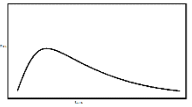
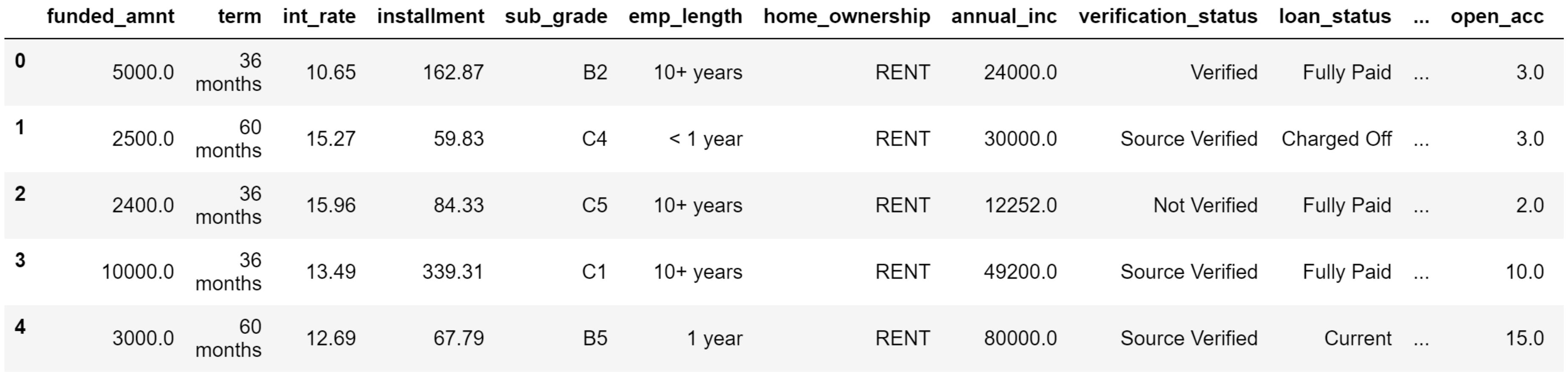


# Applying Machine Learning to Lending Club's Loan Data

# A Lending Tree dataset on Kaggle contained many possibilities for prediction



| | funded_amnt | term | int_rate | installment | sub_grade | emp_length | home_ownership | annual_inc | verification_status | loan_status | ... | open_acc |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 5000.0 | 36 months | 10.65 | 162.87 | B2 | 10+ years | RENT | 24000.0 | Verified | Fully Paid | ... | 3.0 |
| 1 | 2500.0 | 60 months | 15.27 | 59.83 | C4 | < 1 year | RENT | 30000.0 | Source Verified | Charged Off | ... | 3.0 |
| 2 | 2400.0 | 36 months | 15.96 | 84.33 | C5 | 10+ years | RENT | 12252.0 | Not Verified | Fully Paid | ... | 2.0 |
| 3 | 10000.0 | 36 months | 13.49 | 339.31 | C1 | 10+ years | RENT | 49200.0 | Source Verified | Fully Paid | ... | 10.0 |
| 4 | 3000.0 | 60 months | 12.69 | 67.79 | B5 | 1 year | RENT | 80000.0 | Source Verified | Current | ... | 15.0 |

- The full dataset contained 887,379 rows and 74 columns
- Each row represented a loan that was approved and paid out
- Variables contained details on the loan terms, details on the borrower, and details on the current status of the loan
- Loans were issued from Dec 2011 through Jan 2015
- Loan statuses include:
  - Fully Paid, Charged Off, Current, and 4 separate levels of Late (Grace period, 16-30 days, 31-120 days, default)

# We want to predict whether Current and Late loans are likely to end up being Charged Off

| Loan Status | % of Total |
|---|---|
| Charged Off | 5.1% |
| Fully Paid | 23.4% |
| Current | 67.8% |
| Late | 2.4% |
| Other | 1.3% |

| | Late Loans | Current Loans |
|---|---|---|
| Count | 21,420 | 601,779 |
| Value | $335,976,325 | $9,171,214,950 |

- These represent the loans that have concluded with either being **Fully Paid** or **Charged Off**
- Charged Off loans are representative of the **"Bad Debt expense"** incurred by lending organizations
- We can extrapolate the results of the combined '**Charged Off – Fully Paid**' Model on **Current** and **Late** Loans.
- This will allow us to expand our horizons, and estimate the number of outstanding loans that are **likely to not be paid back**

Method Used : Classification

# We can also predict how much we are likely to lose from each Current and Late loan being Charged Off

- Looking within the Charged Off loans we can evaluate what **amount** is **actually paid** before being Charged Off

$$loss\ fraction = \frac{funded\ amnt\ - total\ rec'd\ principle}{funded\ amount}$$

- A **Regression Model** can identify the expected **percentage** of a loan that might not be paid based on the loan & borrower features

$$Features \xRightarrow[regression\ tree]{} E[loss\ fraction|features]$$

- Combining this with the number of likely Charged Off loans gives the **expected value** in dollars for the pool of **Late** loans, something **our Finance Team** is very interested in

$$n\left(Late\ loans \xrightarrow[classified\ as]{} Charge\ Off\right)$$

$$\times\ E[loss\ fraction|Charged\ Off]$$

$$= Accounting\ Writedown$$

Method Used : Regression

# The dataset contains different data types with varied distributions and correlations

### Nominal Variables

- Policy code
- Purpose
- Payment plan
- Loan grade
- State
- Zip code
- Home ownership
- Employment title
- ……

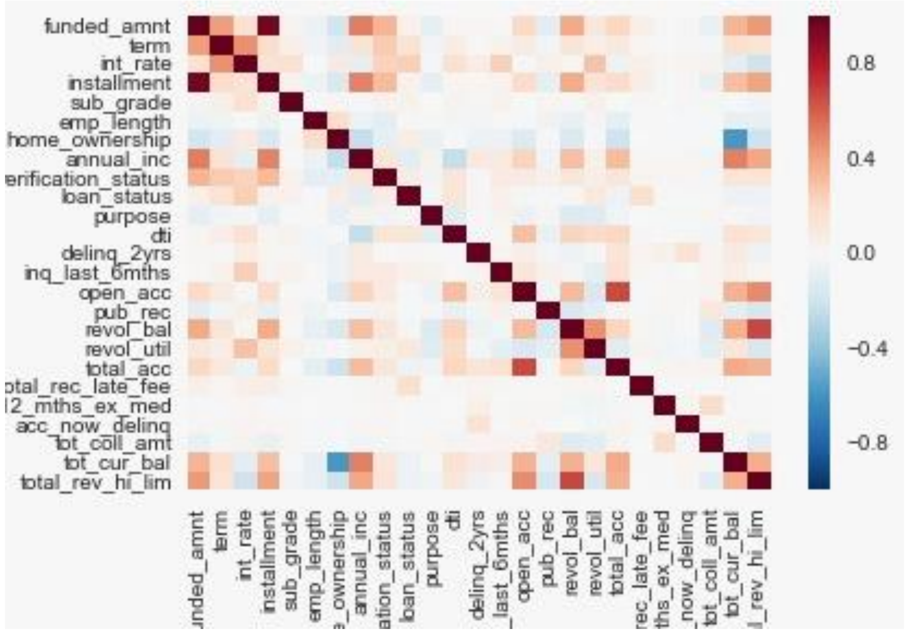
### Numeric Variables

- Annual income
- Employment length
- Recovery fee
- Total loan amount
- Interest rate
- Last total payment
- Total Current balance of all accounts
- ……

### Others

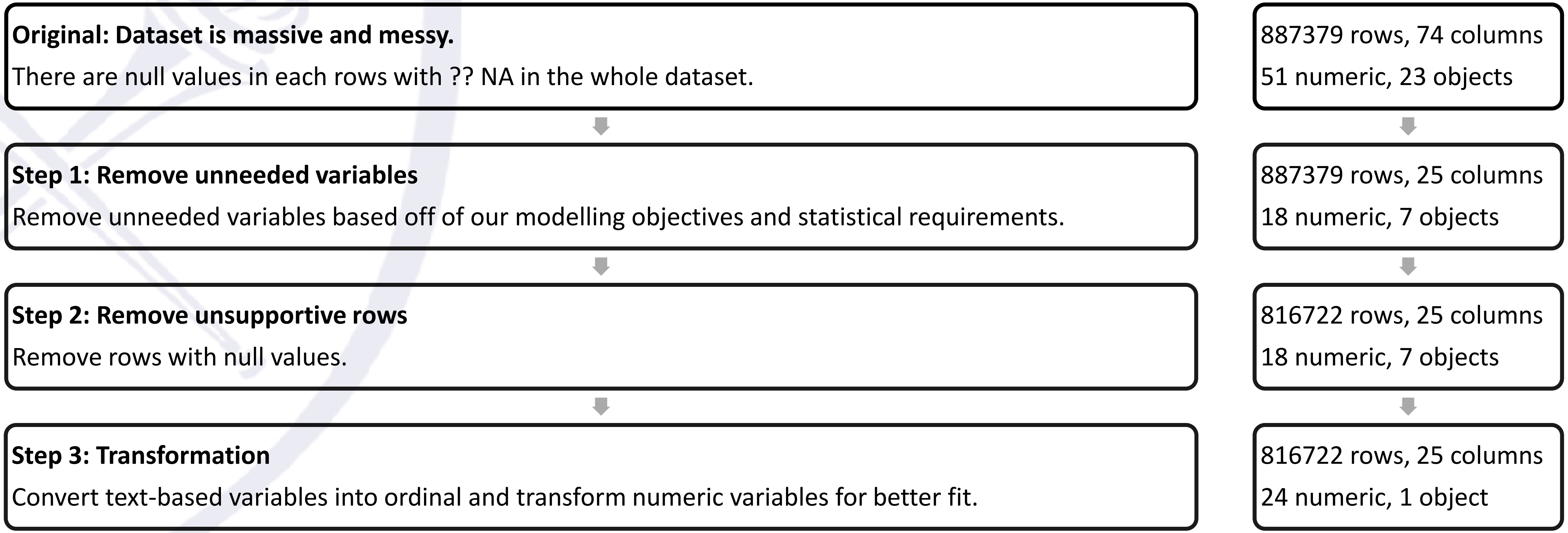
- Verified income (Binary)
- Next scheduled payment date (date)
- ……

### Correlation Map



Phase I - Exploration

# We removed multiple inputs and null values, and executed transformations



**Data Cleaning Process**

**Original: Dataset is massive and messy.**

There are null values in each rows with ?? NA in the whole dataset.

**Step 1: Remove unneeded variables**

Remove unneeded variables based off of our modelling objectives and statistical requirements.

**Step 2: Remove unsupportive rows**

Remove rows with null values.

**Step 3: Transformation**

Convert text-based variables into ordinal and transform numeric variables for better fit.

**Dataset Information**

887379 rows, 74 columns
51 numeric, 23 objects

887379 rows, 25 columns
18 numeric, 7 objects

816722 rows, 25 columns
18 numeric, 7 objects

816722 rows, 25 columns
24 numeric, 1 object

**We initially chose 24 variables to classify loan status**

Phase I - Exploration

# Models were built using Decision Tree, KNN, and Logistic Regression techniques

- These are the 3 common techniques for binary classification

- Cover 3 fundamental data relationships:
  - Rule-based
  - Distance-based
  - Mathematical

- Utilize both ordinal and numeric inputs (or purely nominal inputs given numeric values)

*All three run using SciKit-Learn functions and parameters:*

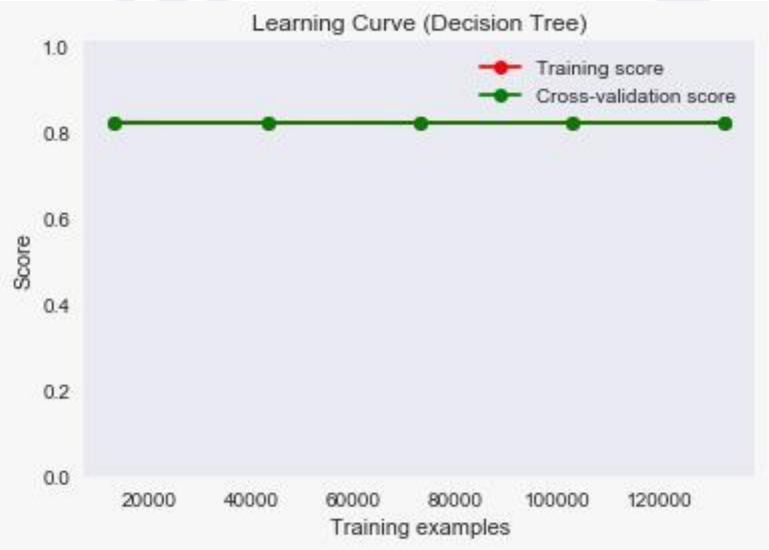
- *DecisionTreeClassifier*
  - *max depth*
  - *min samples per split*
  - *min samples per leaf*
  - *criterion (gini or entropy)*
- *KNeighborsClassifier*
  - *K*
  - *weights*
- *LogisticRegression*
  - *C*
  - *penalty*

Phase II - Classification

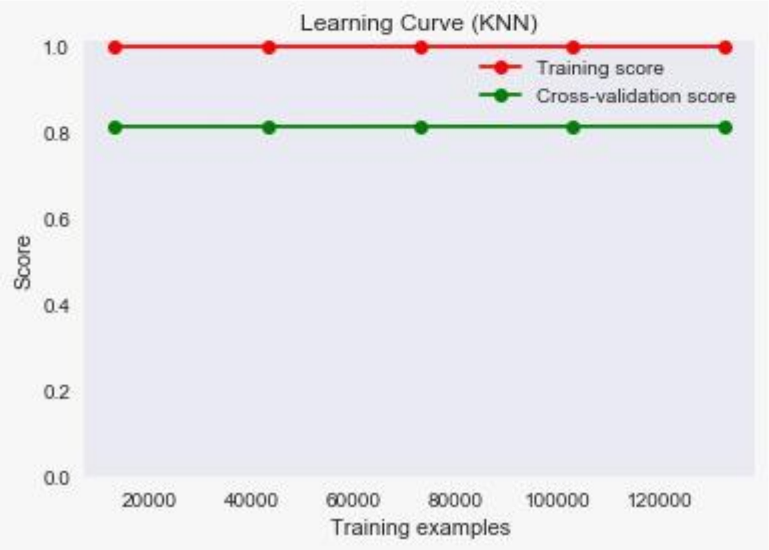
# It was necessary to scale down the dataset for parameter optimization

- A smaller dataset was used for optimizing parameters for each method, due to being computationally intensive
- The dataset was scaled back to its original size for model training, testing and prediction, using Amazon Web Services – EC2

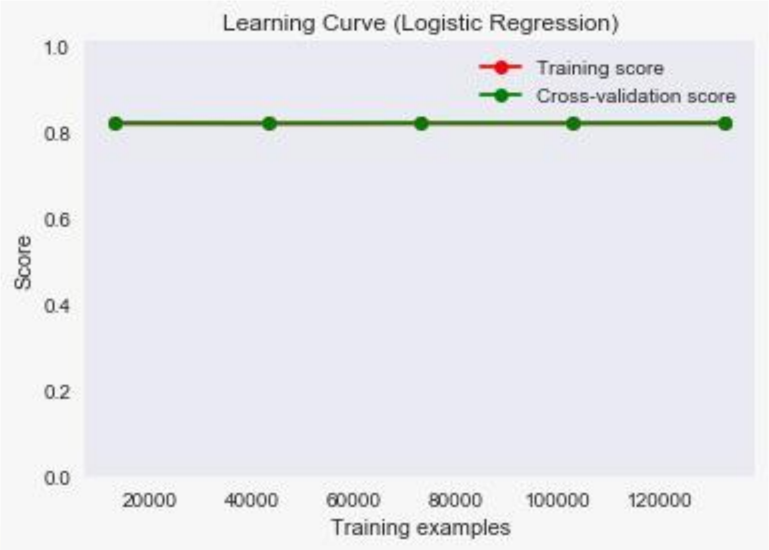
**Decision Tree**



**KNN**



**Logistic Regression**



Phase II - Classification

# Models were then optimized for accuracy by Grid Search and Cross Validation (10 Fold)

- Decision Tree
  - Max depth from 1 to 12: chose **3**
  - Min samples per split from 2 to 10: chose **2**
  - Min samples per leaf from 1 to 10: chose **7**
  - gini or entropy: chose **gini**

- KNN
  - “k” for odd values from 1 to 29: chose **k = 25**
  - Weights as uniform or by distance: chose **by distance**

- Logistic Regression
  - “C” for powers of 10 from 0.00001 to 10,000,000: chose **C = 100**
  - Evaluated L1 and L2 penalties: chose **L2**

Phase II - Classification

# Models were evaluated by ROC/AUC and confusion matrices – Logistic Regression selected

## Model Results

### Decision Tree

| Class | Precision | Recall | F1-score |
|---|---|---|---|
| Fully paid | 0.83 | 1.00 | 0.90 |
| Charged off | 0.75 | 0.06 | 0.11 |
| **Accuracy** | **0.82** | **F-measure** | **0.51** |

### KNN

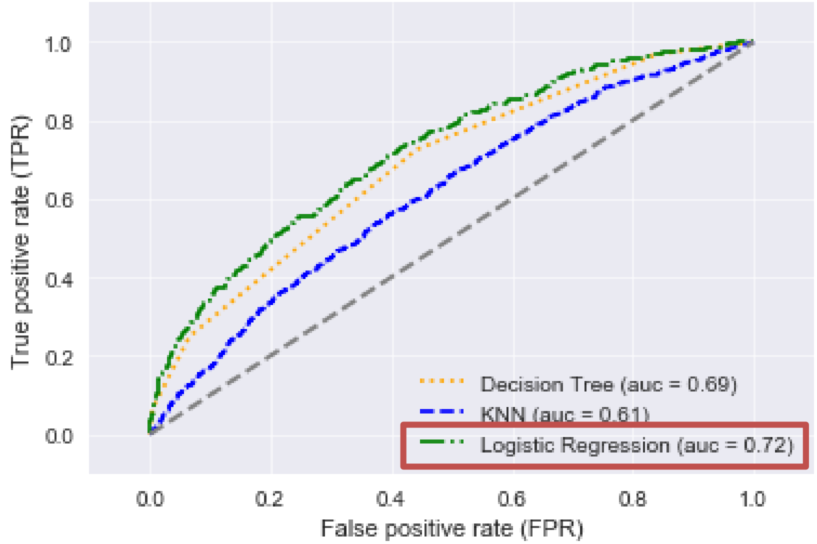
| Class | Precision | Recall | F1-score |
|---|---|---|---|
| Fully paid | 0.83 | 0.99 | 0.90 |
| Charged off | 0.62 | 0.06 | 0.12 |
| **Accuracy** | **0.82** | **F-measure** | **0.51** |

### Logistic Regression

| Class | Precision | Recall | F1-score |
|---|---|---|---|
| Fully paid | 0.83 | 0.99 | 0.90 |
| Charged off | 0.72 | 0.09 | 0.16 |
| **Accuracy** | **0.83** | **F- measure** | **0.53** |

## ROC/AUC

| | |
|---|---|
| **Decision Tree** | 0.68 (+/- 0.02) |
| **KNN** | 0.59 (+/- 0.03) |
| **Logistic Regression** | 0.71 (+/- 0.02) |



## Cross Validation Accuracy

| | |
|---|---|
| **Decision Tree** | 0.822 +/- 0.004 |
| **KNN** | 0.814 +/- 0.003 |
| **Logistic Regression** | 0.821 +/- 0.006 |

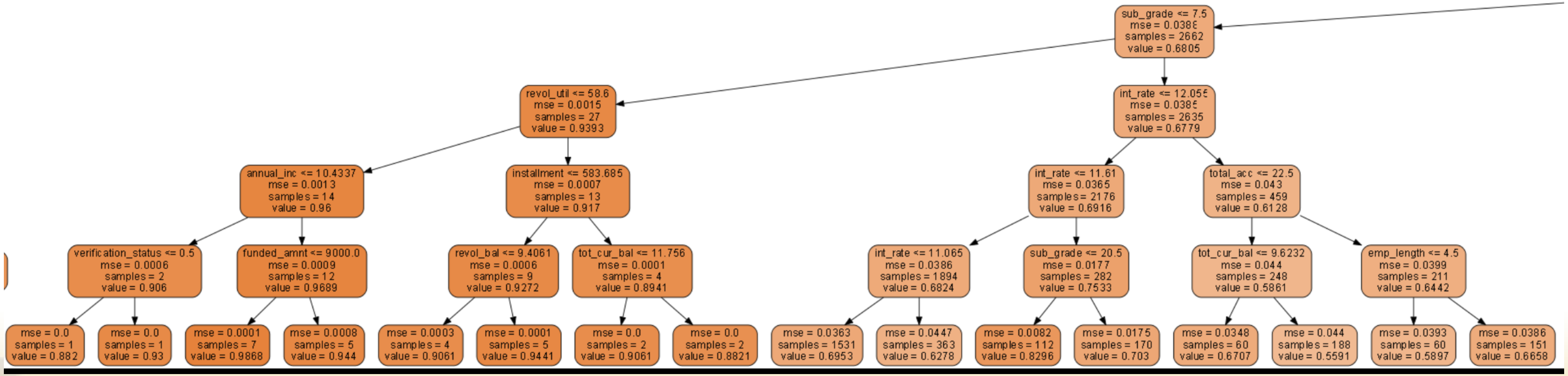
# The final classification model was used on each Current and Late Loan to determine likelihood of being Charged Off

- Taken in aggregate, this gives the proportion of the Current and Late classes that will become Charged Off
  - Of the 601,779 loans in **Current** status, we predict that **1.34%** of the loans will be Charged Off, or **8,046**
  - Of the 21,420 loans in **Late** status, we predict that **10.29%** of the loans will be Charged Off, or **2,203**

# A Regression Model is used to predict the expected loss from the pool of Current and Late loans

- Using the **Regression Tree** technique within the Charged Off classifications of loans, we could estimate the **expected value** for the fraction of a loan that would be not be repaid before being Charged Off, which we named the **Loss Fraction**

- We optimized for **max depth** using Grid Search while using standard minimums for node split, leaf size, and impurity values with a Gini criterion
  - Resulting tree depth was **8** layers



Phase IV - Regression

# In total we can identify how much should be written off from the pool of Current and Late loans

- The MSE’s across the training and test sets for the Regression Tree models were **0.023** and **0.024** respectively, indicating good generalizability

- The model gives a **loss fraction** value for **each instance** predicted to be Charged Off, with the expected value of accounting loss within the loan status category determined by:

$$\sum lossfraction \times funded\ amnt$$

- The total expected value of loss for the **Current** loans is **$129,904,001**
- The total expected value of loss for the **Late** loans is **$34,500,104**

# The models will be deployed to adjust the value of particular loan portfolios based on the risk of the contents

- They can serve both Lending Tree and individual lenders in adjusting their books for the true expected value of a particular portfolio as well as projecting the value of additions to their portfolios
- The combined model does have a tendency to over-estimate losses, likely due to the classification's tendency to classify loans with higher principle values as more likely to be Charged Off
  - This will probably make the accountant's happy as it constitutes an over-estimation of default values and thus a conservative estimate of portfolio value
- The model doesn't evaluate time-based inputs due to the assumption that a loan taken out this year is no more likely to default than one last year, all other things being equal
  - This may ignore changing macroeconomic factors that might cause changes in default probability, for which we have no leading indicators
- From an ethical standpoint, it is important to understand that these predictions are not robust enough to justify drastic action, such as breaching contract against likely defaulters before they have actually violated any terms

# Time to put these models to work and value some loan portfolios

Questions?