Bank Telemarketing Analysis

...

Agenda

- Business Understanding
- Data Exploration and Preparation
- Model Building
- Hyper-parameter Tuning and Model Evaluation
- Result / Outcomes

Business understanding

- Problem Statement: Improve marketing campaign of a Portuguese bank by analyzing their past marketing campaign data and recommending which customer to target
- Problem Motivation: By devising such a prediction algorithm, the bank can better target its customers and better channelize its marketing efforts
- Banco de Portugal offered their clients fixed-term products such as CDs. Data was collected about each client, type of contact, and outcome.
- What can this data tell us about marketing success for this campaign?
- Can these data science techniques be applied to other areas?

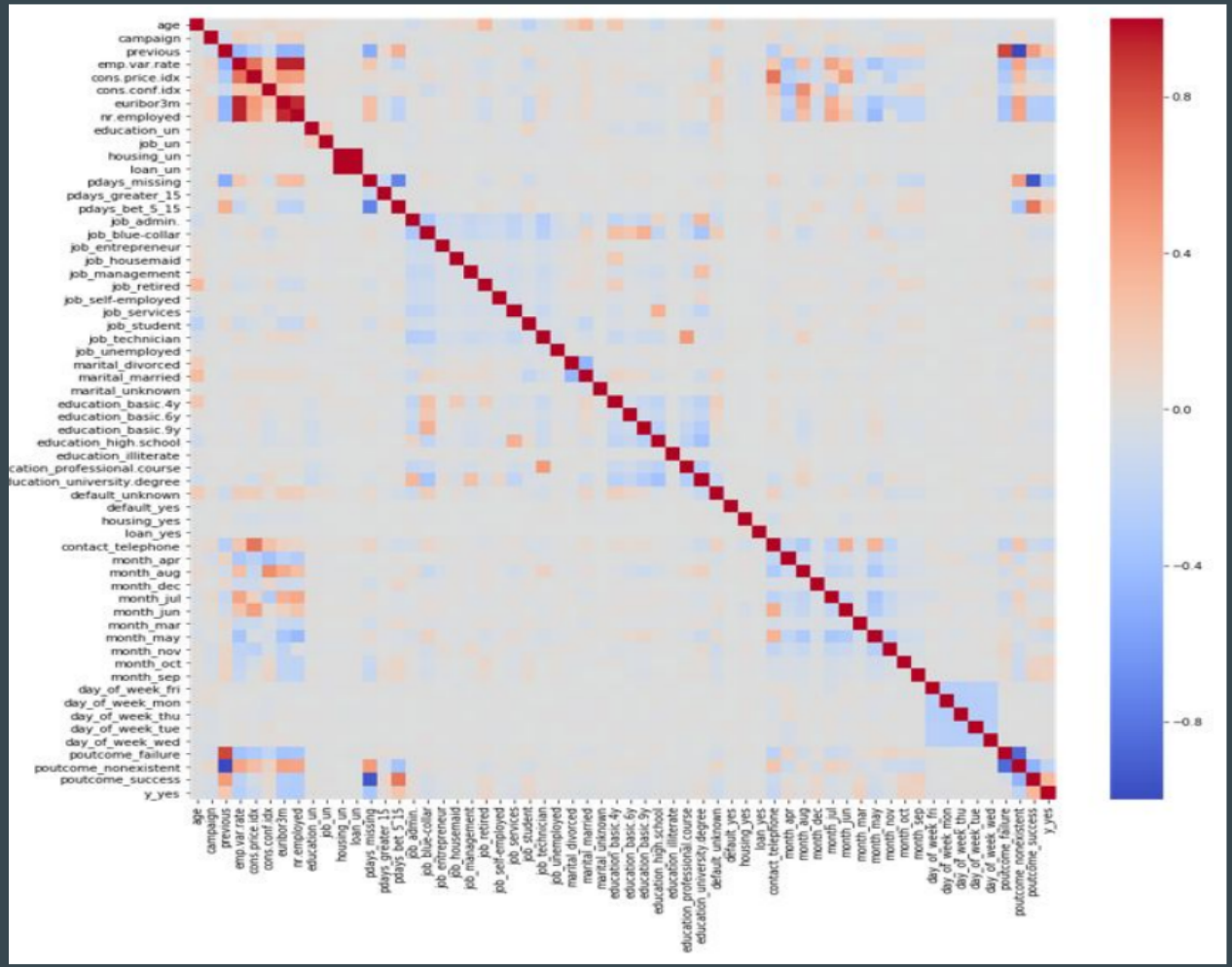
Data Exploration and Preparation

Data Exploration and Preparation (2/2)

- Many features had missing values. How do we handle this?
- For categorical features, imputation using other independent variables. For example,
cross-tabulation between 'job' and 'education'; 'age' and 'job'; 'home ownership' and 'loan
status.'
- Among numerical features, fortunately only column (‘pdays’) had any missing values.
Unfortunately, missing values made up the majority of the column.
- To handle this, ‘pdays’ was converted from a numerical feature to a categorical feature using
buckets: < 5 days, 6-15 days, etc.
- Heatmap using seaborn package was created to show us any particularly strong correlations
between the independent variables and the target variable outcome.

Data Exploration and Preparation (1/2)

- All coding done in Python 3.
- Extensive use of pandas, numpy, matplotlib, as well as seaborn and sklearn packages.
- Dataset contained 20 different features on more than 41,000 clients.
- Features were both categorical and numerical. Target variable was binary (“Yes” or “No”).
- Pandas package was imported and a dataframe was created.
- Categorical variables were looked at first. Visualizations were created using the seaborn package


age
campaign
previous
emp.var.rate
cons.price.idx
cons.conf.idx
euribor3m
nr.employed
education_un
job_un
housing_un
loan_un
pdays_missing
pdays_greater_15
pdays_bet_5_15
job_admin.
job_blue-collar
job_entrepreneur
job_housemaid
job_management
job_retired
job_self-employed
job_services
job_student
job_technician
job_unemployed
marital_divorced
marital_married
marital_unknown
education_basic.4y
education_basic.6y
education_basic.9y
education_high.school
education_illiterate
cation_professional.course
lucation_university.degree
default_unknown
default_yes
housing_yes
loan_yes
contact_telephone
month_apr
month_aug
month_dec
month_jul
month_jun
month_mar
month_may
month_nov
month_oct
month_sep
day_of_week_fri
day_of_week_mon
day_of_week_thu
day_of_week_tue
day_of_week_wed
poutcome_failure
poutcome_nonexistent
poutcome_success
y_yes
0.8
0.4
0.0
−0.4
−0.8

Model Builiding

...

Model Building (1/2)

Logistic Regression

- sklearn.linear_model.LogisticRegression
- Its a classification model though name is Logistic regression
- Fits a sigmoid function to a data
- Outputs probability which is in [0,1] range unlike linear models.

Decision Tree

- sklearn.tree.DecisionTreeClassifier
- Simple to understand and effective
- Splits the data at every node based on one feature
- Uses information gain as measure for split

Model Building (2/2)

Random Forest

- Sklearn.ensemble.RandomForestClassifier
- Constructs multiple decision trees and takes the mode of those trees for an example to make the final decision
- Individual Trees are intentionally over fit and validation set is used to optimize the forest level parameters

AdaBoost and Gradient Boosting

- sklearn.ensemble.GradientBoostingClassifier
- sklearn.ensemble.AdaBoostClassifier
- Many decision trees with single split are constructed
- Instance which is hard to classify gets more attention by giving it a larger weight
- Gradient Boosting is generalized version of AdaBoost
- One weak learner is added at a time and existing weak learners remain unchanged

Hyper-Parameter Tuning and Model Evaluation

Hyper-parameter tuning and Model Evaluation

- Used mean AUC of 5 fold cross validation as the metric for evaluation
- Choose the model with highest mean AUC

Model	Hyper-parameters Tuned	Optimal hyper-parameters	Mean AUC
Logistic Regression	C: Regularization Coefficient Type: L1, L2	C = 0.1 L1 Logistic Regression	0.7903
Decisions Trees	Min. Split Value and Min. Leaf Value	Min. Split Value = 1110 Min. Leaf Value = 132	0.7919
Random Forests	Min. Split Value and Min. Leaf Value	Min. Split Value = 189 Min. Leaf Value = 7	0.7979
Gradient Boosted Trees	Min. Split Value and Min. Leaf Value	Min. Split Value = 85 Min. Leaf Value = 37	0.8006
AdaBoost	Number of Estimators	Number of Estimators = 1000	0.8157

Results / Outcome

...

Best Model and Feature Importance

- Best Model: AdaBoost with 1000 estimators.
- Obtained an AUC of 0.8036 on the test set.
- Below is the Feature Importance Chart for the AdaBoost Model:

Recommendations to the Marketing Team

Significant Variables	Recommendations
Libor Rate, Con.Price.Idx, Con.Conf.Idx	• Collaborate with the economic experts • Be a fast mover, capture customers before the competitors capture them
Age	• Target relatively Old Age people • Convey Peace of mind, Safe investment, steady income source as the value proposition
Duration, Mode of Contact: Telephone	• Try to engage customers and have longer calls • Preferably use Telephone as the mode of contact
Campaign	• Prioritize those customers to who were part of the previous marketing campaigns.