MIS 637 : Data Analytics and Machine Learning

STEVENS INSTITUTE OF TECHNOLOGY 1870

Predicting Loan approval using Decision Tree and Logistic Regression

By:

Eric Arrieta, Swetha Nalanagula, Nirmal Rajan & Lihan Tu

Supervised By: Prof. Mahmoud Daneshmand

Introduction

- One of the key functions of banks and financial institutions is to provide loans to individuals and businesses
- Not all loan applicants, however, are able to fulfill the requirements for loan approval
- Banks and other financial institutions can make better decisions and lower their risk of loan defaults by utilizing machine learning models to predict whether to approve a loan or not
- As a result, the financial institutions may process loans more quickly and be more profitable

STEVENS
INSTITUTE OF TECHNOLOGY
1870

Objectives of the Project

- The main goal of this project is to predict loan application approval via machine learning models

- Two models, Decision Tree and Logistic Regression, will be developed for the purpose of the project

- We are strictly following the steps under the Cross-Industry Standard Process (CRISP-DM) throughout the entire project

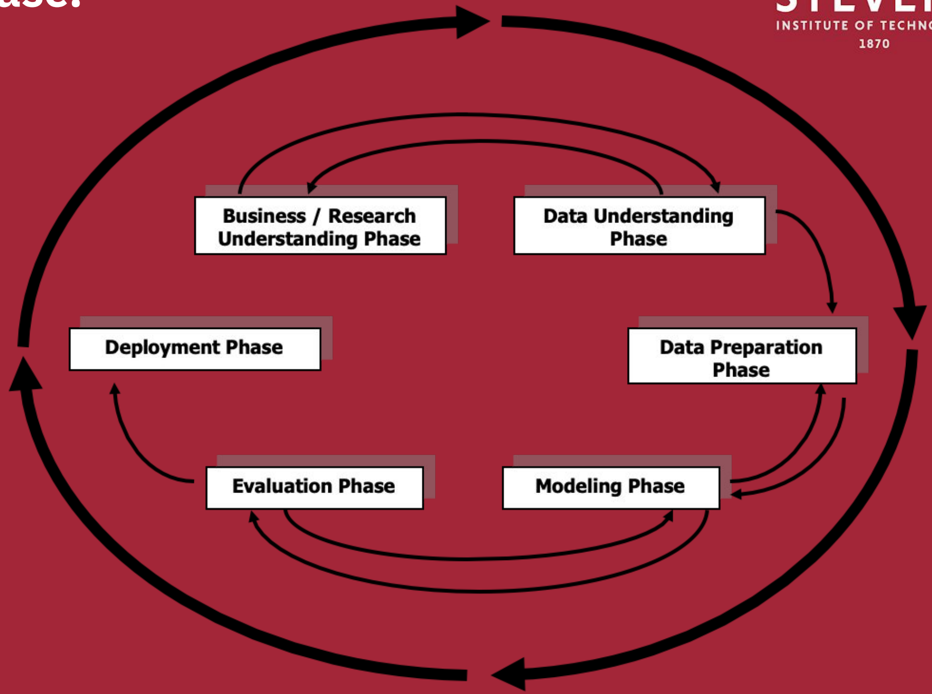
- The software we would use for the project are Python and Jupyter Notebook. Python has many built-in packages, such as pandas, numpy, matplotlib, seaborn and scikit-learn, that help with data analysis and machine learning

- By doing this project, we hope to assist financial institutions in making more precise and informed decisions about loan applications

- The project will also provide us with the chance to learn how various machine learning models perform on classification tasks

Cross-Industry Standard Process (CRISP-DM)

1. **Business Understanding Phase.**
2. **Data Understanding Phase**
3. **Data Preparation Phase**
4. **Modeling Phase**
5. **Evaluation Phase**
6. **Deployment Phase**



STEVENS INSTITUTE OF TECHNOLOGY 1870

Business Understanding

- **Profound Question:**
  Determine if a loan applicant will get their loan approved or not

- **Goal of Project:**
  Building Decision Tree and Logistic Regression Models to predict if a loan application will be approved or not

- **Context of the Question:**
  With machine learning this could help the bank(s) automate the loan approval process. This will help predict the likelihood of a new client getting their loan approved based on various factors such as credit history, income, education and other pertinent data points

- **Key Stakeholders:**
  Bank executives, loan managers at bank, database managers at bank, loan applicants, and all other relevant parties

Data Understanding

- **Data Source:**
  The data set was obtained from www.kaggle.com

- **Dataset Details:**
  The dataset contains 614 loan applications. The dataset has 13 attributes. 12 are independent attributes and 1 is the target attribute. The target attribute is the one that classifies the applicants if they got approved or not for the loan, Loan_Status.

Dataset Sample below:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0	NaN	360	1	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508	128	360	1	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0	66	360	1	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358	120	360	1	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0	141	360	1	Urban	Y
5	LP001011	Male	Yes	2	Graduate	Yes	5417	4196	267	360	1	Urban	Y
6	LP001013	Male	Yes	0	Not Graduate	No	2333	1516	95	360	1	Urban	Y
7	LP001014	Male	Yes	3+	Graduate	No	3036	2504	158	360	0	Semiurban	N
8	LP001018	Male	Yes	2	Graduate	No	4006	1526	168	360	1	Urban	Y
9	LP001020	Male	Yes	1	Graduate	No	12841	10968	349	360	1	Semiurban	N

Data Understanding Cont'd

Dataset attributes description:

Loan_ID : Unique Loan ID

Gender : Male/ Female

Married : Applicant married (Y/N)

Dependents : Number of dependents

Education : Applicant Education (Graduate/ Under Graduate)

Self_Employed : Self employed (Y/N)

ApplicantIncome : Applicant income

CoapplicantIncome : Coapplicant income

LoanAmount : Loan amount in thousands

Loan_Amount_Term : Term of loan in months

Credit_History : 1 - has all debts paid, 0 - not paid

Property_Area : Urban/ Semi Urban/ Rural

Loan_Status : (Target) Loan approved (Y/N)

STEVENS
INSTITUTE OF TECHNOLOGY
1870

Data Understanding - Data Quality

Figure below shows the number of **null values** for each variable

```
# Checking for null values in the columns
loan_df.isnull().sum()
```

```
Loan_ID               0
Gender               13
Married               3
Dependents           15
Education             0
Self_Employed        32
ApplicantIncome       0
CoapplicantIncome     0
LoanAmount           22
Loan_Amount_Term     14
Credit_History       50
Property_Area         0
Loan_Status           0
dtype: int64
```

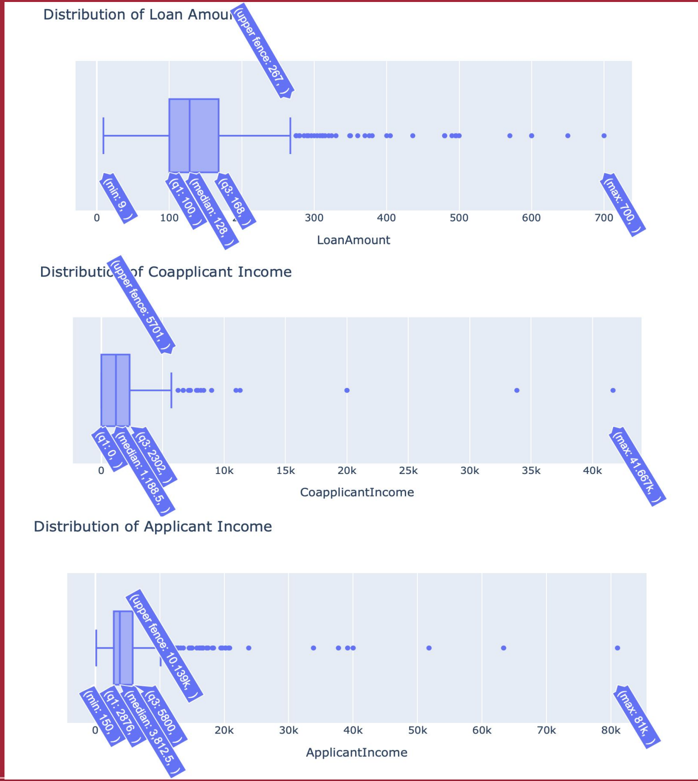
Data Understanding - Data Quality Cont'd

Figure below shows the number of **outliers** for each **numerical** variable

```
Number of outliers under 'LoanAmount': 39
Max Outlier Value: 700.0
Min Outlier Value: 275.0
```

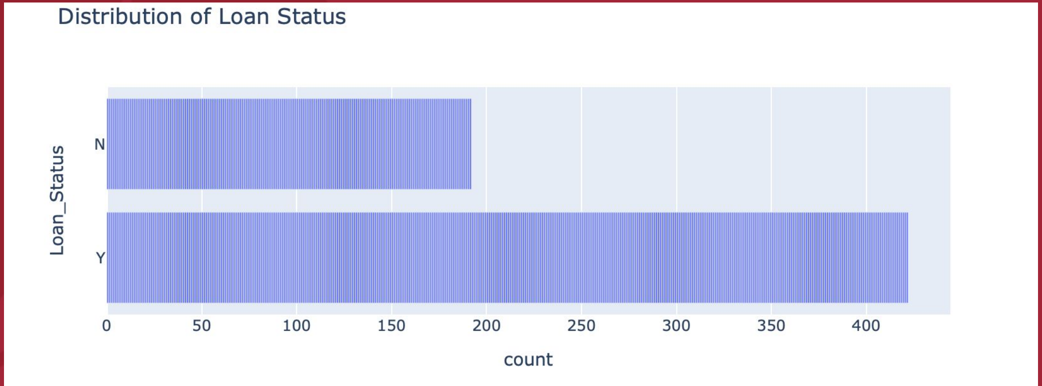
```
Number of outliers under 'CoapplicantIncome': 18
Max Outlier Value: 41667.0
Min Outlier Value: 6250.0
```

```
Number of outliers under 'ApplicantIncome: 50
Max Outlier Value: 81000
Min Outlier Value: 10408
```



Data Understanding - Data Quality Cont'd

Further exploration of data shows that there is **NO** miscategorized data




```
0     0.575960
1     0.170284
2     0.168614
3+    0.085142
Name: Dependents, dtype: float64
```

```
360.0    0.853333
180.0    0.073333
480.0    0.025000
300.0    0.021667
240.0    0.006667
84.0     0.006667
120.0    0.005000
60.0     0.003333
36.0     0.003333
12.0     0.001667
Name: Loan_Amount_Term, dtype: float64
```

```
Yes    0.651391
No     0.348609
Name: Married, dtype: float64
```

```
No     0.859107
Yes    0.140893
Name: Self_Employed, dtype: float64
```

```
1.0    0.842199
0.0    0.157801
Name: Credit_History, dtype: float64
```

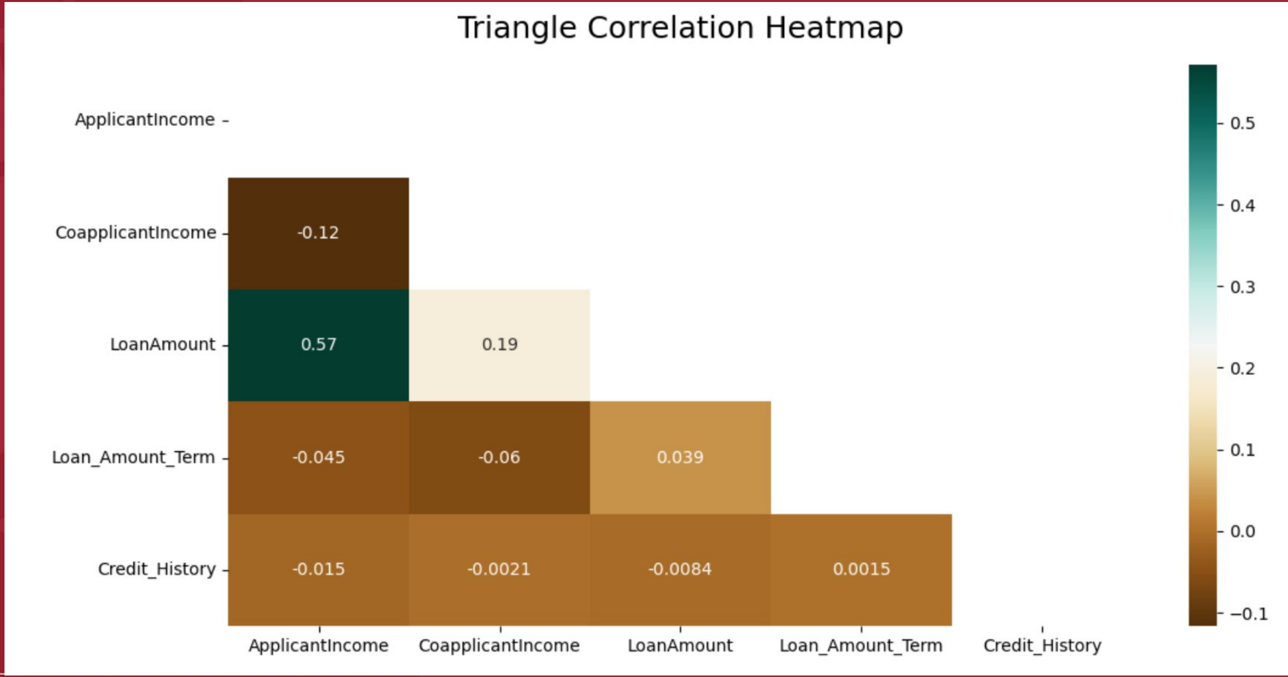
```
Male      0.813644
Female    0.186356
Name: Gender, dtype: float64
```

```
Graduate        0.781759
Not Graduate    0.218241
Name: Education, dtype: float64
```

```
Semiurban    0.379479
Urban        0.328990
Rural        0.291531
Name: Property_Area, dtype: float64
```

Data Understanding - Exploratory Data Analysis

Heatmap with correlation coefficients for all **numerical independent** variables are displayed below. There is **NO** correlation among numerical variables per correlation coefficients. (Statistically significant correlation should have coefficient of 0.7 and above)



Data Understanding - Exploratory Data Analysis Cont'd

Cross-Tabulation analysis is performed to explore relationships between **categorical independent** variables and target variable, Loan_Status. We included Loan_Amount_Term and Credit_History here suspecting that they might have categorical trait.

Loan_Status Gender	N	Y
Female	0.330357	0.669643
Male	0.306748	0.693252

Loan_Status Property_Area	N	Y
Rural	0.385475	0.614525
Semiurban	0.231760	0.768240
Urban	0.341584	0.658416

Loan_Status Married	N	Y
No	0.370892	0.629108
Yes	0.283920	0.716080

Loan_Status Loan_Amount_Term	N	Y
12.0	0.000000	1.000000
36.0	1.000000	0.000000
60.0	0.000000	1.000000
84.0	0.250000	0.750000
120.0	0.000000	1.000000
180.0	0.340909	0.659091
240.0	0.250000	0.750000
300.0	0.384615	0.615385
360.0	0.298828	0.701172
480.0	0.600000	0.400000

Loan_Status Education	N	Y
Graduate	0.291667	0.708333
Not Graduate	0.388060	0.611940

Loan_Status Dependents	N	Y
0	0.310145	0.689855
1	0.352941	0.647059
2	0.247525	0.752475
3+	0.352941	0.647059

Loan_Status Self_Employed	N	Y
No	0.314000	0.686000
Yes	0.317073	0.682927

Loan_Status Credit_History	N	Y
0.0	0.921348	0.078652
1.0	0.204211	0.795789

STEVENS
INSTITUTE OF TECHNOLOGY
1870

Data Understanding - Exploratory Data Analysis Cont'd

- There is no statistically correlations between all numerical variables
- Analysis shows that Gender, Self_Employed, and Dependents do not have much effects on the target variable, Loan_Status
- The only category under variable Property_Area that seems to have some impact on Loan_Status is Semiurban
- Credit_History shows the strongest impact on Loan_Status per cross-tabulation analysis

Data Preparation

- **Dropped Variables:** Through data exploration we came to the conclusion that we can drop the columns Loan_ID, Gender, Self_Employed, Dependents. Loan_ID is an identifier of each loan applicant and serves no purpose. Gender, Self_Employed, and Dependents are shown to have very little to no impact on the target variable, Loan_Status.
- **Treatment of Property_Area Variable:** Property_Area was converted into a binary column with “Semiurban” as positive and “Rural”/”Urban” as negative.
- **Treatment of Missing Values:** We dropped the rows that had NULL values. We have in fact tried multiple ways of treating null values: replacing null values with median, mode, or random values. After multiple loops of the CRISP process, we discovered dropping missing values provided the most accurate testing result.
- **Treatment of Outliers:** We left the outliers as they are in place. We have in fact tried multiple ways of treating outliers: replacing outliers with median or mode or capping outliers with the IQR bounds. After multiple loops of the CRISP process, we discovered that all treatments of outliers reduce the accuracy of model prediction based on testing result and decided to leave the outliers as they are in place.

STEVENS
INSTITUTE OF TECHNOLOGY
1870

Data Preparation Cont'd

After applying all treatments of data from the Data Preparation Phase, we have 527 records available for modeling.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 527 entries, 1 to 613
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Married            527 non-null    object
 1   Education          527 non-null    object
 2   ApplicantIncome    527 non-null    int64
 3   CoapplicantIncome  527 non-null    float64
 4   LoanAmount         527 non-null    float64
 5   Loan_Amount_Term   527 non-null    float64
 6   Credit_History     527 non-null    float64
 7   Semiurban          527 non-null    int64
dtypes: float64(4), int64(2), object(2)
memory usage: 37.1+ KB
```

We split the 527 records available for modeling randomly into 60-40:

- **Training Set: 60%** of all records available for modeling
- **Testing Set: 40%** of all records available for modeling

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=42)
```

Modeling - Decision Tree

The target variable is categorical(yes/no) and it is not continuous. Therefore we will use two of the Machine Learning algorithms for our case.

1. **Decision Tree**

   It is a collection of decision nodes, connected by branches, extending downward from the root node until terminating in leaf nodes. It creates a tree like structure which is easily readable which makes it easier for interpretation and understanding of the model. Since the loan approval process requires a lot of decisions to be made involving complex and diverse data type the decision tree model can handle these requirements and can also handle missing values and outliers that may be found in the data.

   Decision Trees can handle both Categorical and Numerical values which is suitable for the loan approval process.

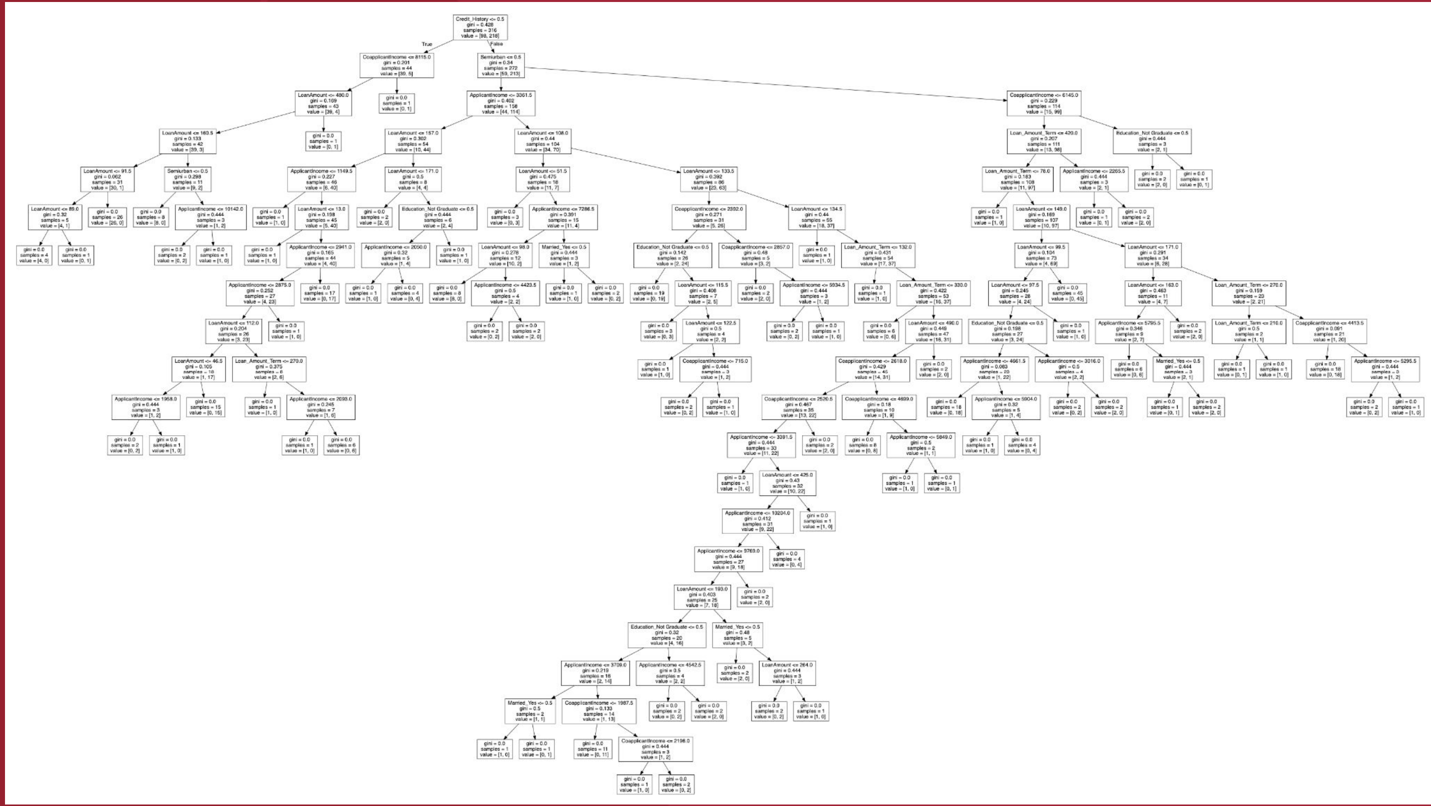
```
# Create a DecisionTreeClassifier instance
tree_clf = DecisionTreeClassifier(random_state=42)

# Train the decision tree on the training data
tree_clf.fit(X_train, y_train)

# Make predictions on the training data
y_pred = tree_clf.predict(X_test)

# Accuracy
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, pos_label = 'Y')
recall = recall_score (y_test, y_pred, pos_label = 'Y')
f1 = f1_score (y_test, y_pred,pos_label = 'Y')
```

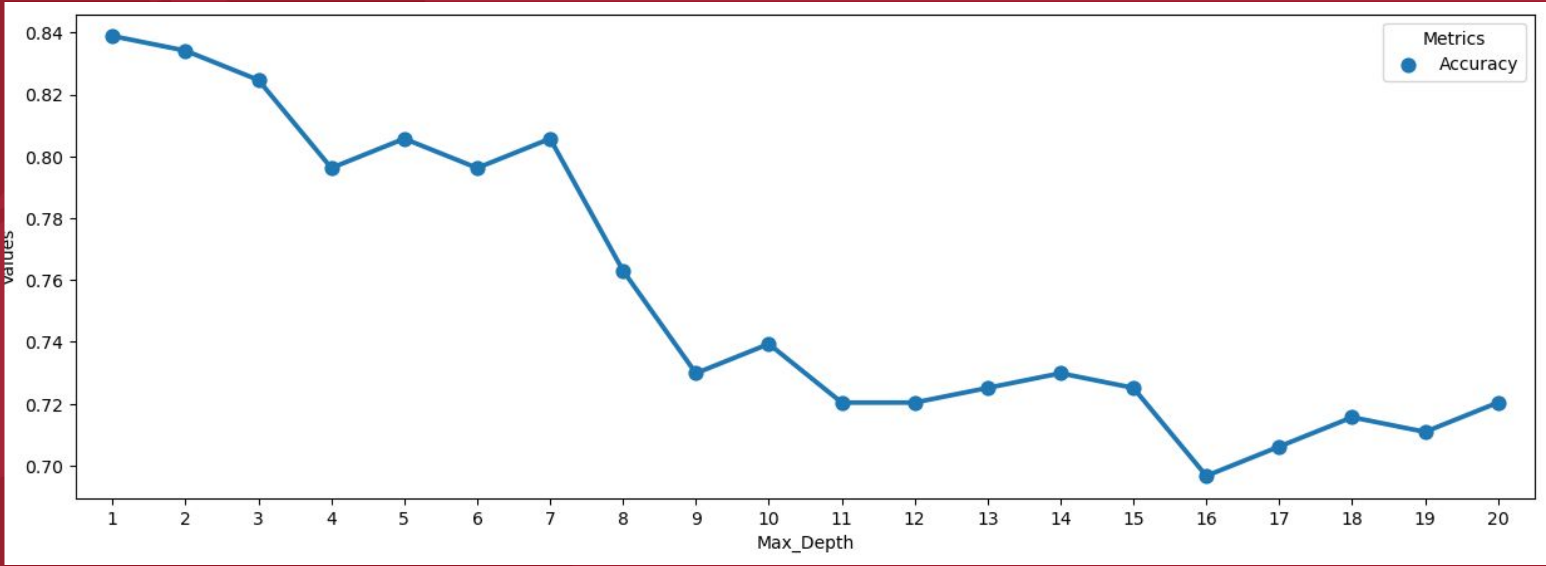
Modeling - Decision Tree Cont'd



The figure on the left shows us the full decision tree in its total depth of 20 and you can see that it is **overfitting** the model. Since this is the case **we are going to prune the data** to see how it fits.

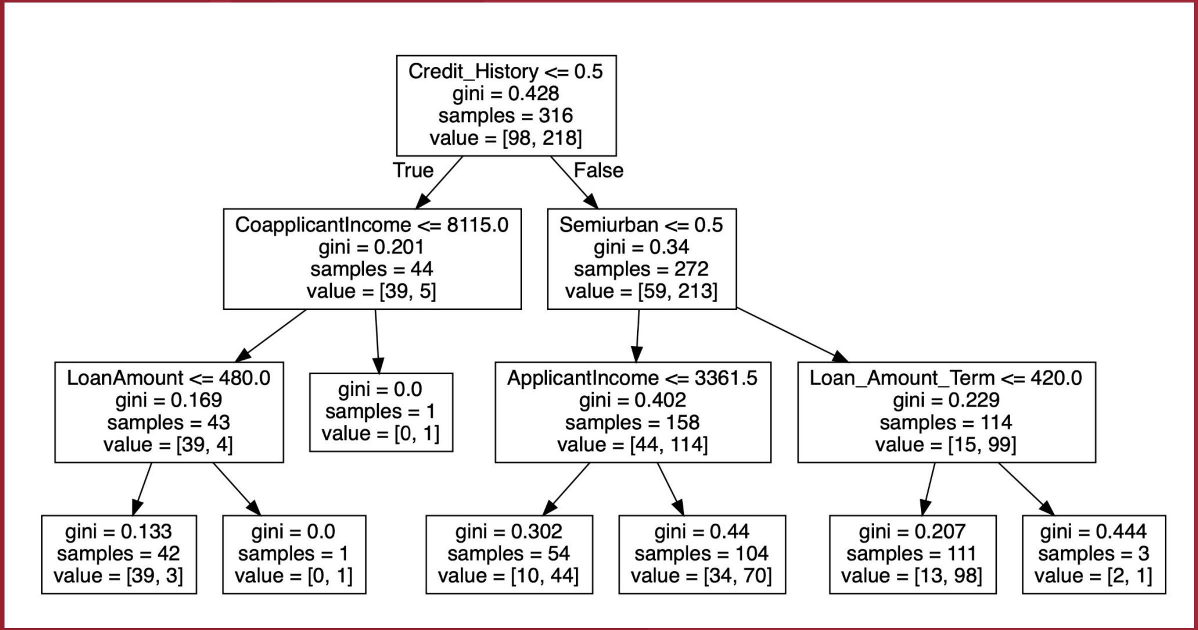
Modeling - Decision Tree Cont’d

The figure below shows us the accuracy of the model while pruning for depths 1 - 20.



Modeling - Decision Tree Cont'd

The figures below show **the final depth of 3** of the decision tree which we decided on based on the results obtained from pruning and the accuracy results that were obtained at depth 3.




```
Accuracy: 0.8246445497630331
Precision: 0.8114285714285714
Recall: 0.9726027397260274
f1: 0.8847352024922118
model_score: 0.8069620253164557
```

STEVENS
INSTITUTE OF TECHNOLOGY
1870

Modeling - Logistic Regression

STEVENS
INSTITUTE OF TECHNOLOGY
1870

2. Logistic Regression

It is statistical model used for classification and predictive analysis. It works well with independent variables and since the outcome is a probability the output is going to be binary. The loan approval process can get data intensive and the model can be scaled accordingly and does not require tuning. A model can be trained easily using logistic regression.

Logistic Regression can handle both Categorical and Numerical values which is suitable for the loan approval process.

```
# Logistic Regression model
model = LogisticRegression()

# Fit it into the training data
model.fit(X_train, y_train)

# Predictions on the data
y_pred = model.predict(X_test)

# Accuracy
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, pos_label = 'Y')
recall = recall_score (y_test, y_pred, pos_label = 'Y')
f1 = f1_score (y_test, y_pred,pos_label = 'Y')
```

Modeling - Logistic Regression Cont'd

Three Elements of the Logistic Regression model are shown below:

- Independent Variables used
- Model Intercept
- Model coefficients

```
Int64Index: 527 entries, 1 to 613
Data columns (total 8 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   ApplicantIncome         527 non-null    int64
 1   CoapplicantIncome       527 non-null    float64
 2   LoanAmount              527 non-null    float64
 3   Loan_Amount_Term        527 non-null    float64
 4   Credit_History          527 non-null    float64
 5   Semiurban               527 non-null    int64
 6   Married_Yes             527 non-null    uint8
 7   Education_Not Graduate  527 non-null    uint8
dtypes: float64(4), int64(2), uint8(2)
memory usage: 29.8 KB
Model intercept is [0.09925374]
Model coefficients are [[ 5.32048395e-06 -5.15330645e-05 -1.11870002e-03 -4.69686502e-03
   2.72599431e+00  8.45332640e-01  4.61999111e-02 -6.15479752e-01]]
```

Evaluation

Logistic regression Model Scores

```
Accuracy: 0.8436018957345972
Precision: 0.8228571428571428
Recall: 0.9863013698630136
f1: 0.8971962616822429
model_score: 0.8006329113924051
```

Decision Tree Model Scores

```
Accuracy: 0.8246445497630331
Precision: 0.8114285714285714
Recall: 0.9726027397260274
f1: 0.8847352024922118
model_score: 0.8069620253164557
```

STEVENS
INSTITUTE OF TECHNOLOGY
1870

From evaluating both models' performance, we should be able to make good predictions on the loan applications using the applicants' information.

The **Accuracy score** is a model's prediction success rate on the **testing data set**. The **model_score** is a model's training score on the **training data set**. For both models, the accuracy score and model_score are close and therefore no sign of overfitting.

The **F1 score** ranges from 0 - 1. The higher the score the better the performance. Here both models have the scores closer to 1 which means it has great performance.

The **Recall score** shows the model's tendency to capture all the positive cases. A high recall score shows that it has a smaller chance of missing the positive cases.

The **Accuracy scores** for both models, 84.36% for Logistic Regression and 82.46% for Decision Tree, indicate high successful rate of predicting whether to approve a loan applicant or not.

Deployment

STEVENS INSTITUTE OF TECHNOLOGY 1870

This is the final phase of the CRISP-DM methodology. After the evaluating the results from both the models we will integrate the model for loan approval into the existing workflow of the bank(s).

The CRISP methodology that we used to run the Decision Tree and Logistic Regression models on the loan approval process for banks resulted in a higher percentage rate of accuracy. Logistic Regression had an accuracy of 84% and Decision Tree had an accuracy of 82% with no overfitting of the models.

Hence we can suggest that these models can be effectively used by banks to determine if a loan applicant's application will be approved or not.

References

- Methodology: Lecture Notes
- Dataset: www.kaggle.com
- Picture of Bank Counter: https://www.cnbc.com/select/best-personal-loans-from-big-banks/
- Picture of Loan Application: https://www.sbdc.uh.edu/sbdc/The_Secrets_to_Loan_Approval_-_UH_SBDC.asp

THANK YOU

Stevens Institute of Technology
1 Castle Point Terrace, Hoboken, NJ 07030