

Credit worthiness evaluation using Machine Learning

Prepared by:
Vidhi Chugh
19th Aug 2020

Contents

- Business problem and data understanding
- Train and test data distribution comparison
- Exploratory Data Analysis
 - Categorical, missing values, outlier detection
 - Data Visualization
 - EDA Questions
- Machine Learning pipeline
- Model Interpretability
 - Predictions and Definition of credit worthiness learned by ML model
- Comprehensive guide to elaborate selection criteria

Business problem: Credit worthiness Evaluation

- **Credit Risk:** Risk of witnessing defaults on a debt that may arise from a borrower failing to make required payments.
- **Business Objective:** To learn from the association between the traits and attributes of different borrowers in history and their repayment status i.e. whether it resulted in Good Risk or Bad Risk
- **Target Variable:** Good (class 1, creditworthy) vs Bad (class 0, not creditworthy) Risk
- **Data* Description:** 1000 records with 10 features
- The dataset is divided into 90%-10%:
 - **Train** - 900
 - **Test** - 100

	Column	Values
Customers' attribute	Credit History	0 to 4
	Age	19 to 75
	Gender	Male and Female
	Job	0 to 3
	Housing	Own, rent and free
Loan specific attributes	Saving Accounts	little, nan, quite rich, moderate, rich
	Credit Amount	250 to 15945
	Duration	4 to 72
Target Variable	Purpose	furniture/equipment', 'radio/TV', 'car', 'business', 'education', 'repairs', 'domestic appliances', 'vacation/others'
	Risk	Good, Bad

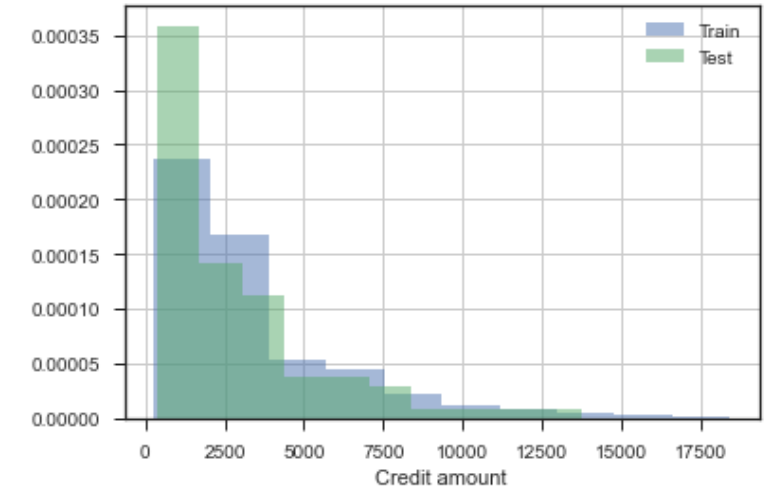
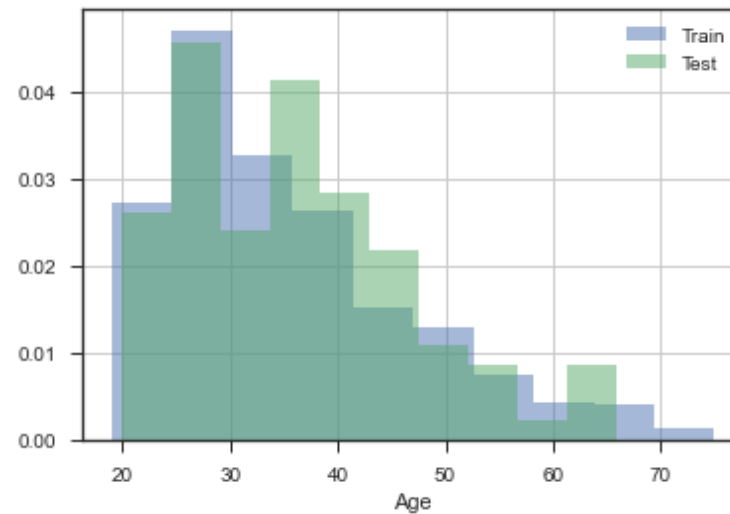
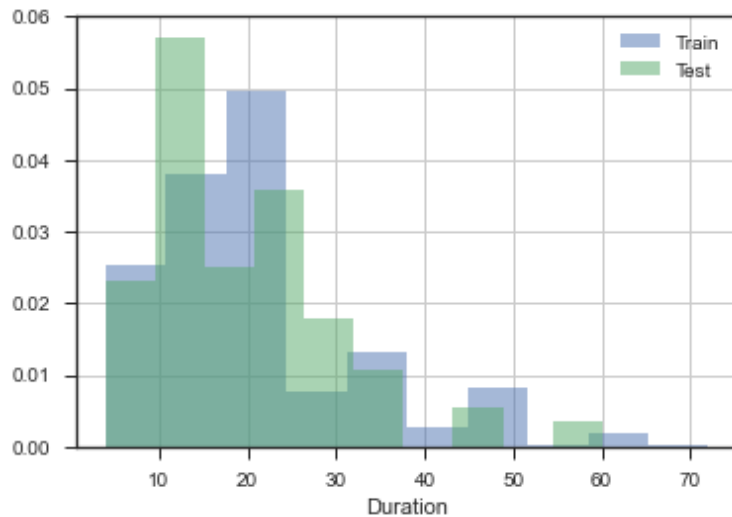
* The data downloaded from the link did not have column headers, hence found discrepancies in column names and their values. This is a trimmed version of German Credit Risk data found by web search

Generalization

Power: train and test distribution similarity

- One of the most critical assumption in ML data modelling is that **train and test dataset belong to similar distribution**, as is evident from graphs above. Note that the train data is used as a reference to **estimate** the future credit worthiness of customers, hence the ML solution is probabilistic one and not guaranteed based on past data. This emphasizes **the property of generalization of ML solution**

KS statistic is a numeric measure to check the hypothesis of whether the distributions of 2 dataset is same. For e.g. KS statistic > p value implies same distribution



Blue colour histogram is from Train data and Green colour is from Test data, the histogram plots from two distribution are sharing significant overlap with each other implying that there is no drift between the two datasets and that they are drawn from same distribution.

```
from scipy.stats import ks_2samp
ks_2samp(df_train['Age'], df_test['Age'])
ks_2samp(df_train['Credit amount'], df_test['Credit amount'])
```

```
Ks_2sampResult(statistic=1.0, pvalue=5.551115123125783e-16)
```

```
Ks_2sampResult(statistic=0.15222222222222223, pvalue=0.02820652574472937)
```

Exploratory Data Analysis

1) Missing Value Treatment

```
df_train.isnull().sum()
```


Credit History	0
Age	0
Gender	0
Job	0
Housing	0
Saving accounts	162
Credit amount	0
Duration	0
Purpose	0
Risk	0

Only 'Saving accounts' has null, looking at the documentation, it implies that customer has 'no account', hence replaced nulls accordingly

```
### So, replacing NaN with 'no account'  
df_train.loc[df_train['Saving accounts'].isnull(), 'Saving accounts'] = 'no account'
```

2) Categorical Value Treatment

```
df_dtypes = pd.DataFrame((df_credit.dtypes == 'object'), columns = ['obj_type'])  
obj_list = df_dtypes[(df_dtypes.obj_type == True) & (~df_dtypes.obj_type.isin(le_list))].index
```



```
['Gender', 'Housing', 'Saving  
accounts', 'Purpose', 'Risk']
```

3) Outlier analysis

The cut off value and corresponding number of instances falling out of 3 sigma are shown in the table to the right:
As these values look legitimate in the context of credit risk modelling, I am not treating them as outliers

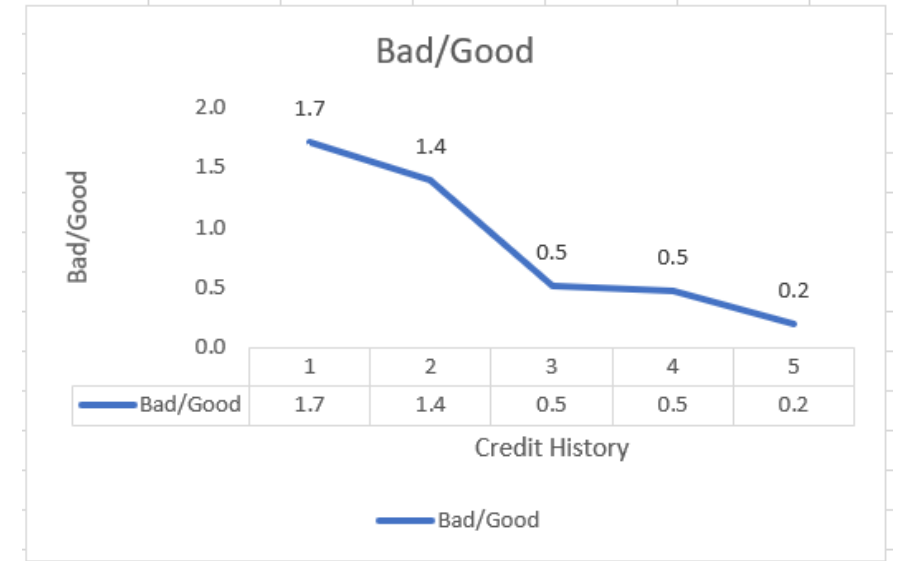
Feature	Cut off value	Instances
Credit amount	11858	21
Age	70	7
Duration	57	12

EDA Questions

Q1: More credit history is equivalent to credit worthiness

Credit History Class	Value counts Bad Risk**	Value counts Good Risk	Bad/Good
0	24	14	1.7
1	25	18	1.4
2	160	314	0.5
3	26	56	0.5
4	44	219	0.2
Class 3 and 4 (in %)	25%	44%	

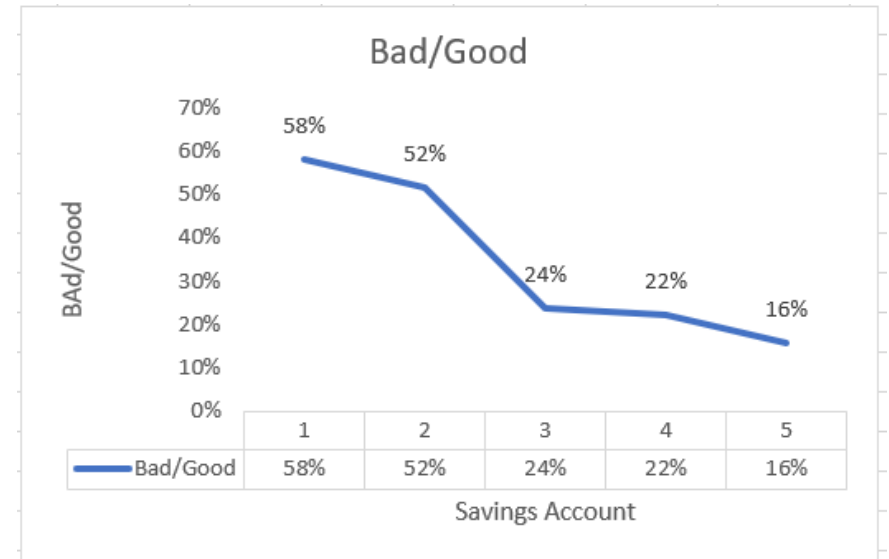
As proportion of class 3 and 4 increases from Bad Risk (25%) to Good Risk (44%) cases, the EDA signifies that credit history brings credit worthiness*. Further, the Bad/Good ratio in last column signifies that bad risk decreases as credit history improves



Q2: More saving accounts equivalent to more credit worthy?

Saving Accounts Class	Value counts Bad Risk	Value counts Good Risk	Bad/Good
0	199	343	58%
1	33	64	52%
2	31	131	24%
3	10	45	22%
4	6	38	16%
Class 3 and 4 (in %)	6%	13%	

As proportion of class 3 and 4 increases from Bad Risk (6%) to Good Risk (13%) cases, we can infer that higher saving accounts class indicate better credit worthiness. Further, the proportion of Bad to Good cases across different saving accounts also shows that Bad Risk decreases with the higher Saving Accounts class.



- Clear definition of credit history and saving accounts is needed to assert this pattern and conclude the inference
- ** Wherever % sign is not written is the count of instances in that bin

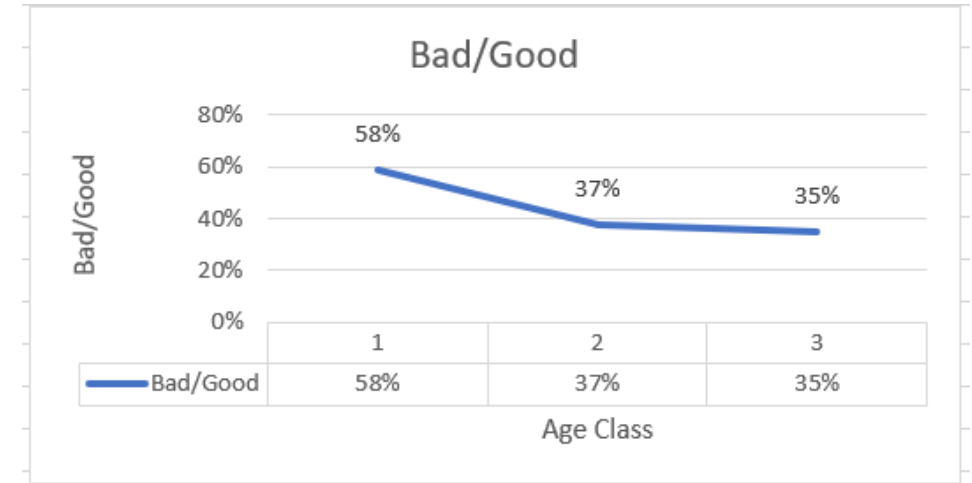
EDA Questions -- Continued

Q3: Are young people more credit worthy?

Dividing Age variable into following 3 brackets:

```
df_train.Age[df_train.Age <= 30] = 0
df_train.Age[(df_train.Age > 30) & (df_train.Age < 45)] = 1
df_train.Age[(df_train.Age >= 45)] = 2
```

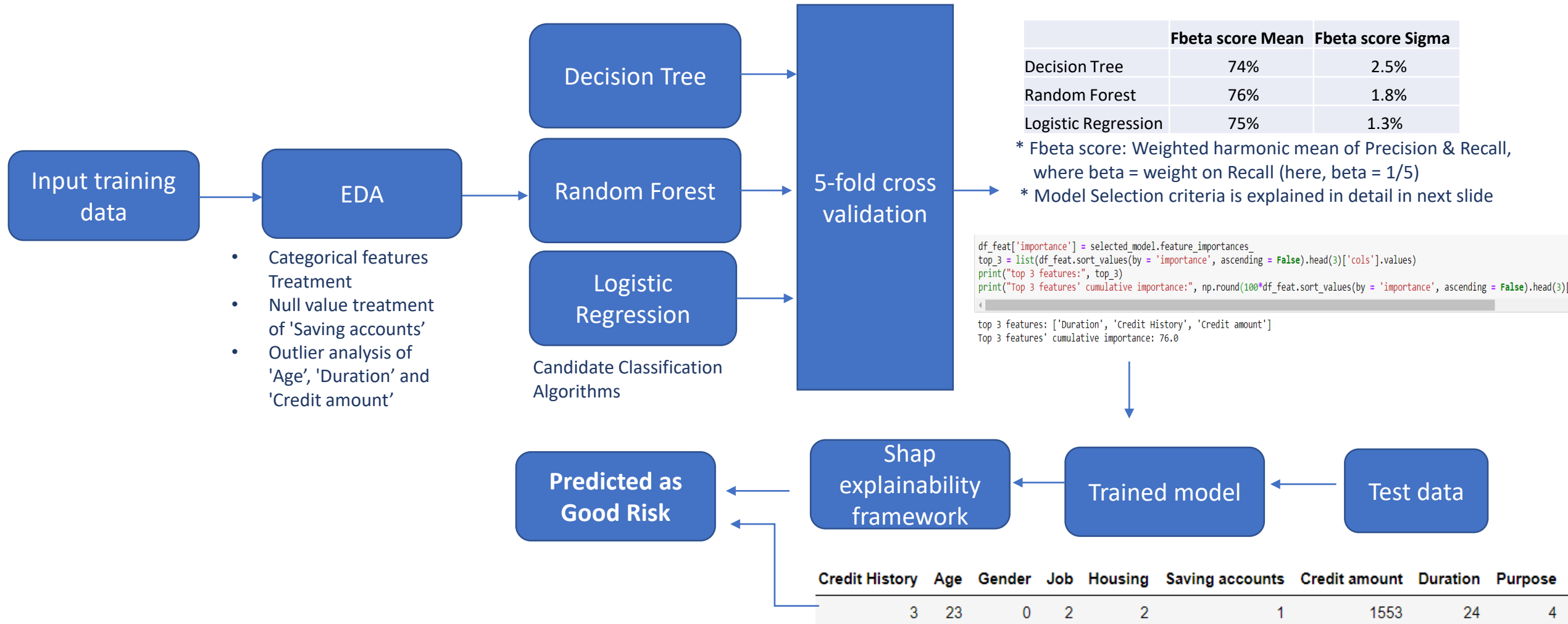
Age Class	Proportion/value counts Bad Risk	Proportion/value counts Good Risk	Bad/Good
0	138	236	58%
1	94	251	37%
2	47	134	35%



As evident in the chart, the proportion of Bad with respect to Good shows a downward trend as the Age increases, hence we conclude that aged people default lesser number of times which in turn says, young people have more tendency to default

- Rationale behind creating such Age bins:
 - Age < 30: Could comprise of Loan seeking, affluent spenders
 - 30 < Age < 45: Generally consists of married people seeking financial security for family, or as an investment vehicle
 - Age > 45: Averse to seeking loans, in general (could have made this to Age 50, but sample size would have gone smaller)

Machine Learning Pipeline



Prediction and Definition of credit worthiness learned from pattern recognition via Machine Learning

```
df_test.loc[710,:]
```

Credit History	4
Age	32
Gender	1
Job	3
Housing	1
Saving accounts	3
Credit amount	629
Duration	18
Purpose	4
Risk	1

Name: 710, dtype: int64

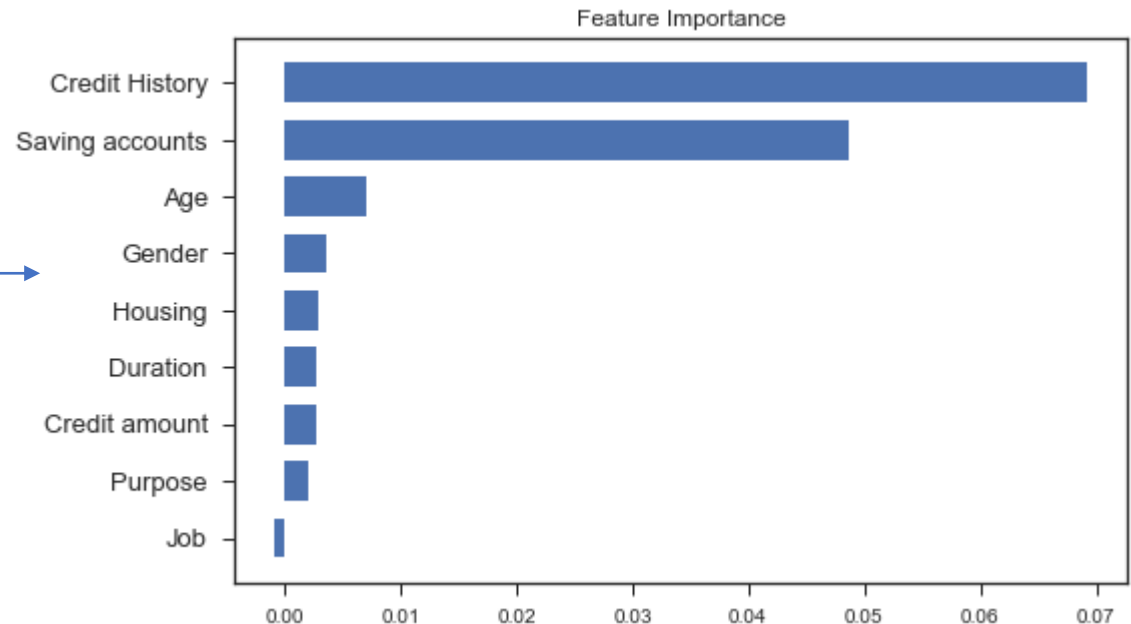
Test row aka profile of new customer applying for loan

Trained model

Shap framework

Global feature importance considers interactions of all historical customer profiles who availed loan wrt their repayment status (classified as Risk vs Good Risk)

- As per selected model, Duration, Credit History, Credit Amount are the key attributes



Local feature importance takes the out of sample/unseen data of new customer applying for loan. Based on learned associations, trained model predicts the probability of risk of this customer (710). Now comes the main part: Why??

- Credit History → Class 4: signifies confidence
- Saving Accounts → Class 3: like credit history (the higher the better)

Model Selection Criteria

- F score is generally used as evaluation metric when there is no preference between Recall and Precision

- $2PR/(P+R)$, where

- P: Precision: focus on reducing False Positives (FN)

- R: Recall: focus on reducing False Negatives (FN)

		Prediction	
		Bad	Good
Actual	Bad	0	5
	Good	1	0
Cost Matrix			

- Relevance of importance of Recall vs Precision in our example:

- As per the given report, **there is a cost on classifying the customer as good when they are bad**, i.e. predicting Good Risk (class 1, positive class) when it is False i.e. False Positives (FP) draw heavy penalty based on our business objective

- Precision = $TP/(TP+FP)$, so our goal is to prioritize Precision which in turn will reduce the FP

- **Had the objective been to give penalty on predicting bad when they are good i.e. based on ML recommendations, we are refraining from lending, we would have lost business.**

- In that case, the focus would be on False Negatives i.e. you are predicting Bad Risk (class 0, negative class) while the customer was in Good Risk category. So, we predicted negative which went wrong implying lesser FN, which in turn would lead to more weight on Recall ($TP/TP+FN$)