## Bank Marketing Campaign

İlke Candan Bengi

Submitted to: Data Glacier

#### Agenda

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

## Business Understanding

A term deposit is a type of investment that a bank or other financial institution offers with a fixed rate (typically better than just opening a deposit account), with the promise that your money will be returned at a certain maturity date.

The sample population should be broken down into groups that are most likely to open term deposit accounts.

# Data Exploration and Preparation Data Exploration and Preparation (1/2)

- All programming was done in Python 3 on Jupyter notebook
- Extensive usage of the seaborn and sklearn packages, as well as pandas, numpy, and matplotlib. Dataset contained 20 different features on more than 41,000 clients.

- 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.



### 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, crosstabulation 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.
- Heatmap using seaborn package was created to show us any particularly strong correlations between the independent variables and the target variable outcome.

#### Correlation Heatmap :

		B.											
age	1	0.0079	-0.0024	0.033	0.049	0.05	-0.05	-0.036	0.13	-0.036	-0.065	0.049	
duration	0.0079	1	-0.068	0.046	0.019	0.045	-0.023	0.013	-0.0095	-0.029	-0.04	0.39	
campaign	-0.0024	-0.068	1	-0.043	-0.081	-0.052	0.16	0.13	-0.012	0.14	0.15	-0.069	
pdays	0.033	0.046	-0.043	1	0.48	0.74	-0.22	-0.031	0.074	-0.25	-0.32	0.27	
previous	0.049	0.019	-0.081	0.48	1	0.53	-0.4	-0.18	-0.028	-0.44	-0.49	0.23	
poutcome	0.05	0.045	-0.052	0.74	0.53	1	-0.25	-0.065	0.095	-0.28	-0.35	0.32	
emp.var.rate	-0.05	-0.023	0.16	-0.22	-0.4	-0.25	1	0.77	0.16	0.97	0.9	-0.31	
ons.price.idx	-0.036	0.013	0.13	-0.031	-0.18	-0.065	0.77	1	0.027	0.67	0.49	-0.13	
ons.conf.idx	0.13	-0.0095	-0.012	0.074	-0.028	0.095	0.16	0.027	1	0.24	0.075	0.062	
euribor3m	-0.036	-0.029	0.14	-0.25	-0.44	-0.28	0.97	0.67	0.24	1	0.94	-0.32	
nr.employed	-0.065	-0.04	0.15	-0.32	-0.49	-0.35	0.9	0.49	0.075	0.94	1	-0.36	
У	0.049	0.39	-0.069	0.27	0.23	0.32	-0.31	-0.13	0.062	-0.32	-0.36	1	
	e Der	duration	campaign	pdays	previous	pontcome	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	rr.employed	>	

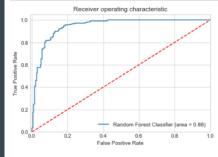
## Model Building

Random t	Forest	Report in precision	test set recall	f1-score	suppor
	0	0.96	0.99	0.97	380
9	1	0.58	0.20	0.30	20
9					
aco 1	curacy			0.95	401
	ro avg	0.77	0.60	0.64	401
weighte 1	ed avg	0.94	0.95	0.94	401

Random Forest Classifer perform best

## Results / Outcome

```
from sklearn.metrics import roc_auc_score
from sklearn.metrics import roc_curve
logit_roc_auc = roc_auc_score(y_test, rf.predict(X_test))
fpr, tpr, thresholds = roc_curve(y_test, rf.predict_proba(X_test)[:,1])
plt.figure()
plt.plot(fpr, tpr, label='Random Forest Classfier (area = %0.2f)' % logit_roc_auc)
plt.plot([0, 1], [0, 1],'r--')
plt.xlim([-0.01, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic')
plt.legend(loc='lower right")
plt.show()
```



#### Significant Variables

Duration, Education

Age

Duration, Mode of Contact: Telephone

Campaign

Thank You