Daiichi – Coding Challenge

Documentation

Approach:

I’ve completed the Daiichi-Sankyo coding challenge by doing the following steps

Step 1: Printing value counts for all columns within the training dataset and

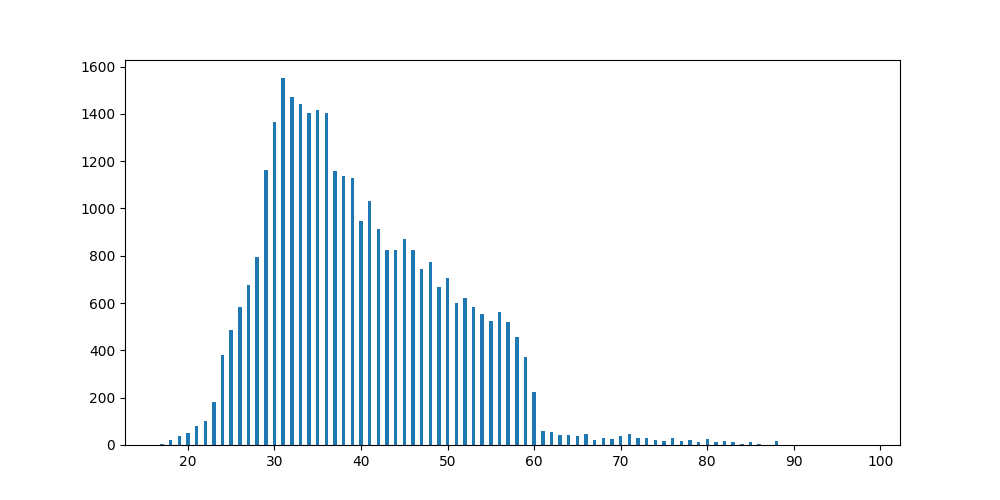
Step 2: Gather first insights from Step 1

Step 3: Correlation between features → Answer question 1

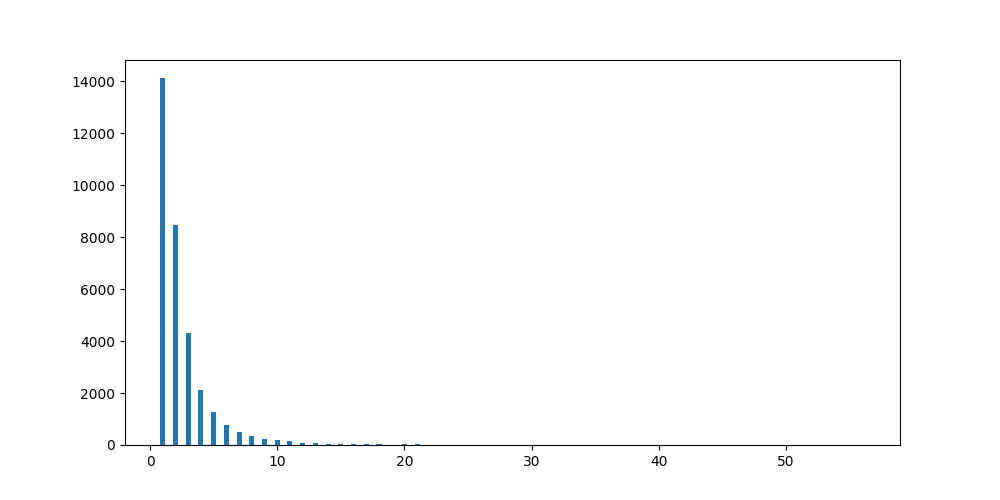
Step 4: Feature importance → Answer question 3

Step 5: Modelling approach

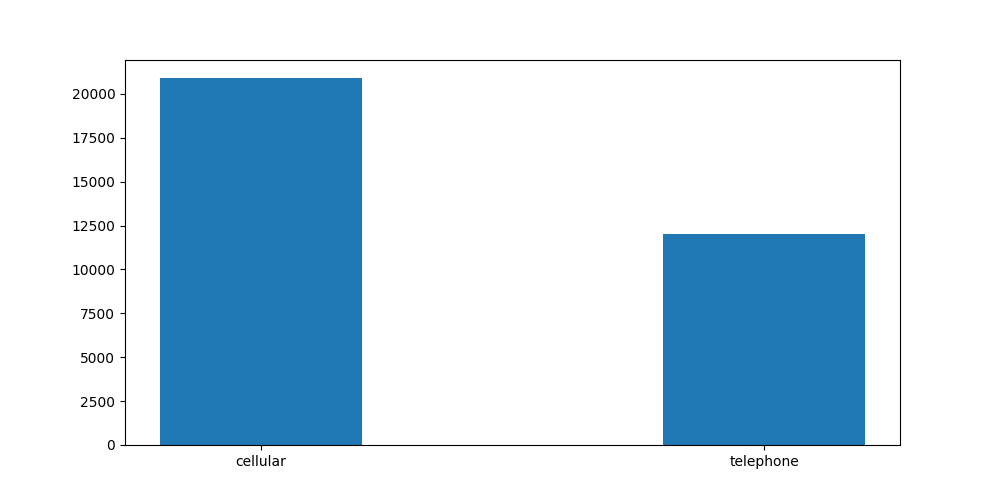
Step 1:



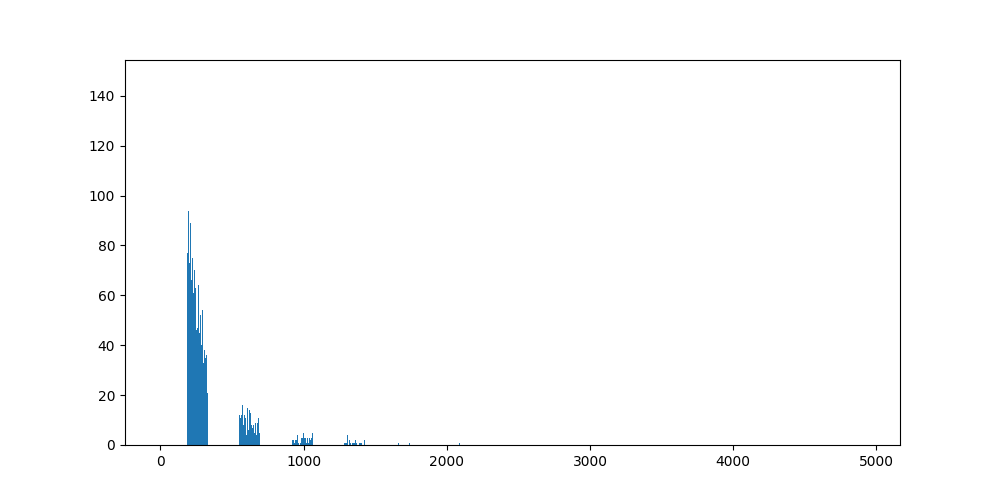
|  |  |
| --- | --- |
|  | age |
| count | 32910.0 |
| mean | 40.01409905803707 |
| std | 10.402947714925428 |
| min | 17.0 |
| 25% | 32.0 |
| 50% | 38.0 |
| 75% | 47.0 |
| max | 98.0 |



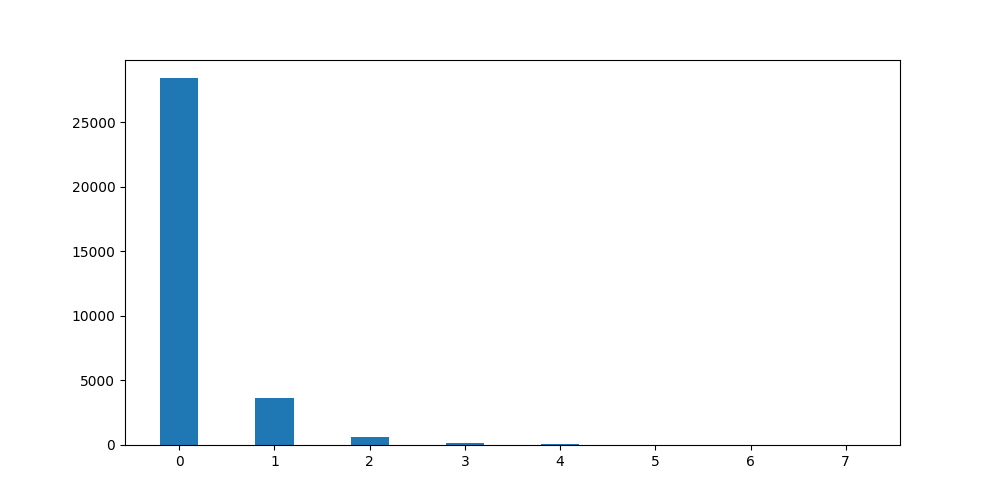
|  |  |
| --- | --- |
|  | campaign |
| count | 32910.0 |
| mean | 2.560619872379216 |
| std | 2.753336387827619 |
| min | 1.0 |
| 25% | 1.0 |
| 50% | 2.0 |
| 75% | 3.0 |
| max | 56.0 |



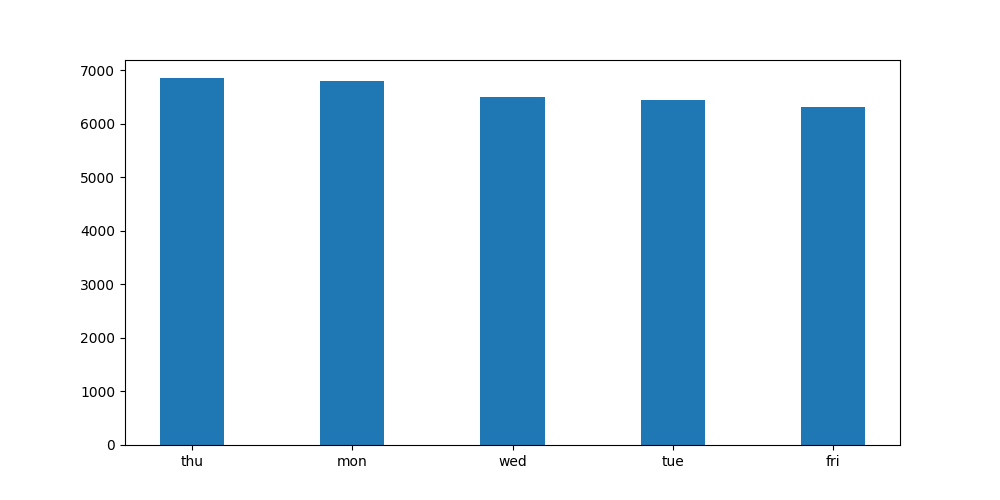
|  |  |
| --- | --- |
|  | contact |
| count | 32910.0 |
| cellular | 20890 |
| telephone | 12020 |



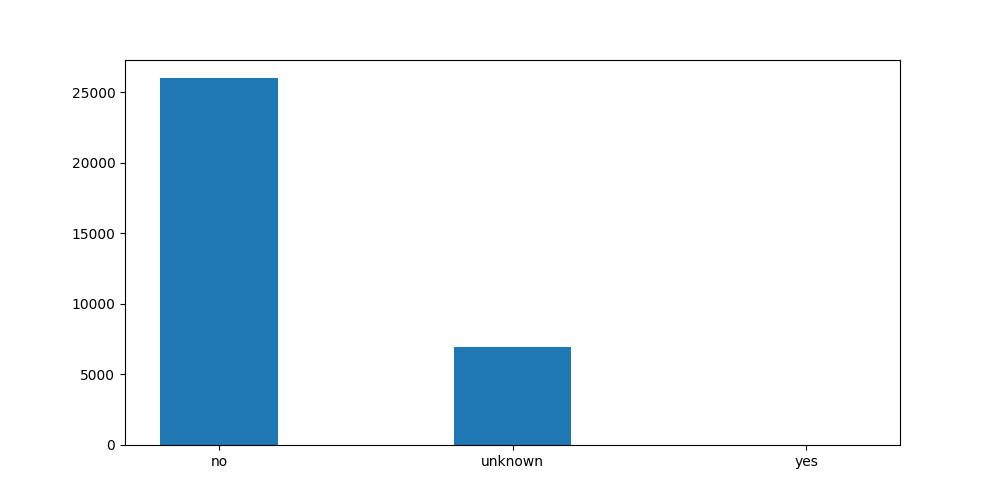
|  |  |
| --- | --- |
|  | duration |
| count | 32910.0 |
| mean | 258.1643269522941 |
| std | 259.07025960950676 |
| min | 0.0 |
| 25% | 103.0 |
| 50% | 180.0 |
| 75% | 319.0 |
| max | 4918.0 |



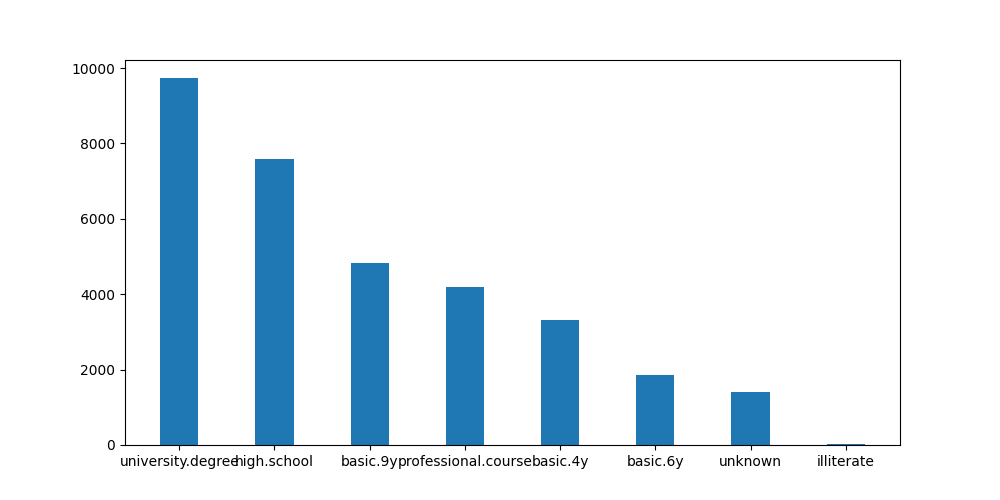
|  |  |
| --- | --- |
|  | previous |
| count | 32910.0 |
| mean | 0.17481008811911272 |
| std | 0.49921137366395457 |
| min | 0.0 |
| 25% | 0.0 |
| 50% | 0.0 |
| 75% | 0.0 |
| max | 7.0 |



|  |  |
| --- | --- |
|  | day\_of\_week |
| count | 32910.0 |
| mon | 6802 |
| tue | 6439 |
| wed | 6508 |
| thu | 6849 |
| fri | 6312 |

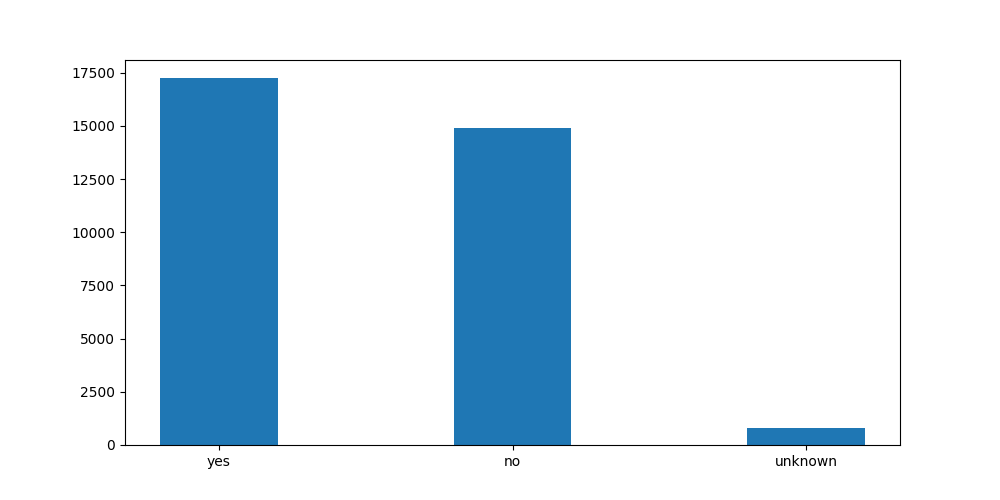


|  |  |
| --- | --- |
|  | default |
| count | 32910.0 |
| yes | 3 |
| no | 25975 |
| unknown | 6932 |

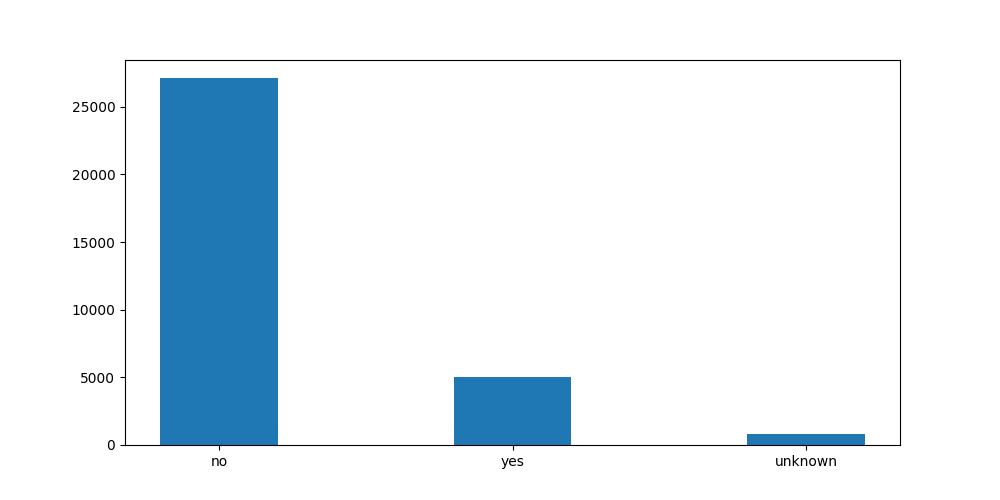


|  |  |
| --- | --- |
|  | education |
| count | 32910.0 |
| University.degree | 9727 |
| High.school | 7585 |
| Basic.9y | 4818 |
| Professional.course | 4184 |
| Basic.4y | 3322 |
| Basic.6y | 1863 |
| unknown | 1395 |
| illiterate | 16 |

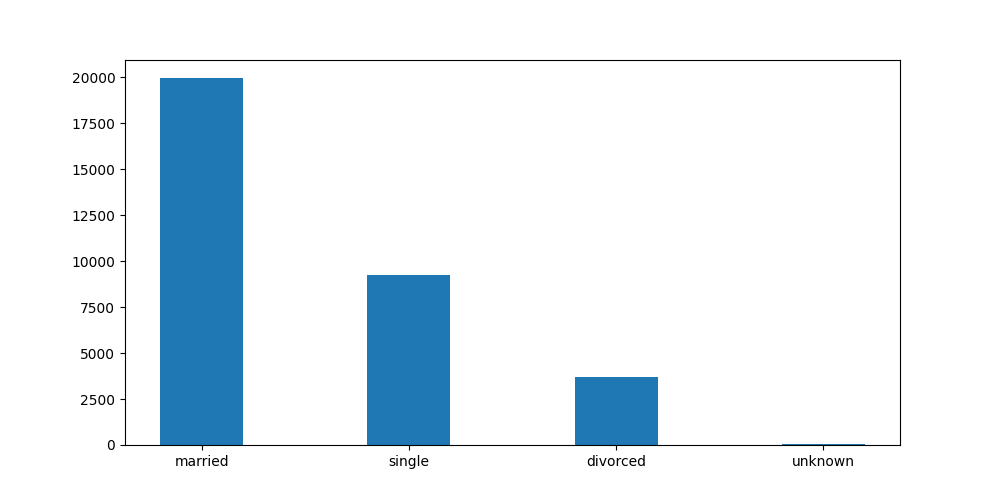
|  |  |
| --- | --- |
|  | job |
| count | 32910.0 |
| Admin. | 8305 |
| blue-collar | 7430 |
| technician | 5392 |
| service | 3192 |
| management | 2343 |
| retired | 1364 |
| entrepreneur | 1159 |
| self-employed | 1098 |
| housemaid | 855 |
| unemployed | 798 |
| student | 710 |
| unknown | 264 |



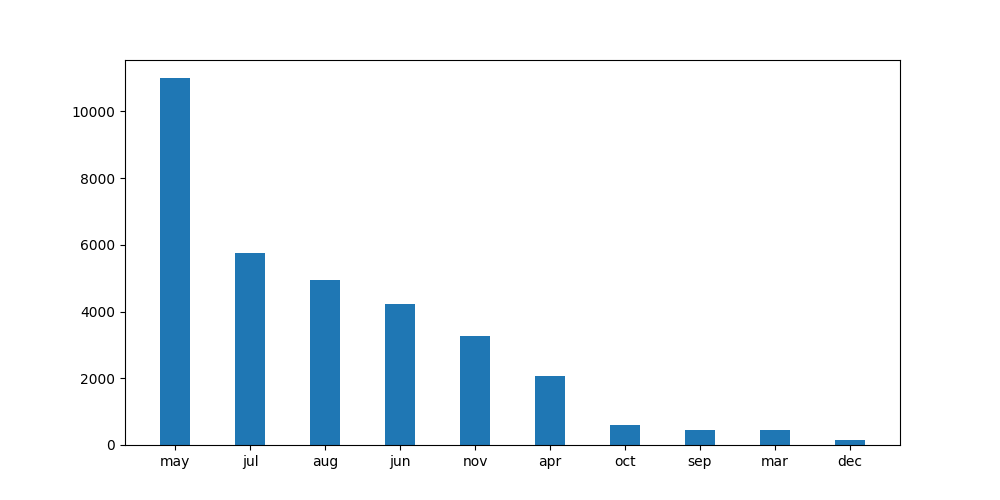
|  |  |
| --- | --- |
|  | housing |
| count | 32910.0 |
| yes | 17236 |
| no | 14879 |
| unknown | 795 |



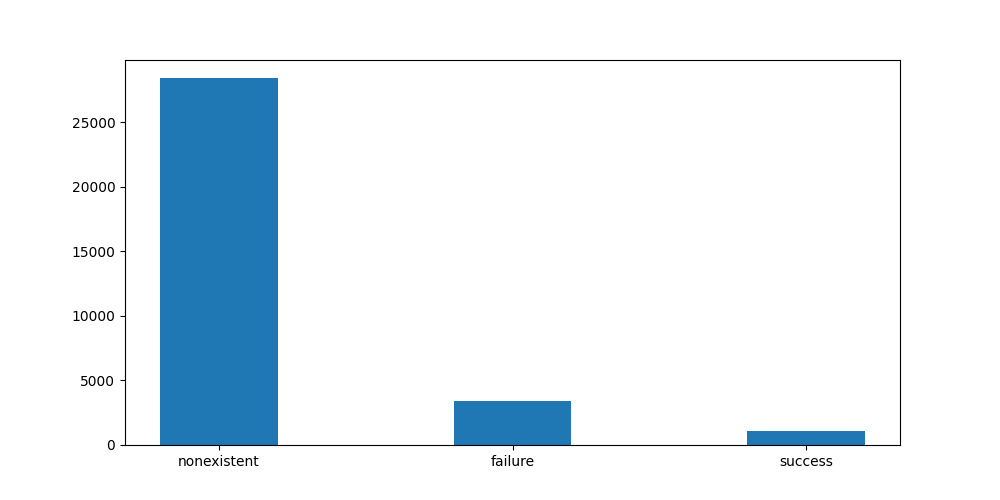
|  |  |
| --- | --- |
|  | loan |
| count | 32910.0 |
| yes | 5016 |
| no | 27099 |
| unknown | 795 |



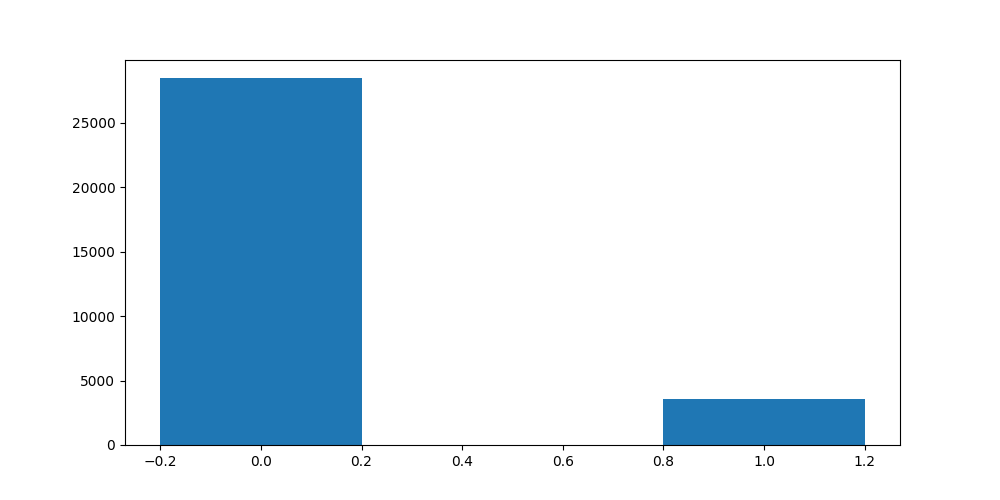
|  |  |
| --- | --- |
|  | marital |
| count | 32910.0 |
| married | 19929 |
| single | 9245 |
| unknown | 65 |
| divorced | 3671 |



|  |  |
| --- | --- |
|  | job |
| count | 32910.0 |
| may | 10993 |
| jul | 5753 |
| aug | 4946 |
| jun | 4242 |
| nov | 3263 |
| apr | 2083 |
| oct | 587 |
| sep | 464 |
| mar | 436 |
| dec | 143 |



|  |  |
| --- | --- |
|  | poutcome |
| count | 32910.0 |
| nonexistent | 28280 |
| failure | 3426 |
| success | 1104 |



|  |  |
| --- | --- |
|  | y |
| count | 32910.0 |
| no | 29203 |
| yes | 3707 |

Step 2:

Insights:

- Training set is unbalanced → oversampling needed

- Many categorical features → one-hot encoding

- Marketing has a high focusing on people that:

- are in their 30s

- are married

- have no loans

- have high education

- have well paying jobs

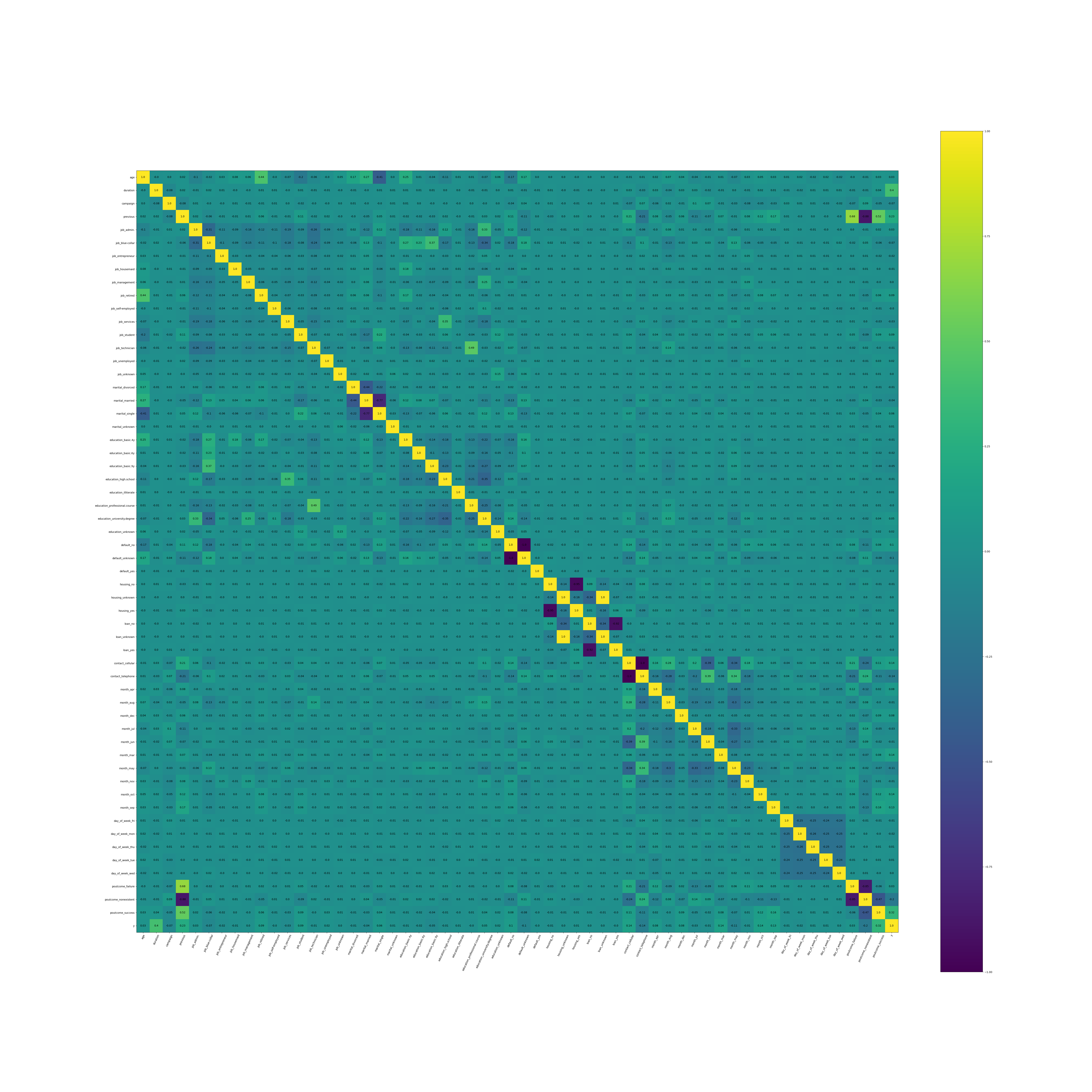
- and have not been called in the past

- Marketing team often calls between May and August. Maybe because people are generaly in a better mood during spring and summer.

- marital, loan, housing and job have only few unknowns. These values can be dropped if they make up for less then 5% of the total rows in the dataset.

Step 3:

The correlation-matrix for all features after one-hot-encoding the categorical features is rather large. A higher quality image of the matrix can be found in the graphics folder.



Answer question 1:

We see some correlations between variables.

1. Previous outcomes are correlated to the number of previous attempts. Because only if there was at least 1 previous attempt, the previous outcome can be a failure or a success.

2. Being retired is correlated to the age of a person. We see the same for people being married or divorced but less significant.

3. Some jobs and educations are correlated

- People with university degree are often admins or work in management

- People who visited a professional course are often technicians

- People with high scholl degrees often work in service

- People with basic 9, 6 or 4 year education often work a blue-collar job

4. Unknown housing is fully correlated with unknown loan

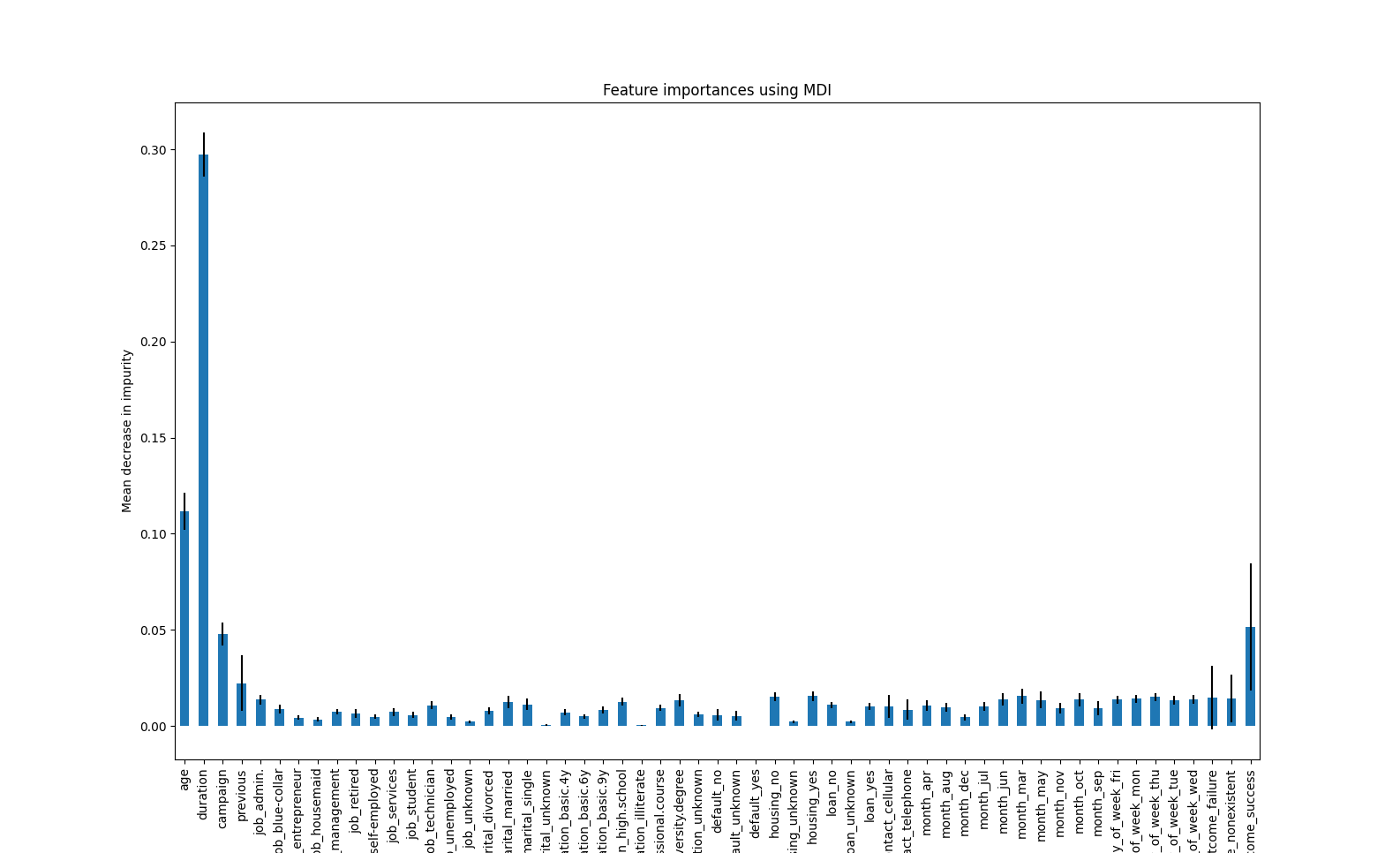
5. There is slight a correlation between people being contacted via telephone and people being contacted in the month of may and june

6. Also people contacted via telephone have more often a non existent previous outcome, maybe because marketing could not reach them. Peolpe contacted by cellphone have more often a previous outcome that’s either success or failure.

7. The target y is mainly correlated to duration, previous attempts and if previous attempts were successfull. Some correlation also exists between y and people contacted via cellphone, and in some specific months (March, October, September). More about most important features in the next Step.

Step 4:

Most important features for classification.



Answer question 3:

I used a random forest classifier to evaluate the improtance of features for the classification. Results show that age, duration, campaign. poutcome\_success and previous have most influence on the classification results. Some of them already showed some correlation with the target in the correlation matrix.

Step 5:

For the modelling I used a grid-search approach to find the best hyperparameters for my list of classification models. I evaluate the models based on the recall because I think it is most important to classify all successes correcly.

Before the grid-search I perform the following preprocessing steps:

1. Remove unknown values as long as the total of removed lines is not more than 5% of the total rows of the dataset. This will remove some features created by one-hot-encoding but when we look into the feature importance plot in Step 4 we see that these features are very unimportant for the classification. Ignoring them should be fine

2. Make the target binary (1,0 instead of yes, no)

3. One-hot-encoding of categorical features

4. Balance dataset with SMOTE oversampling. We could also duplicate the rows of the minority class but it would lead a less generalized model.

5. Run a standardscaler to make the features have mean 0 and standard deviation 1. This improves the performance of some models (SVM, neural networks, logistic regression)

6. Run principle component analysis (PCA) to reduce the dimensions of the classification problem. Alternatively we could also just remove features with low importance

Some results:

Approach 1:

- Columns with ‚unknown‘ values are not dropped

- All categorical values are getting one-hot encoded

- Classifiers: GradientBoosting

RandomForrest

AdaBoost

- High accuracy low recall

Approach 2:

- Columns with ‚unknown‘ values are dropped → 2.4% of data is lost

- Categorical values loan and housing are not getting one-hot encoded but changed to binary values

- Classifiers: GradientBoosting

RandomForrest

AdaBoost

- High accurarcy low recall

Approach 3:

- Columns with ‚unknown‘ values are dropped → 2.4% of data is lost

- Categorical values loan and housing are not getting one-hot encoded but changed to binary values

- Apply balancing of tagret variables by duplicating ‚yes‘ rows

- Classifiers: GradientBoosting

RandomForrest

AdaBoost

-High accuracy, high recall → possible that datapoints used for evaluation have already been seen during training

Approach 4:

- Columns with ‚unknown‘ values are dropped → 2.4% of data is lost

- Categorical values loan and housing are not getting one-hot encoded but changed to binary values

- Apply balancing of tagret variables using SMOTE

- Classifiers:Logistic Regression

GradientBoosting

RandomForrest

AdaBoost

-High accuracy, high recall

Approach 5:

- Columns with ‚unknown‘ values are dropped → 2.4% of data is lost

- Categorical values loan and housing are not getting one-hot encoded but changed to binary values

- Apply balancing of tagret variables using SMOTE

- Prediction pipeline with StandardScaler

- Classifiers:Logistic Regression

GradientBoosting

RandomForrest

AdaBoost

- High accuracy high recall