Arvato Capstone Project

Project Overview

The Arvato Project lies in the domain of Customer Segmentation and Targeting for Mail Order Campaign.
Customer Segmentation, that is finding groups of customers based on their common characteristics is a typical problem in marketing. It allows businesses to

- adjust products for specific groups
- reach each group more effectively by appropriate channels
- identify cross- and up-selling opportunities specific for each group

This improves customer service and leads to better customer relationships. The business can concentrate on the most profitable customers and avoid losses due to scattering.
Segmentaton can be based on

- geographical data
- demographical data
- behavioral data
- known customer lifecycles

Targeting for mail order campaign allow businesses to select specific mail recipients who are likely to become customers. This optimizes the response rate on advertisements and avoids spreading lost. The success of the campaign is typically measured by the uplift in response rate relative to random selection of mail recipients.

Arvato provides 4 datasets with the same 366 demographic (or mostly demographic) features:
1) 891221 datapoints (persons) for general population (azdias)
2) 191652 datapoints for customers of the company; it has 3 extra features with broad information about the customers
3) 42982 datapoints for historical mail order campaign with ‘yes/no’ response variable; this dataset is imbalanced with ca. 1.25% positives
4) 42833 test datapoints for predicting response

Problem statement

In Arvato Project the demographical data of population are used as common characteristics. With this data, the customer segmentation problem can be formulated as clustering (unsupervised learning). The challenge lies in the fact, that there are over 360 characteristics all of which are categorical or ordinal. The same characteristics are know for existing customers. This allows choosing characteristics which are important for distinguishing the customers from general population. The problem can be formulated as binary classification (supervised learning).
Targeting for mail order part is based on the same common characteristics. With the know information of who responded to the mail campaign, the problem can be formulated as binary classification (supervised learning). The result of the model will be the probability for a mail

recipient to become a customer. Thus the campain can choose to address top-N recipients (with the highest probability of becoming a customer).

Metrics

The best evaluation metrics for this case is ROC AUC – area under the receiver operating characteristics curve. It uses the predicted probabilities (actually, only their order is important) and allows easy interpretation and explanation for marketing specialists.
When making a decision about sending a mail proposal, the business should take top-N people with the highest predicted probability of accepting the order. The business value of the Machine Learning Algorithm then can be estimated as an uplift of responses received in comparison with the hypothetical case of sending mail randomly.

Data Exploration

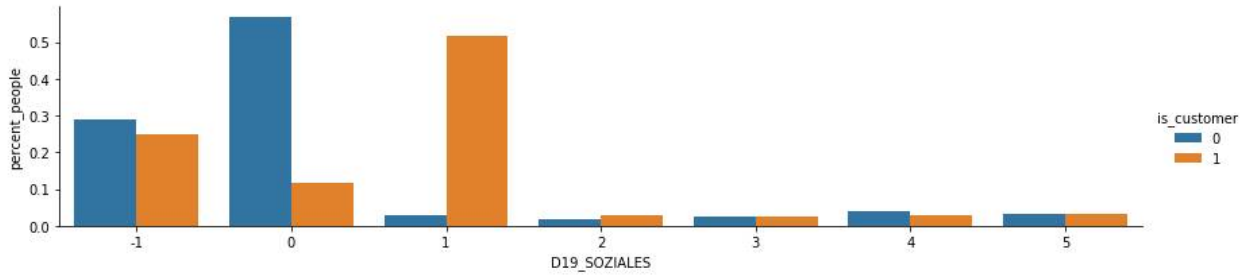
All (or nearly all) 366 features are categorical or ordinal. Nearly all of them have missing values, which in turn can be differently encoded (for example, as -1 or 0 or sometimes 9). Most of features have 5-30 categories. 'LRN' feature serves as index. Some of features, as judged by our human understanding, can be considered as useless for Machine Learning, for example 'EINGEFÜGT AM'. Here is an example of some features

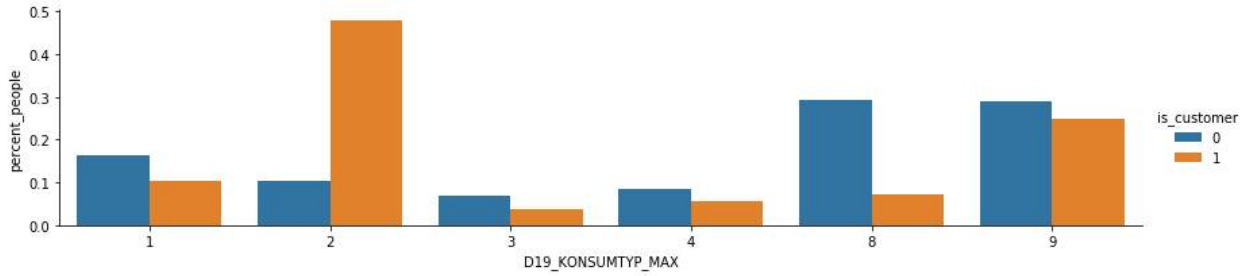
LNR	AGER_TYP	AKT_DAT_KL	ALTER_HH	ALTER_KIND1	ALTER_KIND2	ALTER_KIND3	ALTER_KIND4	ALTERSKATEGORIE_FEIN
9626	2	1.0	10.0	NaN	NaN	NaN	NaN	10.0
9628	-1	9.0	11.0	NaN	NaN	NaN	NaN	NaN
143872	-1	1.0	6.0	NaN	NaN	NaN	NaN	0.0
143873	1	1.0	8.0	NaN	NaN	NaN	NaN	8.0
143874	-1	1.0	20.0	NaN	NaN	NaN	NaN	14.0

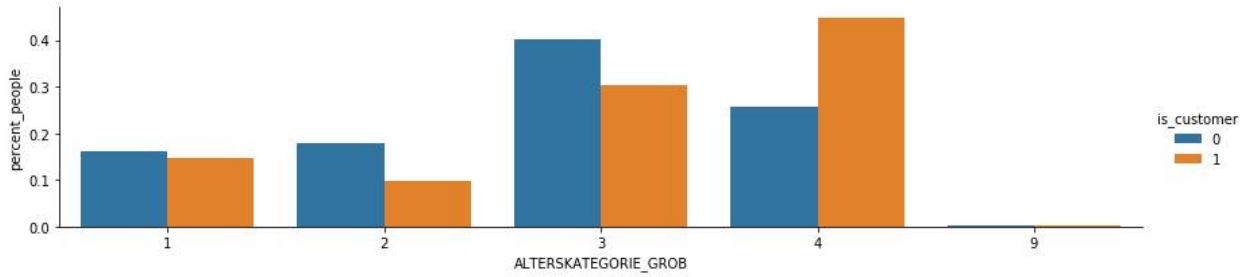
All of features are supposed to be documented in the Excel files 'DIAS Attributes - Values 2017.xlsx' and 'DIAS Information Levels - Attributes 2017.xlsx'. Unfortunately, this is not the case. Some of important features are not documented, for example: D19_SOZIALES, VK_ZG11, VK_DISTANZ, CJT_TYP_6, CJT_TYP_5, CJT_TYP_3, EXTSEL992, AKT_DAT_K, VK_DHT4A, EINGEFÜGT AM.

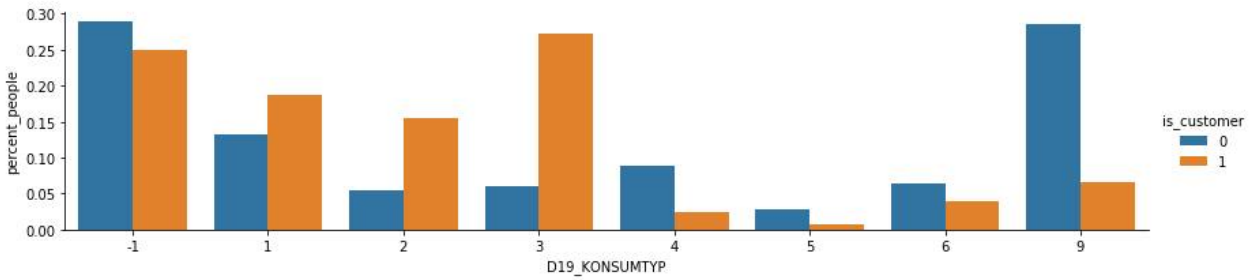
Exploratory Visualization

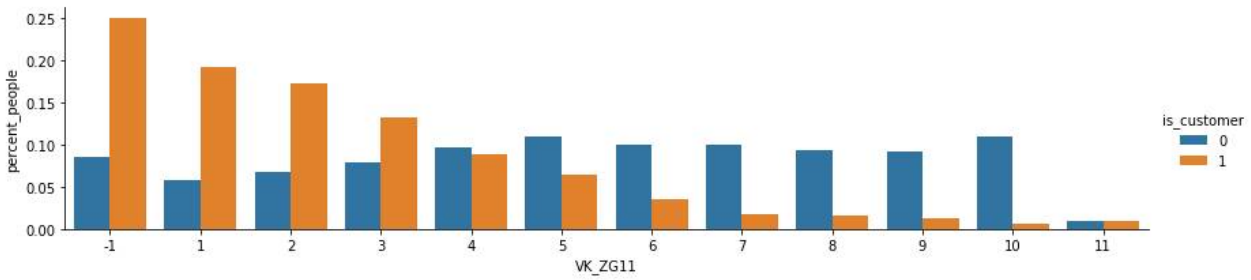
1) customers significantly overrepresented in the group '1' and underrepresented in group '0' of 'D19_SOZIALES'. Unfortunately, 'D19_SOZIALES' is not documented in Excel files.
2) customers significantly overrepresented in the group '2' and underrepresented in group '8' of 'D19_KONSUMTYP_MAX'. Unfortunately, 'D19_KONSUMTYP_MAX' is not documented in Excel files.
3) customers overrepresented in the group '4'='>60 year' and underrepresented in groups '2'='30-45 years' and '3'=46-60 years' of 'ALTERKATETGORIE_GROB'.

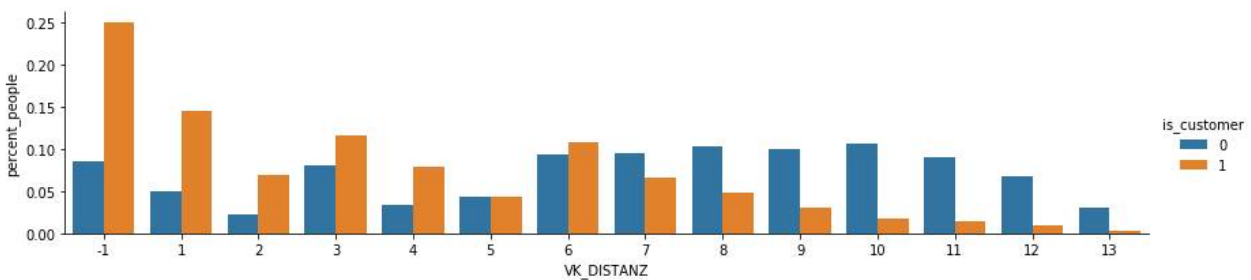

percent_people
0.0
0.1
0.2
0.3
0.4
0.5
-1
0
1
2
3
4
5
D19_SOZIALES
is_customer
0
1


percent_people
0.0
0.1
0.2
0.3
0.4
0.5
1
2
3
4
8
9
D19_KONSUMTYP_MAX
is_customer
0
1

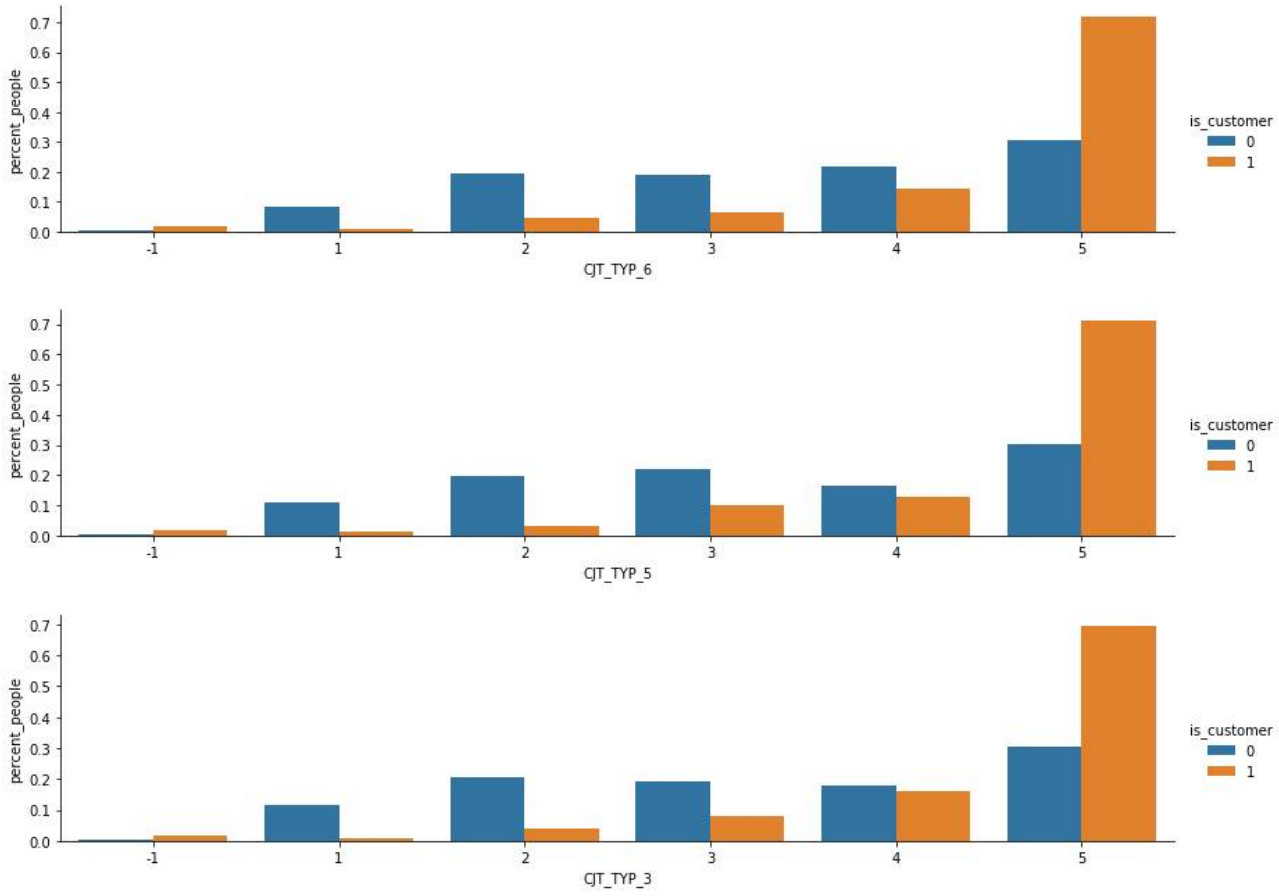

percent_people
0.0
0.1
0.2
0.3
0.4
1
2
3
4
9
ALTERSKATEGORIE_GROB
is_customer
0
1


percent_people
0.00
0.05
0.10
0.15
0.20
0.25
0.30
-1
1
2
3
4
5
6
9
D19_KONSUMTYP
is_customer
0
1


percent_people
0.00
0.05
0.10
0.15
0.20
0.25
-1
1
2
3
4
5
6
7
8
9
10
11
VK_ZG11
is_customer
0
1


percent_people
0.00
0.05
0.10
0.15
0.20
0.25
-1
1
2
3
4
5
6
7
8
9
10
11
12
13
VK_DISTANZ
is_customer
0
1

4) customers overrepresented in the groups '3'='Gourmet' and '2'='Versatile' and significantly underrepresented in groups '9'='Inactive' and '4'=Family' of 'D19_Konsumtyp'='consumption type'.
5) customers significantly overrepresented in the groups with low values and underrepresented in groups with high values of 'VK_ZG11'. Unfortunately, 'VK_ZG11' is not documented in Excel files.
6) customers significantly overrepresented in the group '5' and underrepresented in groups '1', '2' and '3' of 'CJT_TYP_6'. Unfortunately, 'CJT_TYP_6' is not documented in Excel files.
7) customers significantly overrepresented in the group '5' and underrepresented in groups '1', '2' and '3' of 'CJT_TYP_5'. Unfortunately, 'CJT_TYP_5' is not documented in Excel files.
8) customers significantly overrepresented in the group '5' and underrepresented in groups '1', '2' and '3' of 'CJT_TYP_3'. Unfortunately, 'CJT_TYP_3' is not documented in Excel files.

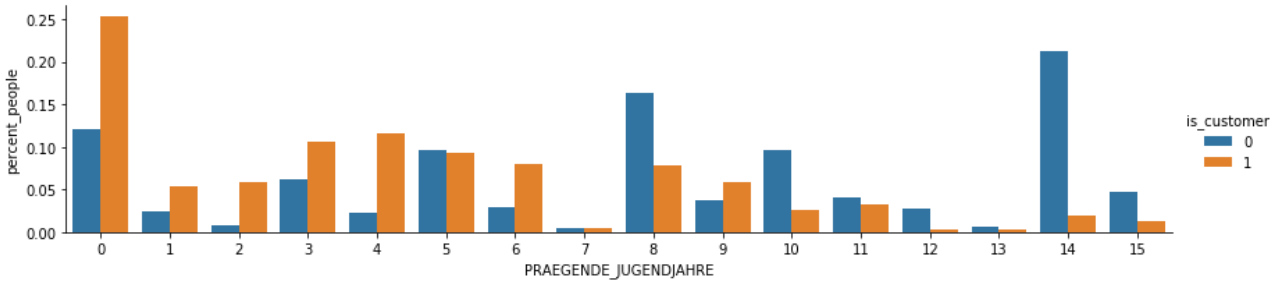


9) customers are overrepresented in the following groups of 'PRAEGENDE_JUGENDJAHRE' = dominating movement in the person's youth (avantgarde or mainstream). This agrees with the previous finding abour the age of the customer.

```
'0'='unknown',
'2'='40ies-reconstruction years (Avantgarde, O+W)',
'4'='50ies - milk bar / Individualisation (Avantgarde, O+W)',
'6'='60ies - generation 68 / student protestors (Avantgarde, W)'
```

10) customers are underrepresented in the following groups of 'PRAEGENDE_JUGENDJAHRE' = dominating movement in the person's youth (avantgarde or mainstream). This agrees with the previous finding abour the age of the customer.

```
'10'='80ies - Generation Golf (Mainstream, W)'
'12'='80ies - FDJ / communist party youth organisation (Mainstream, O)'
'14'='90ies - digital media kids (Mainstream, O+W)'
'15'='90ies - ecological awareness (Avantgarde, O+W)'
```



Algorithm and Techniques

I start by analyzing the data and treating the missing values as described in 'Data Preprocessing' section.

Then I select variables that are the most important for the problem in hand. For this purpose I will use azdias and customers datasets, formulate the problem as binary classification and use random forest for prediction. Random forest provides feature importance and identifies the top features.

I visualize top-10 features and and compare their distribution for customer and general population as described above in section 'Exploratory Visualization'.

The I will make segmentation of general population based on the selected top variables. I will use one-hot encoding (OHE) for all features and apply a clustering algorithm. Before clustering it is desirable to perform dimensionality reduction. The widely known and used PCA is designed to work with numerical, continuous data. In our case with categorical variables I consider it as not applicable and strongly oppose using it. The natural way of handling OHE data (or small integer data, like count data) originally comes from the area of text preprocessing - Latent Dirichlet Allocation (LDA). I use this method for dimensionality reduction and show that it also provides soft clustering. By using argmax soft clustering can always be converted to hard clustering.

For mail order problem I will use random forest as benchmark and feed-forward neural network as described above.

I propose to build a feed-forward neural network for solving the binary classification problem. I will use sigmoid activation in the last layer and binary cross entropy as loss function. To handle categorical variables I will use 2-dimensional embedding for each variable. To handle ordinal variables I will use 1-dimensional embedding for each variable. This approach to handling the categorical and ordinal values is the main reason why I prefer neural network. 1- and 2-dimensional embeddings allow easy visualization of each embedding. This allows comparison of each embedding with our human understanding of the categories.

Benchmark

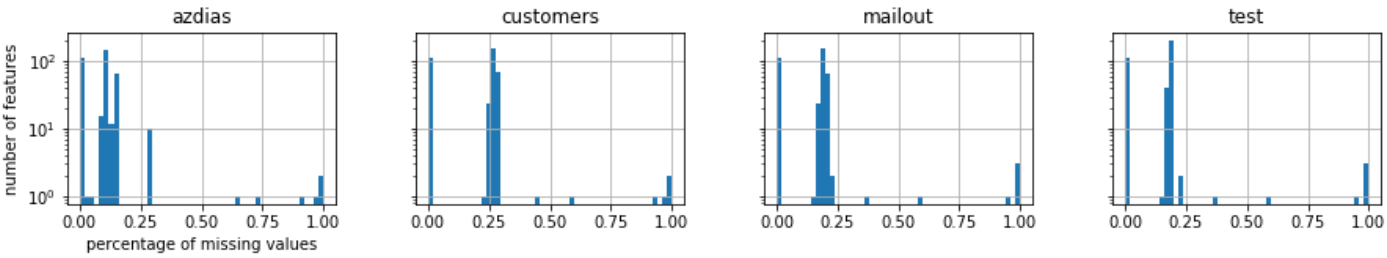
No benchmark model is available for this specific case. Therefore I will build random forest model with original features as my benchmark

Data Preprocessing

Columns that are common for the 4 datasets (azdias, customers, mailout_train and test) contain a lot of missing values. Here is an example for several columns and histograms for all columns

Prozent of missing values in each dataset

	azdias	customers	mailout	test
AGER_TYP	0.000000	0.000000	0.000000	0.000000
AKT_DAT_KL	0.082470	0.243128	0.162213	0.160834
ALTER_HH	0.082470	0.243128	0.162213	0.160834
ALTER_KIND1	0.909048	0.938607	0.953727	0.953004
ALTER_KIND2	0.966900	0.973389	0.982403	0.982210
ALTER_KIND3	0.993077	0.993347	0.995950	0.995307
ALTER_KIND4	0.998648	0.998769	0.999046	0.999089
ALTERSKATEGORIE_FEIN	0.295041	0.270501	0.189819	0.189527
ANZ_HAUSHALTE_AKTIV	0.104517	0.260509	0.181020	0.178064



The histograms above shows that nearly each column has explicitely unknown values (empty strings in CSV file).

Around 100 columns in each dataset have few missing values and over 200 columns have around 25-25% missing values.

At the same time, from data documentation we know, that

-1 always encodes unknown values

0 sometime encodes unknown values and sometimes is meaningful

9 sometime encodes unknown values and sometimes is meaningful

Therefore I replace missing values with -1.

Another important question in preprocessing is handling of categorical data. Here I use different techniques at different steps.

For feature selection with Random Forest I keep features as they are. This allows to estimate the importance of the feature as the whole feature (as not its separate categories at it would be case with OHE features).

For dimensionality reduction and clustering I use OHE in order to guarantee that the distance between different points (different people in our case) is meaningful.

The only technique that captures the distance between categorical variables even better is the concept of encoding. I use this technique in side the Neural Network at the classification step.

Implementation

My implementation closely follows the procedure described in 'Algorithms and Techniques'. For the most part the algorithms available in scikit-learn package are directly used.

The nuance with LDA is the necessity to use the sparse matrix as input. This conversion is done with scipy package.

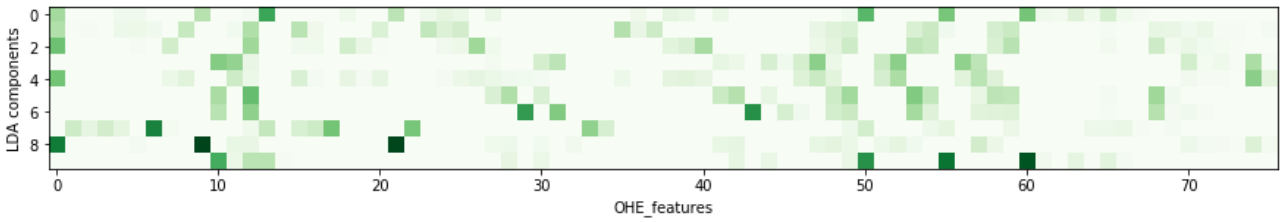
For Neural Network I use tensorflow, keras API. With 5-30 categories for each categorical variable, 2-dimensional embedding for each variable is appropriate. Since the input data are imbalanced, I set class_weight={0:0.01, 1:0.99} during fit of the model.

The progress of training and comparison of train/validation results is observed with TensorBoard.

Results of clustering

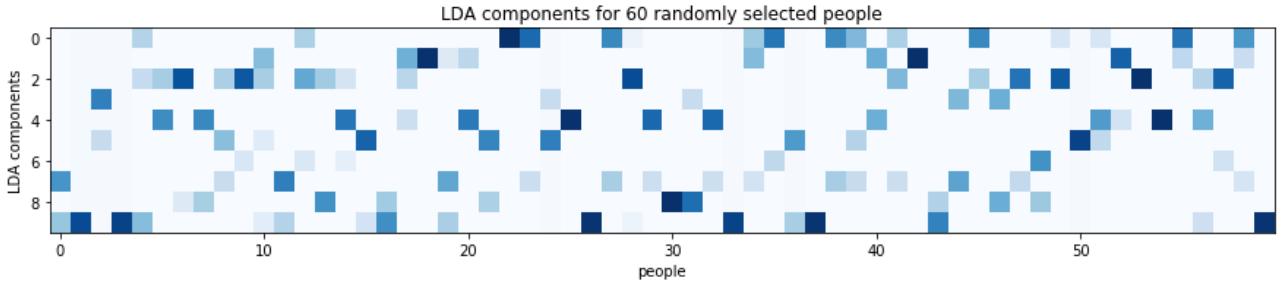
Normalized LDA components represent feature distributions over dimensions. Visulalization below shows, that each LDA component is dominated by a few features and few categories in each feature.

For example, lda_0 is people over 60 (ALTERSKATEGORIE_GROB_4). lda_1 is people from 46 to 60 (ALTERSKATEGORIE_GROB_3) with consumption type 'Universal' (D19_KONSUMTYP _1). Further interpretations is impossible due to missing documentation on other features



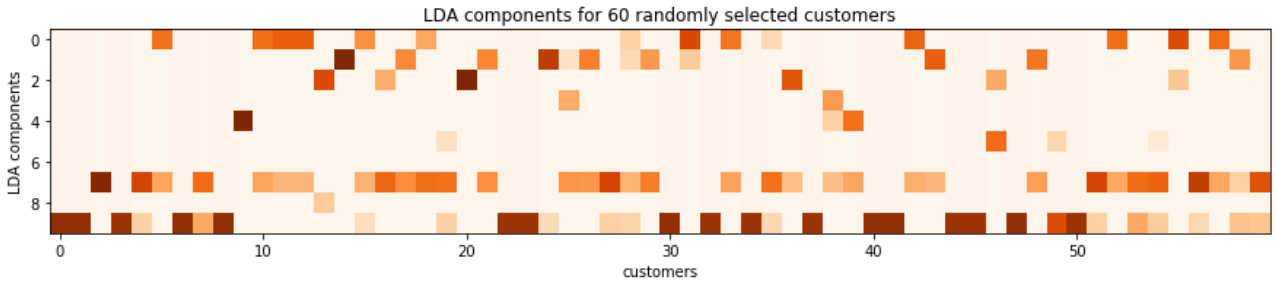
LDA can also be interpreted as soft clustering

Below I show the LDA components (y-axis) for 60 randomly selected people (blue palette) from general population (x-axis).



With each person having several nonzero components, LDA can be considered as soft clustering., The components are probabilities of belonging to each cluster. That is each person belongs to specific group (LDA component) with specific probability.

The following picture is for 60 randomly selected customers (orange palette)



In most cases each person has one dominating component, which allows to convert soft clustering to hard clustering.

Already with these two pictures, one can easily see that most of customers (unlike the general population) belong the 'lda_7' cluster. Customers are also overrepresented in cluster 'lda_9'

These two clusters are characterized by the following top-5 features.

	lda_7
D19_KONSUMTYP_MAX_2	0.170352
D19_KONSUMTYP_3	0.108193
VK_ZG11_1	0.105799
VK_DISTANZ_1	0.091066
ALTERSKATEGORIE_GROB_4	0.056075

	lda_9
CJT_TYP_3_5	0.206232
CJT_TYP_5_5	0.181873
CJT_TYP_6_5	0.157413
D19_KONSUMTYP_MAX_9	0.134490
ALTERSKATEGORIE_GROB_4	0.064397

Both groups are people above 60 (ALTERSKATEGORIE_GROB_4)

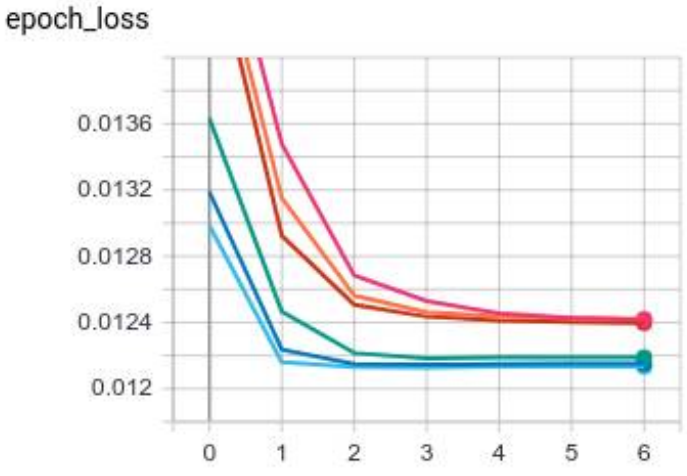
Unfortunately, other features are not described in Excel documentation files.

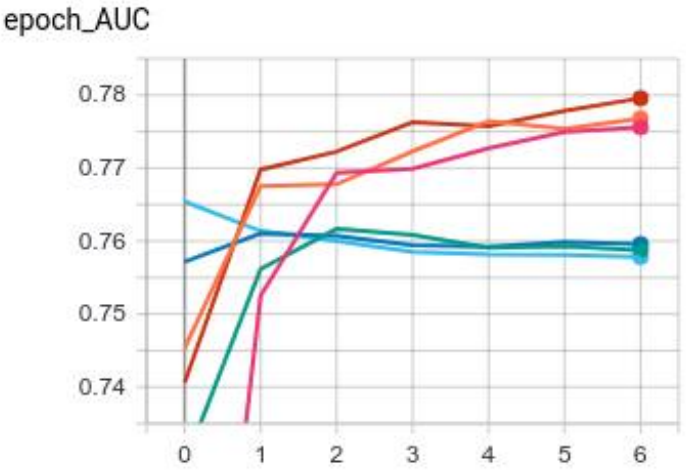
Refinement

In training the neural net, I compared feed-forward architectures with different number of layers and hidden units. I quickly found, that even with few hidden units the neural net saturates very quickly (within 2-3 epochs) and the result on validation set is sensitive to the validation data. Using top-4 features

- D19_SOZIALES,
- D19_KONSUMTYP_MAX,
- D19_KONSUMTYP,
- RT_SCHNAEPPCHEN

leads to the best model as estimated by validation data. Using more features make the training process unstable and does not improve results. Making the network deeper does not improve result either.





Model Evaluation and Validation

Model is evaluated by AUC metric calculated via stratified 5-fold validation. AUC is sensitive only to the order of predicted probabilities, not their actual values. This is a useful property when working with imbalanced data set and using class weight for balancing during the training stage. As

I already mentioned, the validation AUC appear to be sensitive to the train/validation split used. In order to make comparison with the benchmark meaningful, for keras I use the same split (StratifiedRandomSplit with the same split) as in the benchmark.

Justification

The benchmark model (Random Forest) leads to AUC=0.737±0.025.
Neural Network with 1 hidden layer leads to AUC=0.769±0.022.

Histograms for predicted probabilities show that those vary in a range from ca. 0.2 to ca 0.8, thus allowing easy selection of top-percentage of people for targeted mail.