Udacity+Arvato: Identify Customer Segments

By Javier Cordon

Project Overview

Introduction

The company Arvato Financial Services, a Bertelsmann subsidiary, connected with Udacity to propose a capstone project, as one of the options for the Data Science and Machine Learning Nanodegrees.

They provided different datasets that will be discussed in the following sections, that consist of a mail-order sales company in Gernmany that is interested in identifying segments of the general population to target with their marketing in order to grow. This kind of practices is pursued by different companies in order to help them improve their ROI, by addressing to customers who are likely to buy certain product, churn or default on a loan. (Sebastiaan Höppner, 2017)

For this project I will be addressing the mail-order sales data, among other datasets, in order to apply machine learning techniques to provide an estimator that may infer which individuals are most likely to respond to their marketing campaign and become customers of the mail-order company.

Datasets

There are three datasets to work on for this project, they all have features that describe a group of people, each register/row, represents one person, and these features are the same across the three datasets. The difference across the datasets is that the first file is the biggest one and contain demographics information for the general population, while the second one, contains the prior customers of the mail-order company. In the other hand, we have the third dataset that contains potential customers who were addressed a marketing campaign, and one of the extra fields it has, is the response of each potential customer to the marketing campaign. We may inspect these datasets to find out what these groups have in common to make segmentation of the population to understand who the best potential customers are. And since we also have the response to a marketing campaign, try to predict which of the potential customers may give a positive response.

Acknowledgement

In order to reproduce my solution and results, all development Python code is available in my Github repository. It is important to remark that in addition to Udacity's Terms of Use and other policies, the datasets for this project are governed by terms and conditions of AZ Direct GmbH and is prohibited from publishing or keeping after 2 weeks of downloading it from an official source and accepting their terms and conditions.

Problem Statement

There are two main problems that the Arvato Financial Services wants to address using the datasets provided. First, to analyze attributes of established customers and the general population in order to create customer segments. Second, to use previous analysis to build a model using machine learning techniques in order to infer whether an individual will respond to one of their marketing campaigns, or not.

The problems will be addressed in 3 major steps:

Part 1: Customer Segmentation Report. First, I will be addressing the outliers and missing values. For the outliers I will be implementing a z-score, where a standard deviation of -2.33,+2.33 will be the threshold for outliers. If the same row contains outliers in other features, it will be considered to drop those samples. Features will be selected carefully to consider which have outliers and what criteria to take on those. They will be replaced by the median value of that feature, or the max value admitted by the 2.33 standard deviations, depending of the type of feautre. For missing values, the same principle will be applied. The Mean and Median values registered in the general population, will be kept and be used in the other datasets as well and for reproducibility. Once the data is normalized, I will apply a scaling, so the features are in the same order of magnitude before being ingested by the model. I will use the unsupervised learning techniques of PCA and K-Means to reduce the size of features, in order to cover close to 80% of data variance, and then with K-Means, apply the elbow method in order to find an appropriate number of Groups to cluster the population/customers. Then I will be able to describe parts of the general population that are more likely to be part of the mail-order company's main customer base, and which parts of the general population are less so.

Part 2: Having found the clusters of population that are more likely to be customers of the mail-order company, now I can build an estimator to target those who were assigned a mailout campaign, in order to infer their Response (whether a person became a customer of the company following the campaign, or not). For this I will be using a Supervised Learning Model. I will start as a benchmark with a logistic regression. But I will also compare it to a XGBoost estimator, addressing the unbalanced classes appropriately.

Part 3: Now that I have created the estimator to infer which individuals are most likely to respond to mailout campaign, the estimator will be tested in competition through Kaggle. I will get a score of how well the model performs against unseen data (TEST). AUC will be used to evaluate performance.

Metrics

Analysis and Methodology – Data Exploration, Preprocessing and Visualization

The datasets in detail, as explained in (Arvato, n.d.) are the following:

- 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).

“Each row of the demographics files represents a single person, but also includes information outside of individuals, including information about their household, building, and neighborhood. Use the information from the first two files to figure out how customers ("CUSTOMERS") are similar to or differ from the general population at large ("AZDIAS"), then use your analysis to make predictions on the other two files ("MAILOUT"), predicting which recipients are most likely to become a customer for the mail-order company.

The "CUSTOMERS" file contains three extra columns ('CUSTOMER_GROUP', 'ONLINE_PURCHASE', and 'PRODUCT_GROUP'), which provide broad information about the customers depicted in the file. The original "MAILOUT" file included one additional column, "RESPONSE", which indicated whether or not each recipient became a customer of the company. For the "TRAIN" subset, this column has been retained, but in the "TEST" subset it has been removed; it is against that withheld column that your final predictions will be assessed in the Kaggle competition.” (Arvato, n.d.)

Missing values

I start by analyzing top missing values across the 3 main datasets, General Population from now on “GP”, Customers, and Mailout will be referred as “Training”. By limiting the top 20 features with highest percentage of missing values, we can notice immediately that there are 6 that highlight and are over 30% across the 3 datasets. I will dropped these 6 features since missing values are high across them, but then I notice when analyzing the Customers Groups/Clusters that many of the most important features had a high class value named unknown or similar. Since this doesn’t give enough information for the Segmentation report another approach was required.

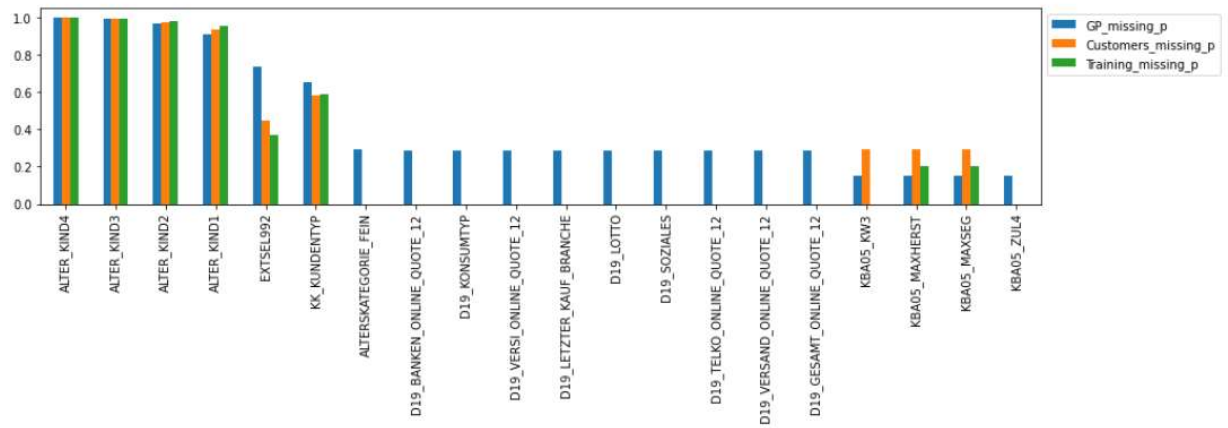


Figure 1: Missing values among GP, Customers and Training Mailout RAW

Therefore, I run a decoding of this values which description was labeled as unknown or similar, and treated them as NaNs, NonAvailable (missing values).

Impressingly the number of features that passed the treshold of 30% was big. From having 6, it was increased nearly 10x, to 65 features containing categories with values that didn't provide enough relevant information. These features/columns were dropped and not considered in the datasets.

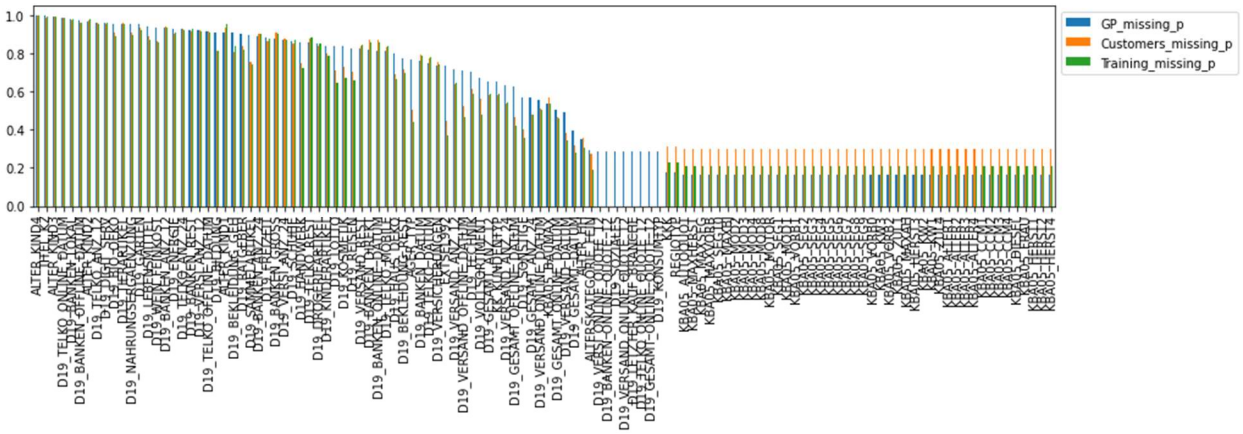


Figure 2: Missing values among GP, Customers and Training Mailout, With Decoding

High Unique values

I will select those features, which have more than 1000 number of unique values per feature, this will help to determine if it is a continuous value, it has many classes, or it is too detailed, that makes it unique per record.

	GP	Customers	Train
LNR	1.000000	1.000000	1.000000
EINGEFUEGT_AM	0.005793	0.015836	0.037242
KBA13_ANZAHL_PKW	0.001416	0.006527	0.028653

Figure 3: High detailed features

- LNR: Is unique per record, and is considered an ID of each one, will not be used for training.
- EINGEFUEGT_AM: Will not consider high number of dates, more than 5k.
- KBA13_ANZAHL_PKW: Related to number of cars in the PLZ8, will not be considered.

Now I will consider as high number of unique values if it has more than 15 unique values per feature, getting the following results:

	unique_values
ANZ_HAUSHALTE_AKTIV	293
ANZ_STATISTISCHE_HAUSHALTE	269
GEBURTSJAHR	117
VERDICHTUNGSRAUM	47
CAMEO_DEU_2015	44
CAMEO_INTL_2015	43
LP_LEBENSPHASE_FEIN	42
EINGEZOGENAM_HH_JAHR	38
D19_LETZTER_KAUF_BRANCHE	35
MIN_GEBAEUDEJAHR	33
ANZ_PERSONEN	31
ALTERSKATEGORIE_FEIN	27
ANZ_HH_TITEL	22
PRAEGENDE_JUGENDJAHRE	16

Figure 4: Unique values on features which threshold is over 15

I will visually inspect their distribution, to find possible outliers and take actions.

Outliers

Now I plot the distribution of features with high number of unique values, in order to detect possible outliers or distributions that may not be helpful for my estimator.

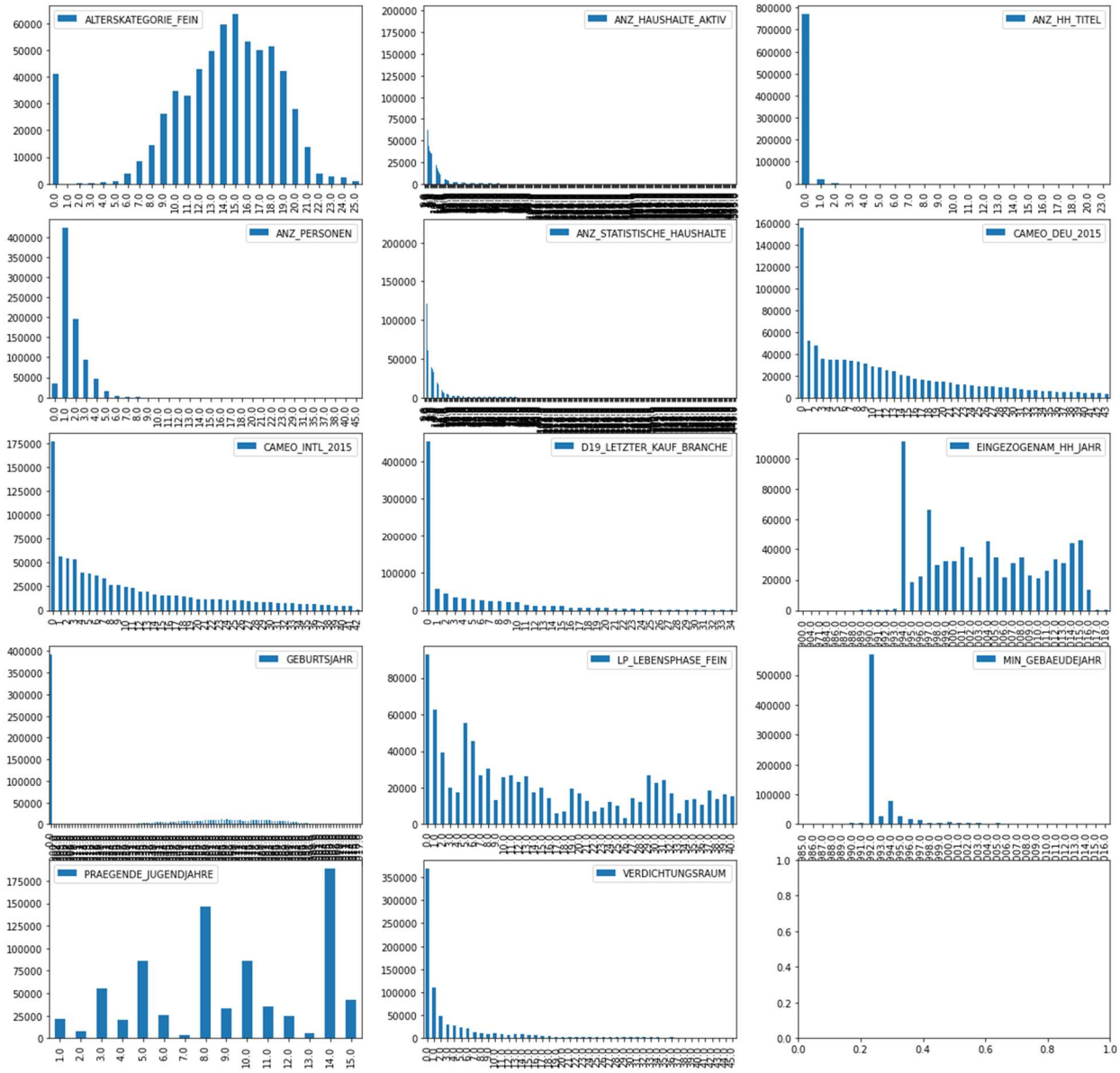


Figure 5: Possible features with outliers, distributions.

After analyzing each of them, I take the following actions:

Apply outlier replacer by max to:

- ANZ_HAUSHALTE_AKTIV: Number of people, if too high, aprox to max allowed in outliers.
- ANZ_PERSONEN: Same criteria being adult persons in the household.
- ANZ_STATISTISCHE_HAUSHALTE: Because of distribution will apply the max rule.

And outlier replacer by median to:

- ALTERSKATEGORIE_FEIN: It is related to age classification, it is a fair number of 0s which may be NaNs, and be considered as outliers.

Will drop:

- ANZ_HH_TITEL. Too unbalanced, attributes say it should be from 1-10
- GEBURTSJAHR: Is a birthday year, and distribution is unbalanced, I will not consider this.

Pass:

- CAMEO_INTL_2015: Not enough information the attribute descriptions.
- CAMEO_DEU_2015: Detailed information about environment
- D19_LETZTER_KAUF_BRANCHE: Purchase Sector related, pass.
- EINGEZOGENAM_HH_JAHR: Has a fair distribution and is related to a year number, will not apply outliers.
- LP_LEBENSPHASE_FEIN: Has a fair distribution, will not apply outliers function.
- MIN_GEBAEUDEJAHR: Will not apply outliers function. Year related.
- PRAEGENDE_JUGENDJAHRE: Will not apply outlier function, has a fair distribution. Dominating movement in the person's youth.
- VERDICHTUNGSRAUM: Not enough information.

Scaling

Before applying a Principal Component Analysis, I will apply a data scaler in order to have them to the same scale. For this matter I will use StandardScaler in order to remove the mean and scale it to unit variance.

After applying the scaler, I get a dataset that looks like the following:

	AKT_DAT_KL	ALTERSKATEGORIE_FEIN	ANZ_HAUSHALTE_AKTIV	ANZ_KINDER	ANZ_PERSONEN	ANZ_STATISTISCHE_HAUSHALTE	ANZ_TITEL
0	-0.371973	0.069501	-0.363725	-0.292527	-0.631373	-0.464667	-0.057885
1	1.338692	2.186757	0.727740	-0.292527	0.372074	1.070323	-0.057885
2	1.338692	0.775253	0.571817	-0.292527	-0.631373	0.217551	-0.057885
3	-0.942195	-0.636251	-0.831495	-0.292527	-1.634819	-0.635222	-0.057885
4	-0.942195	-0.283375	-0.519648	-0.292527	2.378967	-0.464667	-0.057885

Figure 6: Scaled dataset sample.

Principal Component Analysis (PCA)

Principal Component Analysis will be used to reduce the number of features while keeping features that explain most of the data variance.

I will apply the PCA to the N_FEATURES -1, in order to have them all so I can find the explained variance for the top N principal components.

My n_components is 295 after performing all the data cleaning and normalization. I will get the Explained variance from component 1, to 295, being ordered by maximum data variance to minimum. This will let me capture the N components that explain nearly 80% of the data variance.

After applying PCA and plotting the explained variance I get the following plot:

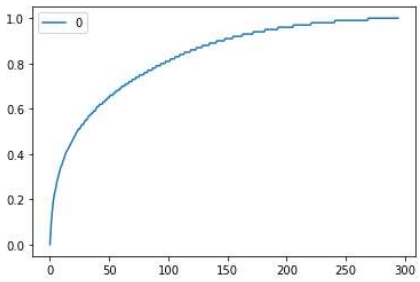


Figure 7: PCA explained variance

And now I find that the first 95 components, explain 80% of the data variance. I will use these components as a baseline, since it does a good job reducing the number of components to 1/3rd and keeps 80% of data variance.

KMeans

KMeans will be used to make clusters of the data, in order to target population segmentation, so I can compare the groups of segmentation between the General Population and the Customers. This will help to identify what are the characteristics of the customers, among the population, that can make a good target for the company's marketing.

In order to choose a proper number of clusters, I calculate several numbers of clusters, ranging from 1 to 35, in order to plot the sum of distances of samples to their closest cluster center. I will apply the Elbow criterion in order to choose the right number of clusters.

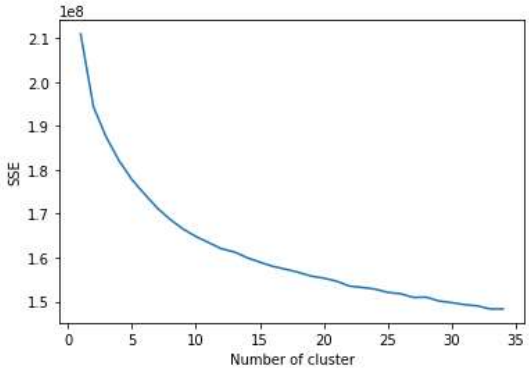
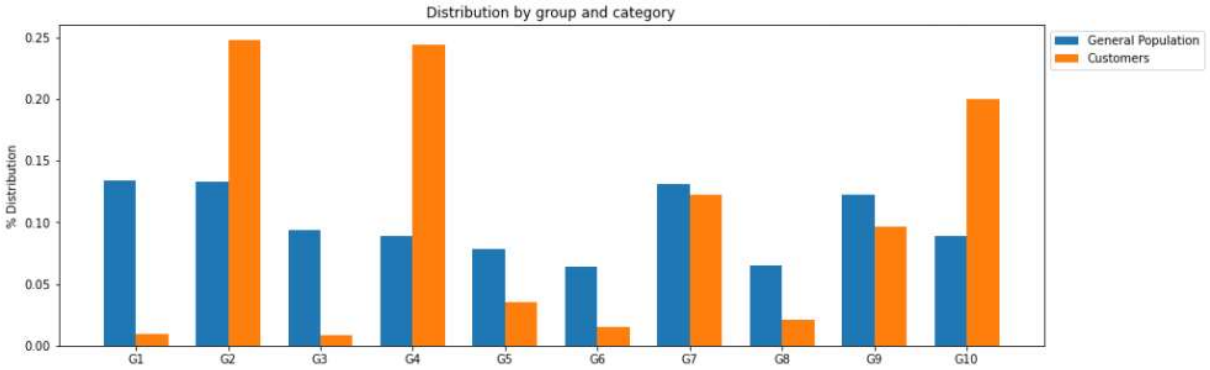


Figure 8: KMeans SSE

From the plot, I find that the first 10 clusters show the biggest steps one from the previous one, before slightly slowing the distance of the samples to their closest clusters centers. I will use as N_CLUSTER the number 10.

After applying and fitting the KMeans estimator to the number of clusters defined to the General Population, and then applying the same estimator to the Customers, I can now plot and compare the percentages of distributions for each of them as follows:

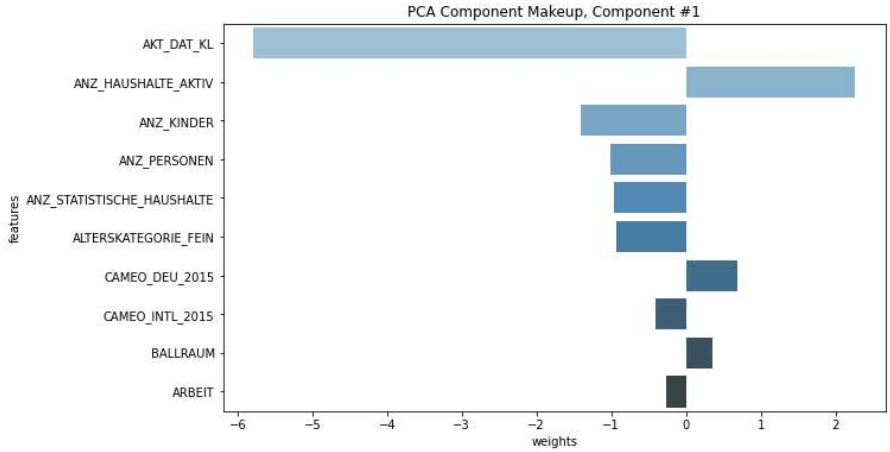


Interestingly, there are 4 groups that describe more than 10% each, of the total of the customers. I will address these groups in order to describe what are their characteristics that better describe them.

Customer Segmentation Report

In order to describe their characteristics, I will use their KMeans centroids, and use their most important principal components, in order to analyze which are the features that describe them.

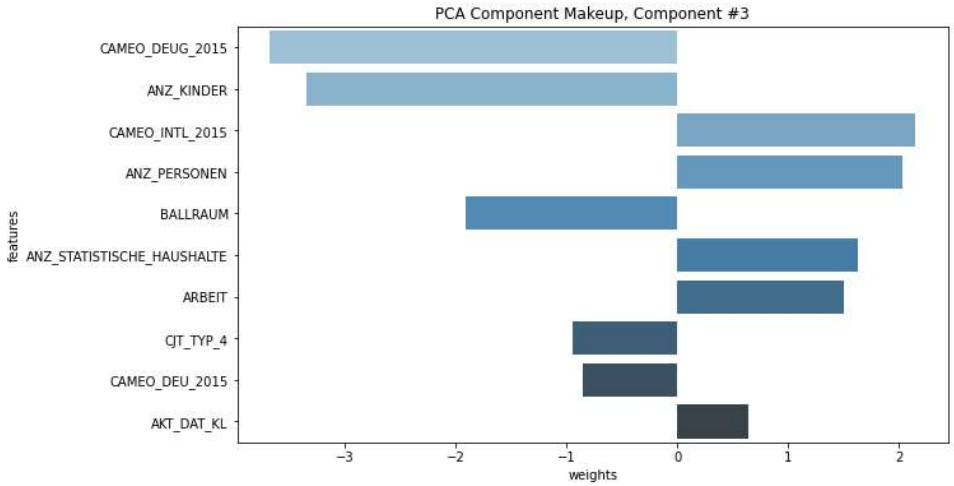
For the group that represents most of the customers, I get the following PCA Component Makeup:




```
AKT_DAT_KL 1 : No Match
ANZ_HAUSHALTE_AKTIV 1 : number of households in the building
ANZ_KINDER 0 : No Match
ANZ_PERSONEN 2 : number of adult persons in the household
ANZ_STATISTISCHE_HAUSHALTE 1 : No Match
ALTERSKATEGORIE_FEIN 10 : No Match
CAMEO_DEU_2015 8 : CAMEO classification 2015 - detailled classification Family Starter
CAMEO_INTL_2015 3 : (each German CAMEO code belongs to one international code) Prosperous Households-Older Families & Mature Co
uples
BALLRAUM 6 : distance to next urban centre  50-100 km
ARBEIT 2 : No Match
KKK 3 : purchasing power average
MOBI_REGIO 5 : moving patterns very low mobility
ANREDE_KZ 1 : gender male
HH_EINKOMMEN_SCORE 4 : estimated household net income  average income
CJT_GESAMTTYP 2 : customer journey typology Advertising- and Consumptiontraditionalist
EWDICHTE 2 : density of inhabitants per square kilometer 34 - 89 HH/km²
HEALTH_TYP 2 : health typology sanitary affine
SHOPPER_TYP 3 : shopping typology demanding shopper
```

Figure 9: PCA Component description 1

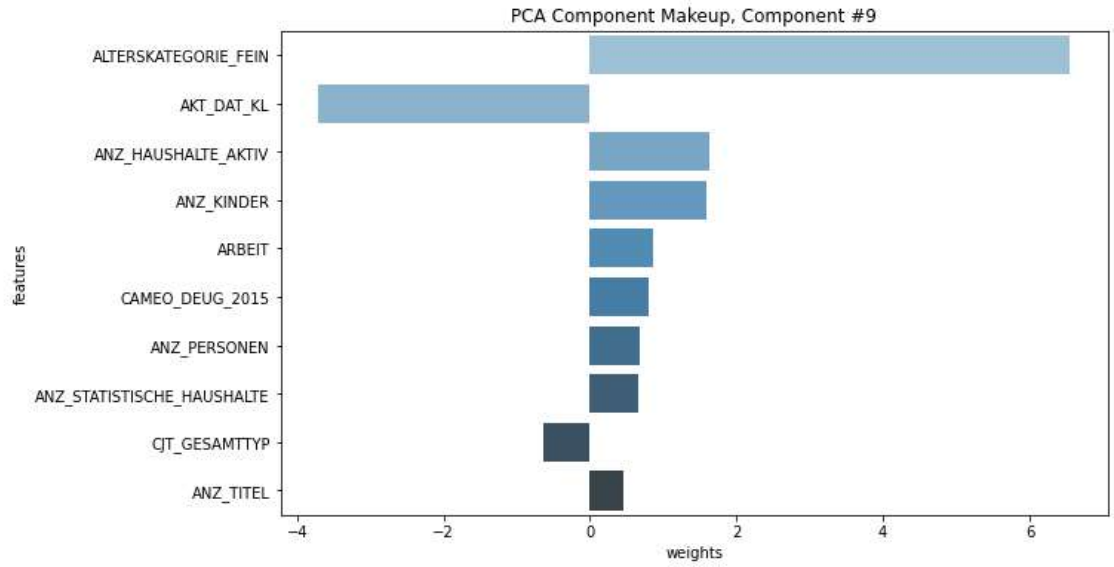
For the Second highest one, the following:




```
CAMEO_DEUG_2015 6 : CAMEO classification 2015 - Uppergroup low-consumption middleclass
ANZ_KINDER 0 : No Match
CAMEO_INTL_2015 0 : (each German CAMEO code belongs to one international code) Poorer Households-Pre-Family Couples & Singles
ANZ_PERSONEN 1 : number of adult persons in the household
BALLRAUM 5 : distance to next urban centre  40 - 50 km
ANZ_STATISTISCHE_HAUSHALTE 3 : No Match
ARBEIT 3 : No Match
CJT_TYP_4 5 : No Match
CAMEO_DEU_2015 0 : CAMEO classification 2015 - detailled classification Petty Bourgeois
AKT_DAT_KL 3 : No Match
KKK 3 : purchasing power average
MOBI_REGIO 3 : moving patterns middle mobility
ANREDE_KZ 2 : gender female
HH_EINKOMMEN_SCORE 2 : estimated household net income  very high income
CJT_GESAMTTYP 6 : customer journey typology Advertising-Enthusiast with restricted Cross-Channel-Behaviour
EWDICHTE 4 : density of inhabitants per square kilometer 150 - 319 HH/km²
HEALTH_TYP 2 : health typology sanitary affine
SHOPPER_TYP 2 : shopping typology family-shopper
```

Figure 10: PCA Component description 2

And for the Third highest one:

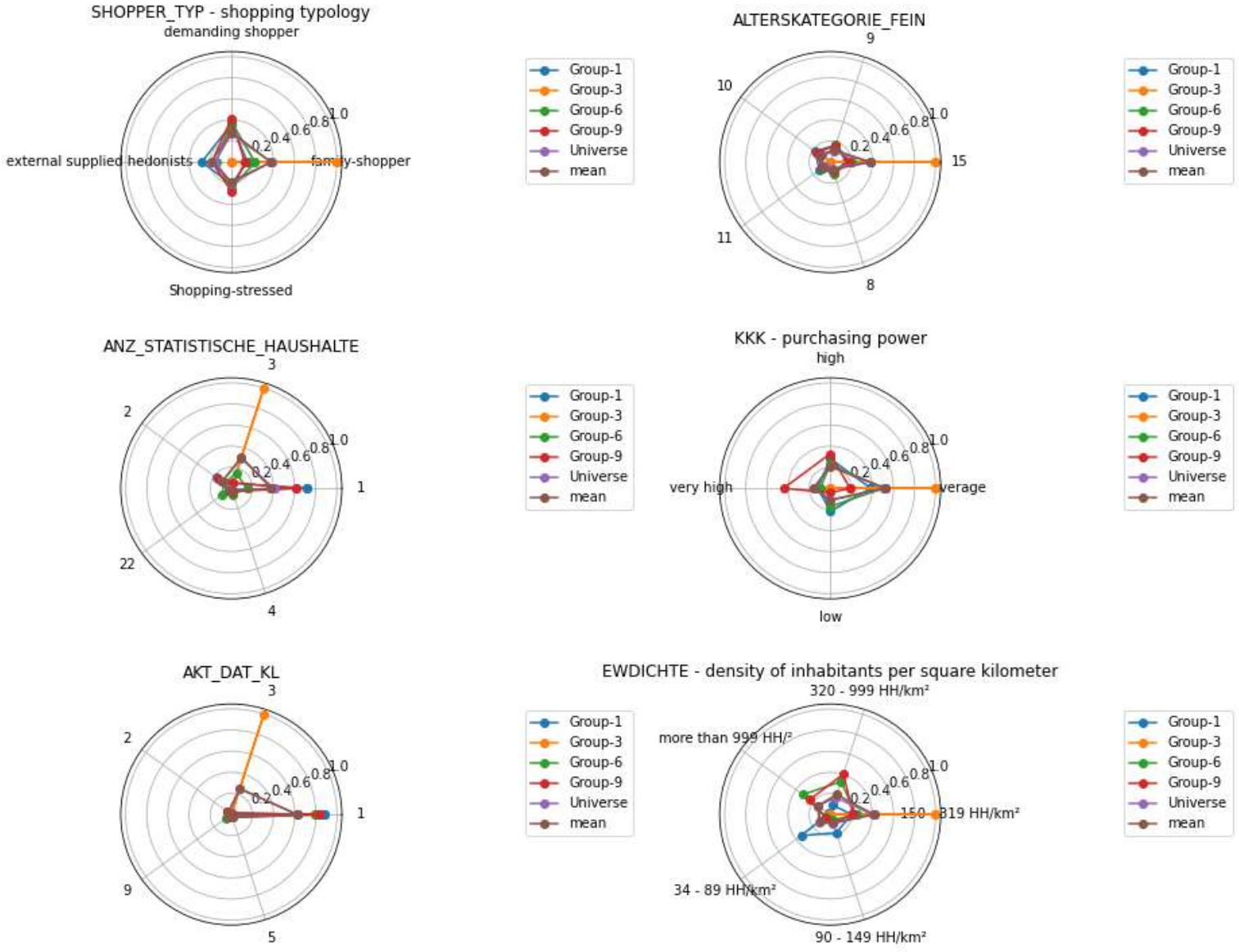


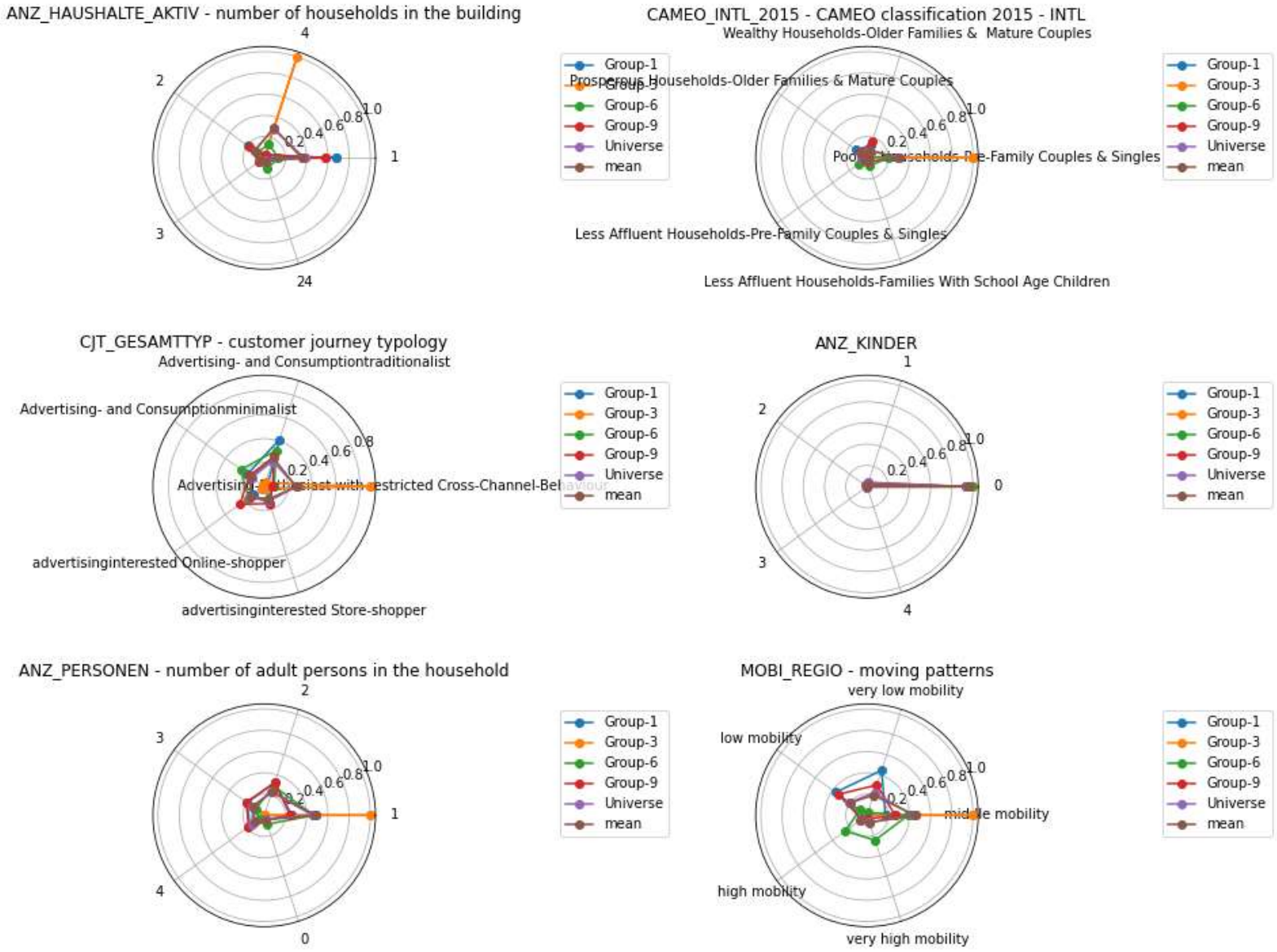

```
ALTERSKATEGORIE_FEIN 15 : No Match
AKT_DAT_KL 1 : No Match
ANZ_HAUSHALTE_AKTIV 1 : number of households in the building
ANZ_KINDER 0 : No Match
ARBEIT 3 : No Match
CAMEO_DEUG_2015 1 : CAMEO classification 2015 - Uppergroup upper class
ANZ_PERSONEN 2 : number of adult persons in the household
ANZ_STATISTISCHE_HAUSHALTE 1 : No Match
CJT_GESAMTTYP 2 : customer journey typology Advertising- and Consumptiontraditionalist
ANZ_TITEL 0 : number of professional title holder in household
KKK 1 : purchasing power very high
MOBI_REGIO 4 : moving patterns low mobility
ANREDE_KZ 1 : gender male
HH_EINKOMMEN_SCORE 1 : estimated household net income  highest income
CJT_GESAMTTYP 2 : customer journey typology Advertising- and Consumptiontraditionalist
EWDICHTE 5 : density of inhabitants per square kilometer 320 - 999 HH/km²
HEALTH_TYP 1 : health typology critical reserved
SHOPPER_TYP 3 : shopping typology demanding shopper
```

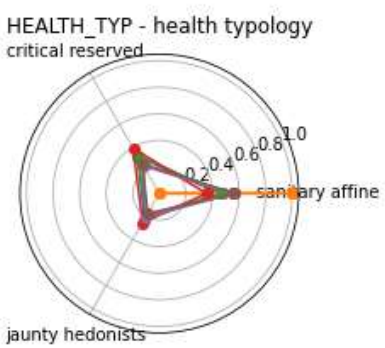
Figure 11: PCA Component description 3

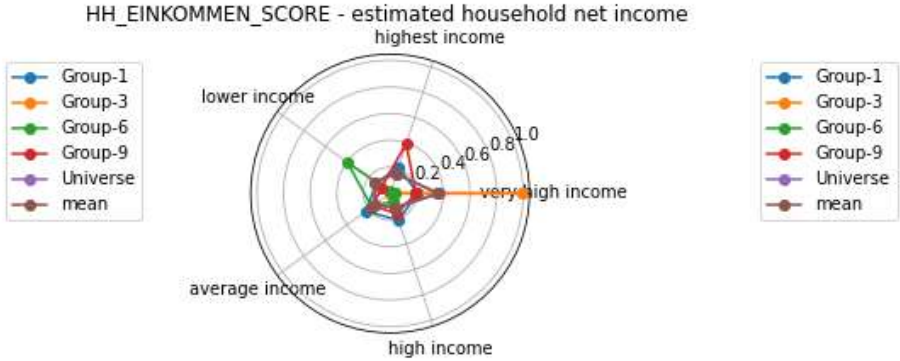
In addition to their highest principal components, I include other features that also describe the people, in order to get a profile or pattern: 'KKK', 'MOBI_REGIO', 'ANREDE_KZ', 'HH_EINKOMMEN_SCORE', 'CJT_GESAMTTYP', 'EWDICHTE', 'HEALTH_TYP', 'SHOPPER_TYP'

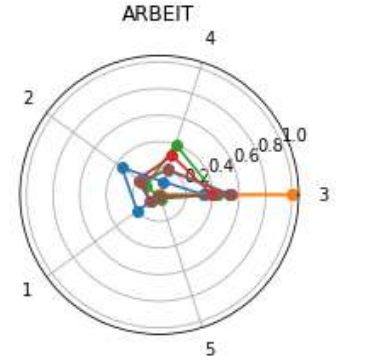
A good way for comparing attributes in marketing studies is through Spider Charts (a.k.a. Radar Charts). It is great for comparing and finding patterns or relationships. I proceed to plot these charts for the groups that represent customers the most, in addition to the mean of the selected groups and the whole customer's (Universe)

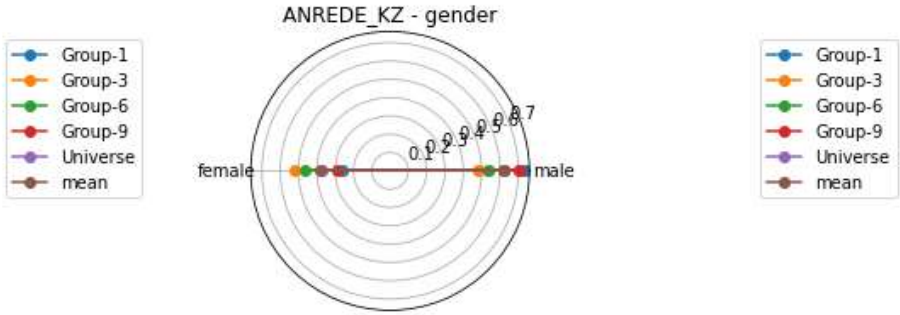

SHOPPER_TYP - shopping typology
demanding shopper
external supplied hedonists
family-shopper
Shopping-stressed
ALTERSKATEGORIE_FEIN
9
10
15
11
8
ANZ_STATISTISCHE_HAUSHALTE
3
2
1
22
4
KKK - purchasing power
high
very high
low
AKT_DAT_KL
3
2
1
9
5
EWDICHTE - density of inhabitants per square kilometer
320 - 999 HH/km²
more than 999 HH/²
150 - 319 HH/km²
34 - 89 HH/km²
90 - 149 HH/km²
Group-1
Group-3
Group-6
Group-9
Universe
mean
0.2
0.4
0.6
0.8
1.0

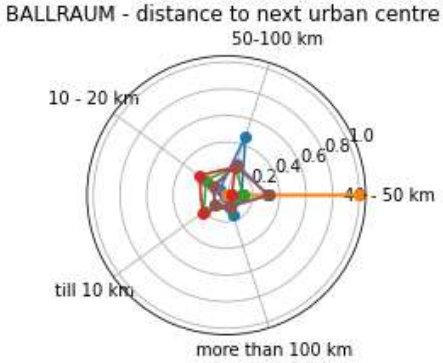

ANZ_HAUSHALTE_AKTIV - number of households in the building
4
2
1
3
24
0.2
0.4
0.6
0.8
1.0
CAMEO_INTL_2015 - CAMEO classification 2015 - INTL
Wealthy Households-Older Families & Mature Couples
Prosperous Households-Older Families & Mature Couples
Less Affluent Households-Pre-Family Couples & Singles
Less Affluent Households-Families With School Age Children
CJT_GESAMTTYP - customer journey typology
Advertising- and Consumptiontraditionalist
Advertising- and Consumptionminimalist
advertisinginterested Online-shopper
advertisinginterested Store-shopper
ANZ_KINDER
1
2
0
3
4
ANZ_PERSONEN - number of adult persons in the household
2
3
1
4
0
MOBI_REGIO - moving patterns
very low mobility
low mobility
middle mobility
high mobility
very high mobility
Group-1
Group-3
Group-6
Group-9
Universe
mean

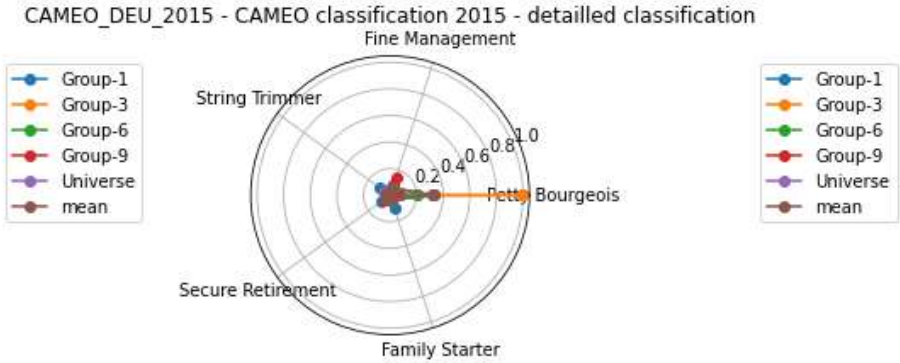

HEALTH_TYP - health typology
critical reserved
sanitary affine
jaunty hedonists
0.2
0.4
0.6
0.8
1.0


HH_EINKOMMEN_SCORE - estimated household net income
highest income
very high income
high income
average income
lower income
Group-1
Group-3
Group-6
Group-9
Universe
mean


ARBEIT
1
2
3
4
5
0.2
0.4
0.6
0.8
1.0


ANREDE_KZ - gender
female
male
Group-1
Group-3
Group-6
Group-9
Universe
mean


BALLRAUM - distance to next urban centre
50-100 km
10 - 20 km
40 - 50 km
till 10 km
more than 100 km
0.2
0.4
0.6
0.8
1.0


CAMEO_DEU_2015 - CAMEO classification 2015 - detailled classification
Fine Management
String Trimmer
Petty Bourgeois
Secure Retirement
Family Starter
Group-1
Group-3
Group-6
Group-9
Universe
mean

The Group 4 outstands from the others, it seems to be more narrowed through the different categories values and it will be described separately.

Group 4: This is the second largest group that corresponds to almost 25% of the customers. Given the outstanding category values, narrowed to one in most of the categories, I made an in detailed analysis of this group, and suspected that it could be related to the normalization and missing values management.

I get the following results:

```
HEALTH_TYP
           Customers Total NaN: 0.0
           Group 3 NaNs from Total: 0.0
           Customers over NaNs: 0.0
ALTERSKATEGORIE_FEIN
           Customers Total NaN: 0.37
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.9
KKK
           Customers Total NaN: 0.39
           Group 3 NaNs from Total: 0.34
           Customers over NaNs: 0.86
MOBI_REGIO
           Customers Total NaN: 0.41
           Group 3 NaNs from Total: 0.34
           Customers over NaNs: 0.83
CAMEO_INTL_2015
           Customers Total NaN: 0.36
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
AKT_DAT_KL
           Customers Total NaN: 0.32
           Group 3 NaNs from Total: 0.32
           Customers over NaNs: 1.0
ANZ_STATISTISCHE_HAUSHALTE
           Customers Total NaN: 0.35
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
EWDICHTE
           Customers Total NaN: 0.35
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
ANZ_HAUSHALTE_AKTIV
           Customers Total NaN: 0.35
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
```

```
ANZ_KINDER
           Customers Total NaN: 0.32
           Group 3 NaNs from Total: 0.32
           Customers over NaNs: 1.0
ARBEIT
           Customers Total NaN: 0.36
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.92
ANREDE_KZ
           Customers Total NaN: 0.0
           Group 3 NaNs from Total: 0.0
           Customers over NaNs: 0.0
ANZ_PERSONEN
           Customers Total NaN: 0.32
           Group 3 NaNs from Total: 0.32
           Customers over NaNs: 1.0
CJT_GESAMTTYP
           Customers Total NaN: 0.02
           Group 3 NaNs from Total: 0.0
           Customers over NaNs: 0.0
SHOPPER_TYP
           Customers Total NaN: 0.0
           Group 3 NaNs from Total: 0.0
           Customers over NaNs: 0.0
CAMEO_DEU_2015
           Customers Total NaN: 0.36
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
BALLRAUM
           Customers Total NaN: 0.35
           Group 3 NaNs from Total: 0.33
           Customers over NaNs: 0.93
HH_EINKOMMEN_SCORE
           Customers Total NaN: 0.02
           Group 3 NaNs from Total: 0.0
           Customers over NaNs: 0.02
```

Most of the features with Missing Values (NaNs) were replaced by the mean or median, and we can see how most of them (72%), were originally NaNs, and represents more than 90% of the total among the NaNs per each of these features. Therefore I can't use their description to give conclusions of this group, and will only consider those which have low number of NaNs in the original dataset:

- HEALTH_TYP
- ANREDE_KZ
- CJT_GESAMTTYP
- SHOPPER_TYP
- HH_EINKOMMEN_SCORE

They are highlighted by being sanitary affine, gender is equally distributed, and from their customer journey typology, they are strongly considered in advertising as Enthusiast with restricted Cross-Channel-Behavior. Also, they are all considered family shoppers with very high income, I will refer to them as **Family Shoppers traditionalist**.

Describing now the groups 2, 7 and 10. Anticipating their strongest quality I will refer to them as:

2 – Traditionalist: Middle Class

7 – Mobile Urbans: Low budget

10 – Online Shoppers: High Income

They share similar number of households in the building, 1 or 2 with exception of the Mobile Urbans which have a higher number; their healthy typology is very similar as well, with a small highlight on Online Shoppers being critical reserved.

Traditionalist: They belong to Middle class, having from average to high income, they have from low to very low moving patterns formed by prosperous households and mature couples. Their environment is not densely populated, they tend to live in more open areas and this is why they probably have lower mobility. Their shopping typology is highlighted by demanding shoppers and external supplied hedonists.

Mobile Urbans: They have high moving patterns, and live in very dense areas, given these high dense areas they live in big, crowded cities. They conform part of poor families or people with the lower income, and this drives them to be advertising and consumption minimalist. Their shopping typology is highlighted by demanding shoppers and external supplied hedonists.

Online Shoppers: This group has a very high purchasing power with the highest income, most of them tend to be fine in management. They are highly motivated by Online Shopping and are also considered Consumption Traditionalist. Their moving patterns are not very high. Their shopping typology is highlighted by demanding shoppers and stressed.

Family Shoppers Traditionalist and **Traditionalist** represent nearly 50% of the Customers, followed by nearly 20% with **Online Shoppers** and lastly nearly 12% of **Mobile Urbans**.

Implementation

Mailout Training data highlights

Before proceeding to the model construction, I compare the Mailout Training to the Customer's dataset, and there are slight differences in the segmentation groups, which I describe as follows.

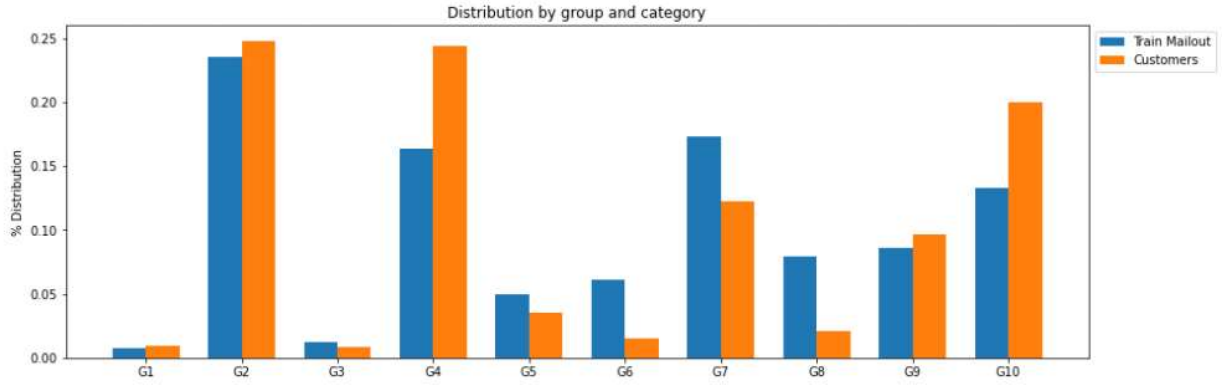


Figure 12: Clusters distribution comparison (Customers Vs Train Mailout)

There is less presence of Family Shoppers Traditionalists on the training mailout dataset, and slightly more in Mobile Urbans. Also, there is slightly less presence on Online Shoppers.

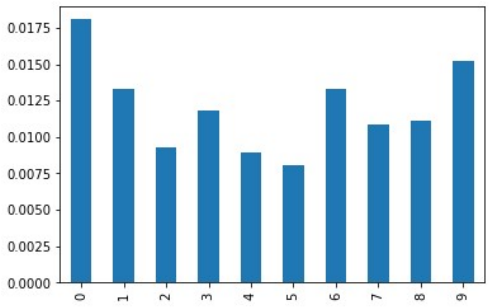


Figure 13:Positive Respondants distribution on the Training Mailout Dataset, among Clusters.

The ratio of the positive respondents to the mailout campaign is well distributed among each of their clusters, there is no highlight in a particular Group, where there could be a noticeable higher proportion of respondents, among that group segment. They range from 0.75% to 1.75% per cluster.

Using the K-Means for clustering as my data input for the model prediction, might not be as useful as I would have desired, I would only have 10 features and as demonstrated, there is no strong relationship between a cluster and respondents' ratio. If I had seen that one group had a striking ratio over the other groups, it would have been a good indication of a relationship, and it could have been used for predicting.

If I get enough time, using K-Means output as input for a model to predict positive responses, I will consider it for stacking with the other estimators.

Feature Selection

Before proceeding with modeling, I will explore if any feature is strongly related to the target variable, and select those that are more likely to be related in order to reduce the amount of features and only select those which may lead to better inference.

First I normalize the training Mailout dataset with previous pre processors, and filter those records that only have a positive response (1). Then I select each feature, one at the time, and group them into their categories. This helps me to visualize if one of their categories has a stronger ratio of more positive response than its other classes, I call this feat_importnace. Once I have this distribution among a feature's classes, I calculate different ratios.

- Ratios_per_class = Feat_importance/total_positive
- Overall Total = feat_importance/feat_distribution, where the latter is the distribution without only filtering positive responses
- Feat Dist = feat_importance/feat_distribution_total, where the latter is the sum of positive records
- Feat Dist Inv = 1 – Feat Dist
- Feature response Weight = ratios_per_class * Feat Dist Inv

D19_SOZIALES	RESPONSE	Over Total	Feat Dist	Feat Dist Inv	Feat - Response Weight
0	0.2124	0.0066	0.4003	0.5997	0.1274
1	0.703	0.034	0.2558	0.7442	0.5232
2	0.015	0.0054	0.0347	0.9653	0.0145
3	0.0432	0.0027	0.2007	0.7993	0.0346
4	0.0226	0.0038	0.0744	0.9256	0.0209
5	0.0038	0.0014	0.034	0.966	0.0036

Figure 14: Example of feature processed for finding important features

I found that Feature response Weight gives a really good sense of how good it relates to the target feature, and when setting a threshold higher than 0.25 to 0.30, it manages well the unbalanced data. From the result of normalized features equal to 297, and choosing the threshold to 0.25, I manage to keep 105 important features. I will perform a small A/B Testing for selecting either all normalized or the important features under y criteria.

For a first benchmark I will address the following estimators:

- Logistic Regression
- Gradient Boosting Classifier
- Random Forest Classifier
- Extra Trees Classifier
- LGBM Classifier
- XGB Classifier
- Ada Boost Classifier
- SVC
- MLP Classifier

Each of them with four flavors of inputs:

- All features normalized with MinMax Scaler
- All features normalized with Standard Scaler
- Important features normalized with Min Max Scaler
- Important feature normalized with Standard Scaler

The results are the following:

	Model	Score	Model	Score
LogisticR	MinMaxFullFeat	0.6636	MinMaxImportantFeat	0.6802
GradientB	MinMaxFullFeat	0.7619	MinMaxImportantFeat	0.7661
RandomF	MinMaxFullFeat	0.6105	MinMaxImportantFeat	0.6254
ExtraTrees	MinMaxFullFeat	0.5886	MinMaxImportantFeat	0.6148
LGBM	MinMaxFullFeat	0.7208	MinMaxImportantFeat	0.7243
XGB	MinMaxFullFeat	0.6772	MinMaxImportantFeat	0.6764
AdaB	MinMaxFullFeat	0.7392	MinMaxImportantFeat	0.754
SVC	MinMaxFullFeat	0.6174	MinMaxImportantFeat	0.5955
MLP	MinMaxFullFeat	0.5942	MinMaxImportantFeat	0.6053
LogisticR	StandardFullFeat	0.6537	StandardImportantFeat	0.6753
GradientB	StandardFullFeat	0.762	StandardImportantFeat	0.7661
RandomF	StandardFullFeat	0.6117	StandardImportantFeat	0.6263
ExtraTrees	StandardFullFeat	0.5886	StandardImportantFeat	0.6148
LGBM	StandardFullFeat	0.714	StandardImportantFeat	0.7249
XGB	StandardFullFeat	0.6772	StandardImportantFeat	0.6764
AdaB	StandardFullFeat	0.7392	StandardImportantFeat	0.754
SVC	StandardFullFeat	0.6044	StandardImportantFeat	0.5853
MLP	StandardFullFeat	0.5999	StandardImportantFeat	0.5859

Top estimators are **GradientBoost**, **AdaBoost** and **LGB** for Standard Scaler and using only filtered **Important Features**. I know from experience **XGB** can perform better, so I will give it a try for a GridSearch as well.

Estimators hyper-parameters

I ran different grid search to find best hyper-parameters for the estimators in contest. Thanks to the AWS EC2 Instances I was able to get faster results, be aware that it comes with its prices, you only pay for what you use but some of these grid search lasted hours.

Estimator	Candidates	Fits	Time	ROC AUC Score (CV)	EC2	vCPU
XGBoost	160	800	10h 49m	0.7833	c5.24xlarge	96
XGBoost	120	600	7h 48m	0.7829	c5.4xlarge	16
LGB	1500	7500	18min	0.729	c5.4xlarge	16
LGB	240	1200	4min	0.7819	c5.4xlarge	16
AdaBoost	351	1755	18min	0.787	c5.4xlarge	16
AdaBoost	330	1650	25min	0.7888	c5.4xlarge	16
GradientBoost	900	4500		Stopped it.	c5.4xlarge	16
GradientBoost	1152	5760		Stopped it.	c5.24xlarge	96

Figure 15: Estimators GridSearchCV Best Hyper-Parameters

I had to stop the GridSearchCV for GradientBoost since it was taking a long time to cover the candidates. And it was not under a free aws account tier.

After keeping best hyper-parameters, I trained the best estimator with all the training set, since Cross Validation showed a low variance, hance, not apparent over fitting.

If Test dataset was randomly selected from the same universe as training, I should be getting similar score when submitting, otherwise I either overfit, or there is not enough information about test set to get conclusions, since this could be a different distribution, or selected from a different universe than the training.

Experimental Stacking

I ventured to experiment with estimators stacking. I stacked the output of the four estimators XGBoost, LGB, AdaBoost and GradientBoost, and tried applying the same types of estimators for the second layer. I noticed overfitting and would have need more time and advise on how to properly address this in estimators stacking.

Results of this experiment were not as bad, but I believe they could be improved, the best AUC score I have got was 0.71

Results

As previously shown in Estimators hyper-parameters section, after finding the best, I trained again the estimators using 80% of the Training Mailout dataset, and obtained the following AUC.

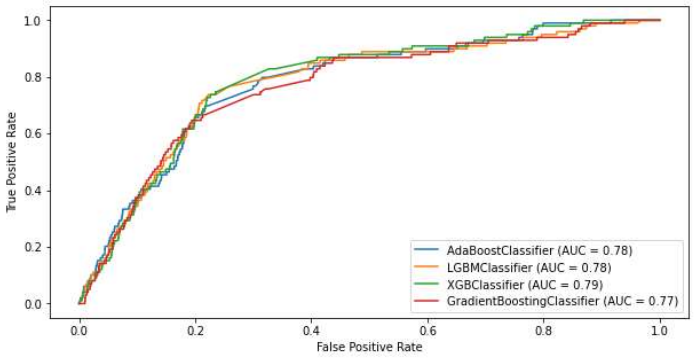
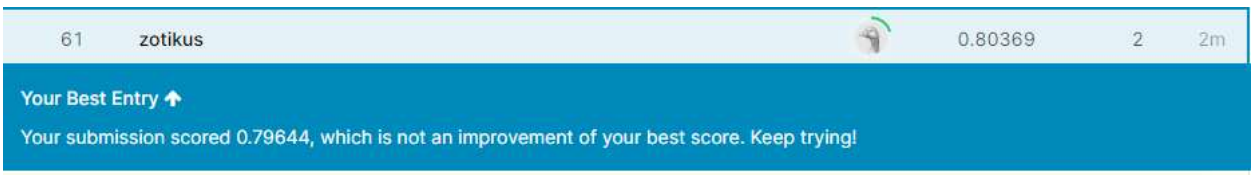


Figure 16: ROC-AUC Results for best estimators.

Where best estimator was XGBClassifier with AUC =0.79

With this estimator I managed to get a AUC score of 0.80369 with the Test MAILOUT dataset.



I then tried training the whole Training Dataset, but the score was a little lower, getting 0.79644.

Conclusion

Real world ML projects can be very tricky when handling data normalization and cleaning. There are many methods and criteria to perform these tasks, and it will impact on the model outcome, thus, it is important to take a good time for working on this stage of the project.

Finding the Customer's Segmentation required a good effort for cleaning data. It was required to remove features that had a high number of missing values, and it was only through training the unsupervised model that I could find new features that were not managed properly. It was a must to perform the feature decoding, where there were categories that could be interpreted as missing, and the number of features with high number of missing increased dramatically. This helped in cleaning the data, but it was especially important to normalize it, analyzing each feature to take the proper action. I found that using quantiles for replacing outliers or missing values performed poorly, since it was considering some classes as outliers when they were not. I rather chose to use Z-Score and it performed better. Finally, I had to work on a CSV table that could help me retrieve the features description, for plotting the radar (spider) plots to compare each of the segments and make it visually easier to understand and interpret.

Having done a good preprocessor with classes and methods easy to reproduce data cleaning and normalization was a key to work on the predictions. Feeling comfortable enough with the data in order to focus on the modeling is important so we do not carry-on possible uncertainties that lead to estimators not being able to improve. Using AWS EC2 instances for training was useful for reducing timing compared to my local host, it is important to keep track of time using the instances and the type being used in order not to incur on undesired expenses. Estimator's benchmarking was important to select the models to work on for performing a grid search in order to find the best hyper-parameters to get the best estimators.

It was satisfying experimenting with model stacking, but future work and research is required to take better advantage of this strategy.

By Javier Cordon

Bibliography

Arvato. (n.d.). *Udacity+Arvato: Identify Customer Segments*. Retrieved from Kaggle: https://www.kaggle.com/c/udacity-arvato-identify-customers/overview/description

Sebastiaan Höppner, E. S. (2017). *Profit Driven Decision Trees for Churn Prediction*. Retrieved from arxiv.org: https://arxiv.org/abs/1712.08101