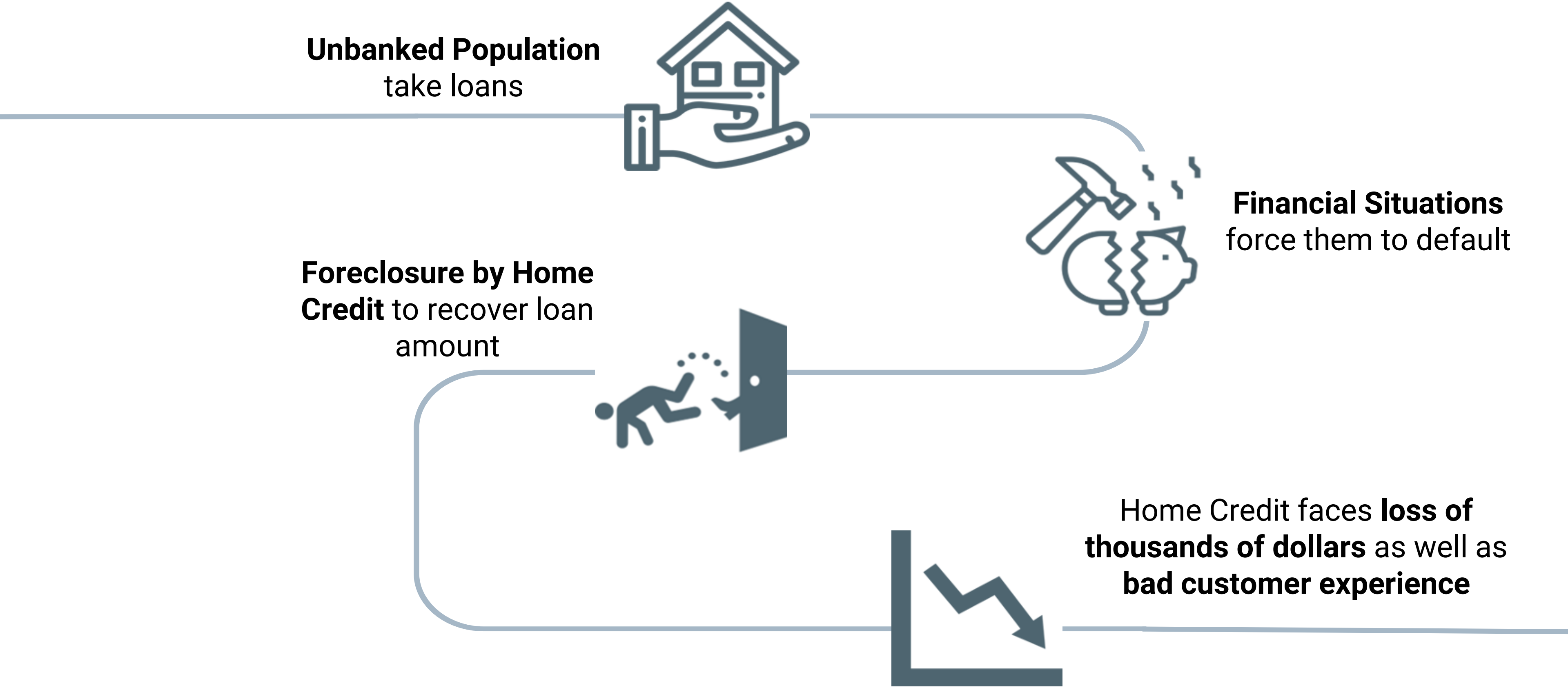
Home Credit Default Risk

Power to know who will default

Manikanta Chinta

Problem


Unbanked Population take loans
Financial Situations force them to default
Foreclosure by Home Credit to recover loan amount
Home Credit faces **loss of thousands of dollars** as well as **bad customer experience**

What Home Credit is doing right now?

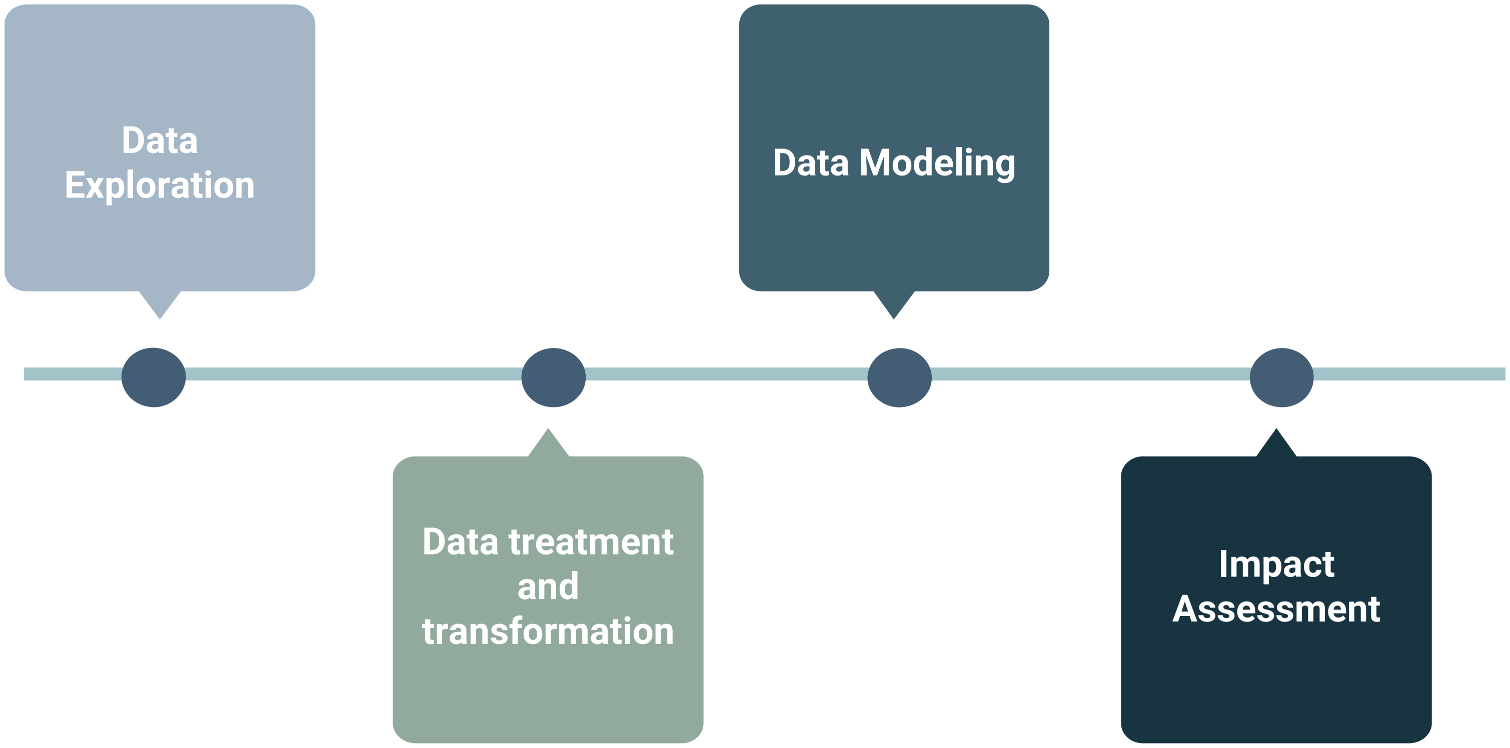
Usually, Financial Institutions predict which customers will default using **logistic regression** which is one of the popular techniques

With the data home credit collects, logistic regression model ranks only **56%** of random defaulters above random non defaulters

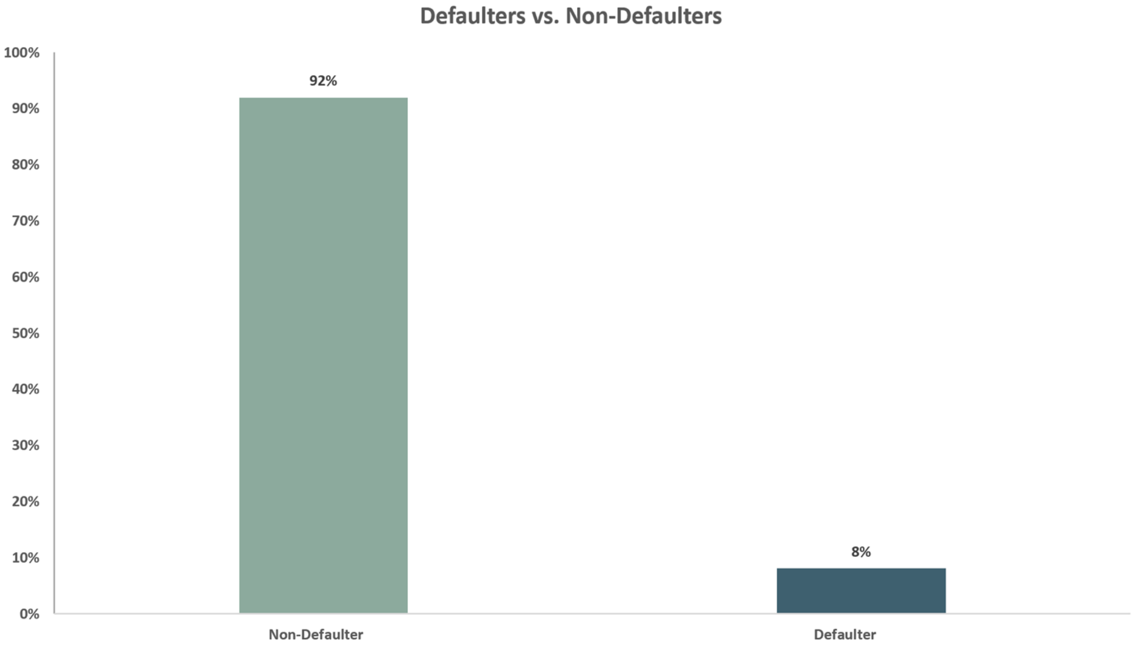
Our Results

Our model can rank **76%** of times a random defaulter above a random non defaulter

So how did we create this model?



Data Exploration

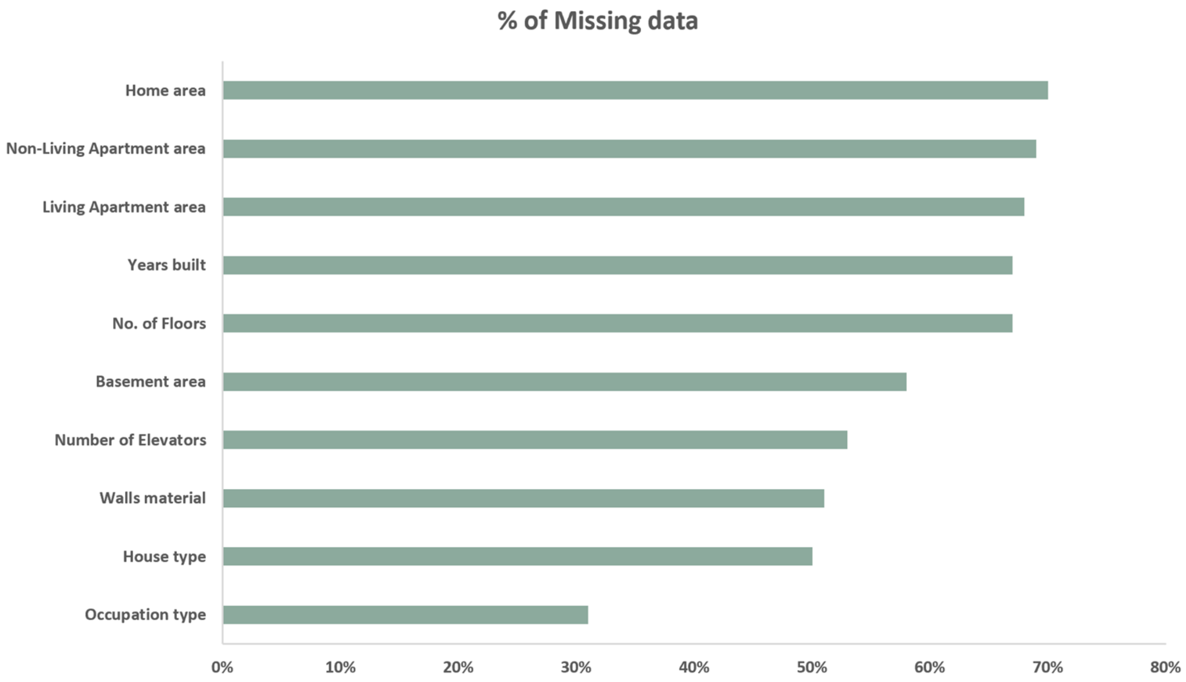


Number of non-defaulters are 10 times defaulters

Missing values of various features

Features are not able to provide any distinction in the target class

Data Exploration

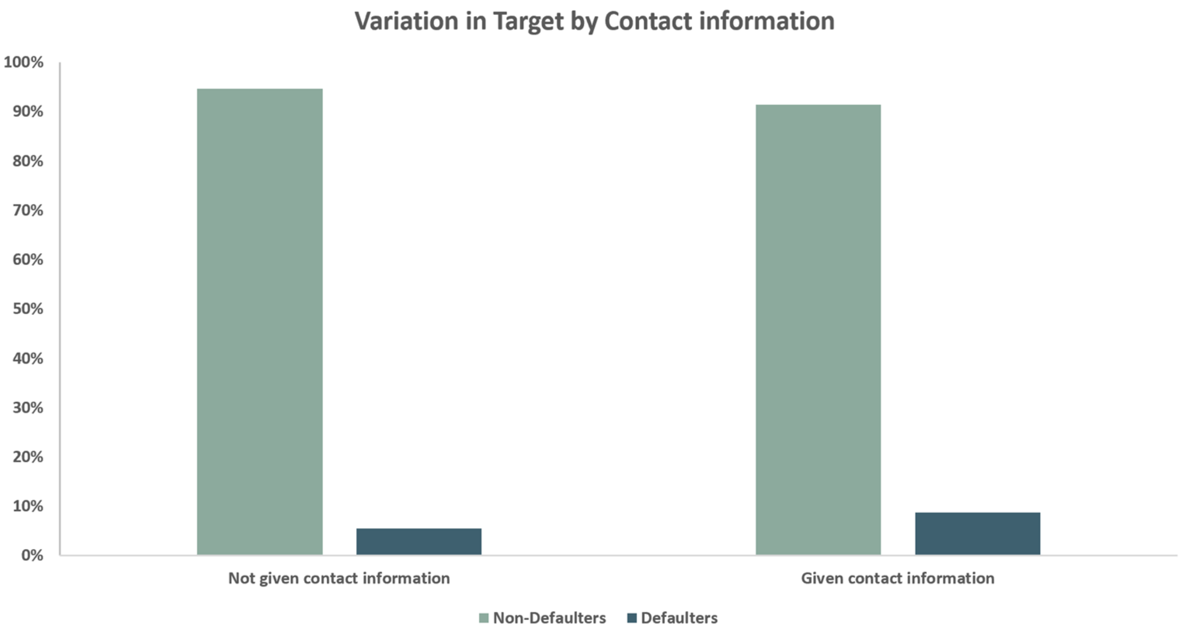


Number of non-defaulters are 10 times defaulters

Missing values of various features

Features are not able to provide any distinction in the target class

Data Exploration



Number of non-defaulters are 10 times defaulters

Missing values of various features

Features are not able to provide any distinction in the target class

Data Transformation

Ancillary data sources
- Previous application data
- 3rd party application data

Treat categorical data
- One Hot Encoding

Deal with Null values
- Remove columns with high number of null values
- Impute others by Median

Normalization
- Standardized data based on z-scores

Feature Engineering

Using domain knowledge to transform some existing variables and Creating new variables which can help the algorithm in performing better. Some of fields are listed below.

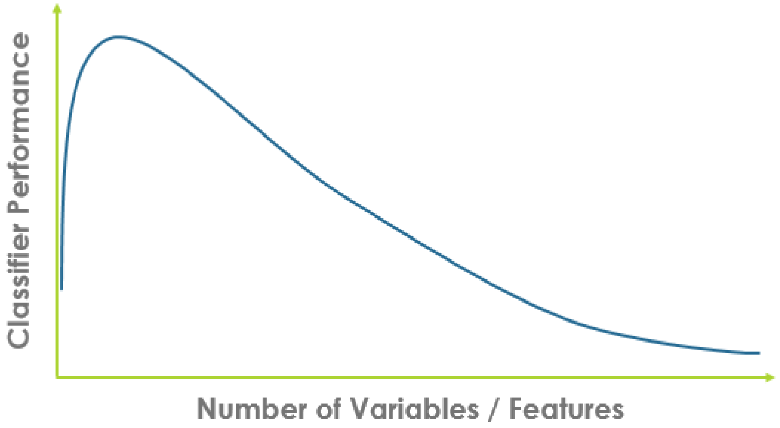
Transformed the below existing columns:

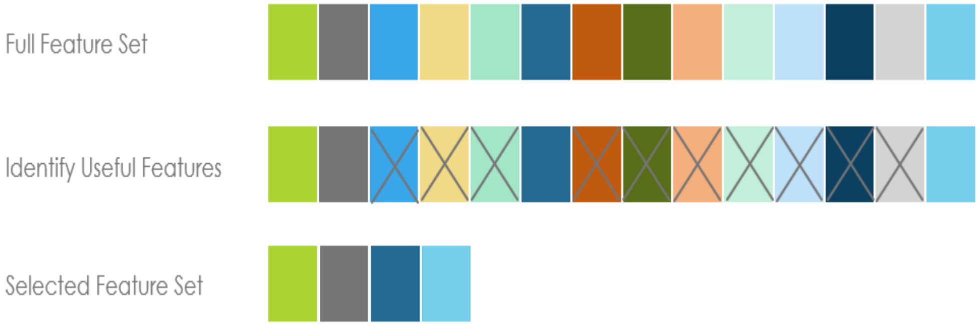
- % of loan amount approved
- No. of previous applications

Created the below new columns:

- Previous Rejects
- Previous applications(yes or no)
- Percent of time employed

Feature Selection





With too many features, models struggle to find a pattern because of noise introduced by features that do not hold any information about the target.

More features leads to increased calculations impacting model performance

Using feature selection algorithm we eliminated all such features and retained only the features that are important.

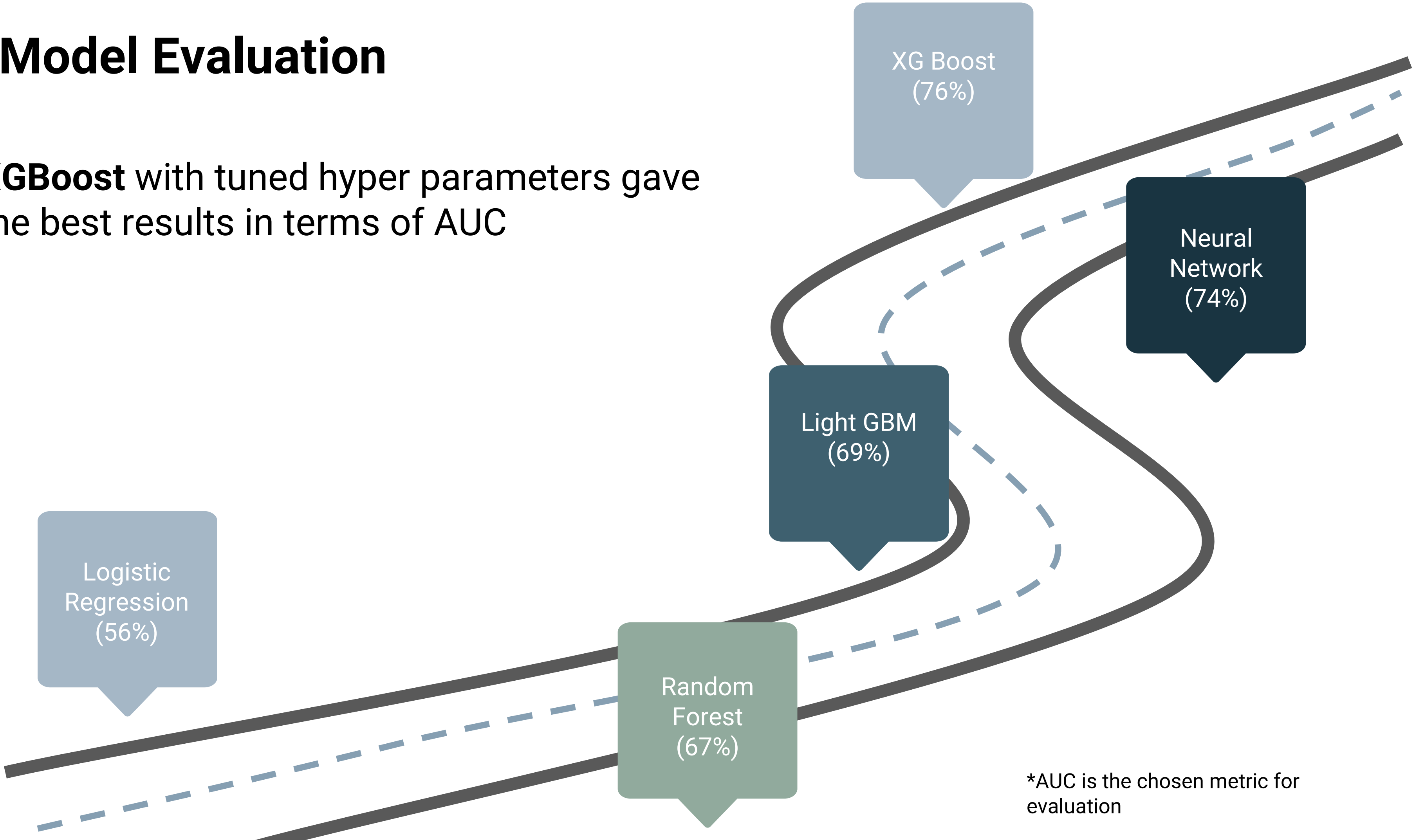
Feature Selection Technique:
XGBOOST feature importance based on information gain.

Final Feature Count : 76

Total Features: 490 features after one hot encoding categorical variables*

Model Evaluation

XGBoost with tuned hyper parameters gave the best results in terms of AUC



*AUC is the chosen metric for evaluation

Impact in terms of $$$

<table>
<tr><th></th><th>AUC</th><th>Cost Savings</th></tr>
<tr><td>Logistic</td><td>56%</td><td rowspan="2">$108k*</td></tr>
<tr><td>XG Boost</td><td>76%</td></tr>
</table>

*Assuming that the bank incurs a loss of $60k per defaulter

Thank YOU !!!!!!