest Classifier

Variable: ALTER\_HH Importance: 0.01

Variable: ALTERSKATEGORIE\_FEIN Importance: 0.01

Variable: ARBEIT Importance: 0.01

Variable: CAMEO\_DEUG\_2015 Importance: 0.01

Variable: CAMEO\_INTL\_2015 Importance: 0.01

Variable: CJT\_GESAMTTYP Importance: 0.01

Variable: CJT\_KATALOGNUTZER Importance: 0.01

Variable: CJT\_TYP\_2 Importance: 0.01

Variable: EXTSEL992 Importance: 0.01

Variable: GEBURTSJAHR Importance: 0.01

Variable: GFK\_URLAUBERTYP Importance: 0.01

Variable: INNENSTADT Importance: 0.01

Variable: KBA05\_FRAU Importance: 0.01

Variable: KBA05\_HERST3 Importance: 0.01

Variable: KBA05\_VORB2 Importance: 0.01

Variable: KBA05\_ZUL4 Importance: 0.01

Variable: KBA13\_ANZAHL\_PKW Importance: 0.01

Variable: KBA13\_AUDI Importance: 0.01

Variable: KBA13\_BJ\_2009 Importance: 0.01

Variable: KBA13\_CCM\_1400 Importance: 0.01

Variable: KBA13\_CCM\_3000 Importance: 0.01

Variable: KBA13\_HALTER\_20 Importance: 0.01

Variable: KBA13\_HALTER\_65 Importance: 0.01

Variable: KBA13\_HERST\_AUDI\_VW Importance: 0.01

Variable: KBA13\_KMH\_211 Importance: 0.01

Variable: KBA13\_KRSHERST\_FORD\_OPEL Importance: 0.01

Variable: KBA13\_KW\_40 Importance: 0.01

Variable: KBA13\_NISSAN Importance: 0.01

Variable: KBA13\_SEG\_MINIVANS Importance: 0.01

Variable: KBA13\_SEG\_SPORTWAGEN Importance: 0.01

Variable: KONSUMNAEHE Importance: 0.01

Variable: LP\_LEBENSPHASE\_FEIN Importance: 0.01

Variable: LP\_STATUS\_FEIN Importance: 0.01

Variable: ONLINE\_AFFINITAET Importance: 0.01

Variable: ORTSGR\_KLS9 Importance: 0.01

Variable: PRAEGENDE\_JUGENDJAHRE Importance: 0.01

I decide to perform tuning on this model to get the best model possible. I did this using GridsearchCV and found the best parameters were:

'max\_depth': 2, 'n\_estimators': 5

Finally, I fit the best model to the test dataset and predicted which customers would be likely to become customers. I first cleaned the dataset in the same way I had with the previous sets, then standardised it before fitting it to the best model. I predicted the values and then found which customers would be most likely to become customers. However, it returned 0 customers. As the data in the training set was so imbalanced, it has affected the predictions on the test data set.

**Conclusion**

The problem which needed to be solved is a customer segmentation which classifies customers into groups of those likely to be converted into becoming customers and those who are not likely. This will help the company decide on which customers to target in their marketing campaign.

I solved this problem by first assessing the customer and azdias datasets and identified areas of cleaning. I cleaned all the datasets by changing the object columns that were numbers into floats, dropped the columns that I found were not needed due to providing very little information, removed any duplicate rows that were in the datasets, filled missing values with 0 to represent that they were unknown, and changed all the ‘unknown’ values to 0. I analysed the datasets using PCA, which found that to keep 90% of the variance, I should keep 100 components. Then used this in k-means clustering which I found I should use 2 clusters. Comparing the customers dataset with the azdias showed that there was a similar split in the sets. I then tested out 3 different classifiers and used MAE and accuracy scores to decide the best model, which was Random Forest Classifier. I then tuned this model using GridSearchCV to find the best parameters for the model. I applied the model to the test data and generated predictions. When I tried to find the set of customers who were likely to become customers, the results came back as zero customers, which is not what I would have liked to see.

An area I found difficult in this project was the size of the datasets, mainly the azdias one as it kept making my kernel die when I was trying to clean it. In addition to this, I found that the large size slowed down a lot of the process. I had to restart the kernel many times, reload the workspace, and change the functions I had made. I tried to work on the different sections individually to get around this problem but that was also very time consuming, having to jump about the notebook. After a while, I decided to make the dataset smaller by taking a sample and this solved a lot of issues. Another area I found difficult was having the dataset columns in German. I found that it made it harder to understand what the columns were and I also spent a lot of time searching through the excel file or googling translations to understand the data better.

An area I have enjoyed in this project is of the challenge of dealing with a real-life problem and data set. My confidence in my skills has grown so much doing this project and I feel like I have the skills, ability, and knowledge to deal with data science problems in the real world. It has sparked a passion for data science projects and I will definitely be looking for more on Kaggle after this.

For further improvements, I would look at the different number of components in the PCA, trying only keeping 80% variance and 50 components to see what impact that would have on the clustering of the customers and azdias customers. I would also look at a way I could equalise the class imbalance of the training dataset to see if that improved the predictions. I would expect that having a larger group of customer responses in the training data would create results in the test data. I could also look at other classifiers to see if they would improve the results but as 98.86% accuracy is extremely high, I don’t think changing the classifier would have much impact on the results.

Overall, even though the results did not turn out as I expected and would not be very useful to the company, I have really enjoyed this project and will continue to try new things over the next 2 weeks to see if I can get any improvements.