Machine Learning Engineer Capstone Project

Project Report

Project Setup and Installation

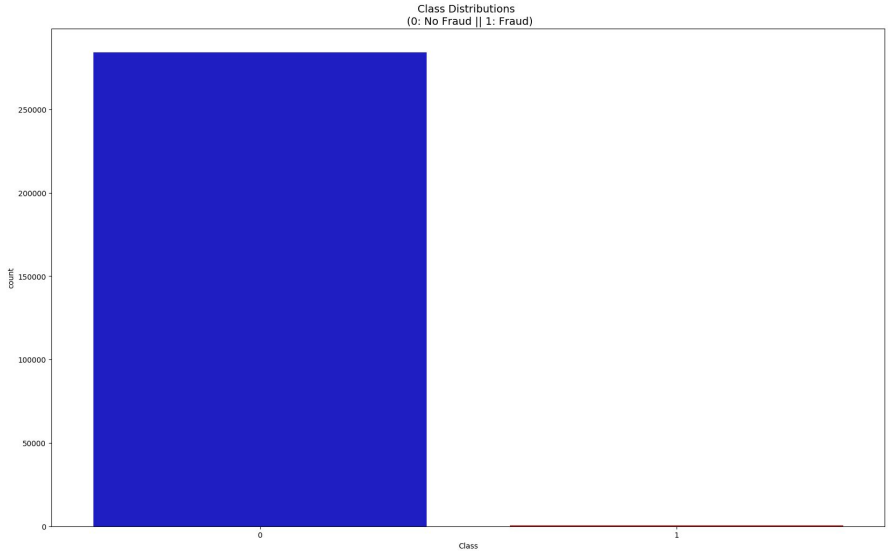
- Setup
  - Pipeline
    - Download Training Data: First you will have to download the training data to local storage.
    - Data Exploration and processing: handle imbalanced data, remove outliers,...
    - Data Splitting: using cross validation with k-fold is 5.
    - Train model: run that training script and train your model.
    - Hyperparameter Tuning: tuning some classification models and choose the best estimator.
    - Model Evaluation and save model to disk.
  - Dependencies
    - Python version: python>= 3.7
    - Packages: you can install necessary packages by executing the command: *pip install -r requirements.txt*
- Installation
  - For this project, we strongly advise utilizing SageMaker Studio within the AWS workspace provided by the course. This approach will greatly streamline the setup process, reducing the need for complex installations.

Data Exploration

- Data has not null values
- Data has 30 columns are *'Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18', 'V19', 'V20','V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28', 'Amount','Class'.*
- Fraudulent transactions represent only 0.172% of the total transactions, which makes the task of identifying them much more complex.

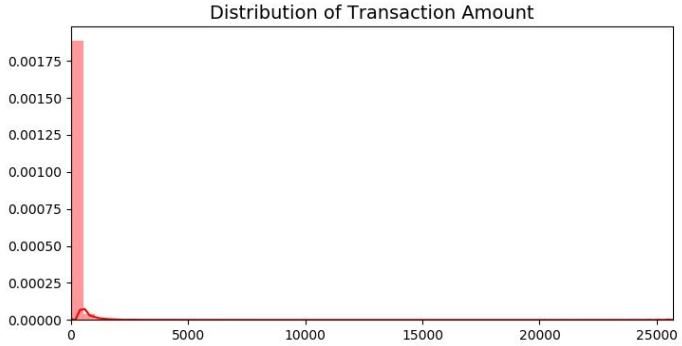
Data Exploration

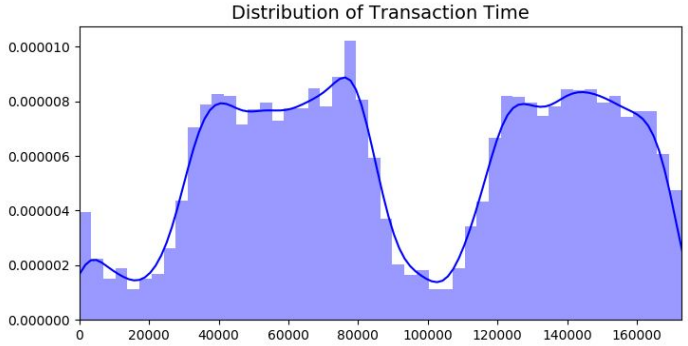
- Through analyzing the distributions, we gain insights into the skewness of these features, while also getting a glimpse into the distributions of other variables. Various techniques exist to reduce skewness in these distributions, which we will look to incorporate into this notebook in upcoming iterations.



Data Exploration

- Initially, we'll standardize the 'Time' and 'Amount' columns to align with the scaling of other columns. Concurrently, we'll generate a balanced subset of the dataframe, ensuring equal representation of Fraud and Non-Fraud cases. This balanced distribution aids our algorithms in discerning patterns that accurately differentiate between fraudulent and legitimate transactions.





Data Exploration

- It spans across two days and includes a total of 284,807 transactions, out of which 492 were fraudulent. A significant challenge with this dataset, and indeed the issue of credit card fraud detection in general, is the high imbalance between legitimate and fraudulent transactions.

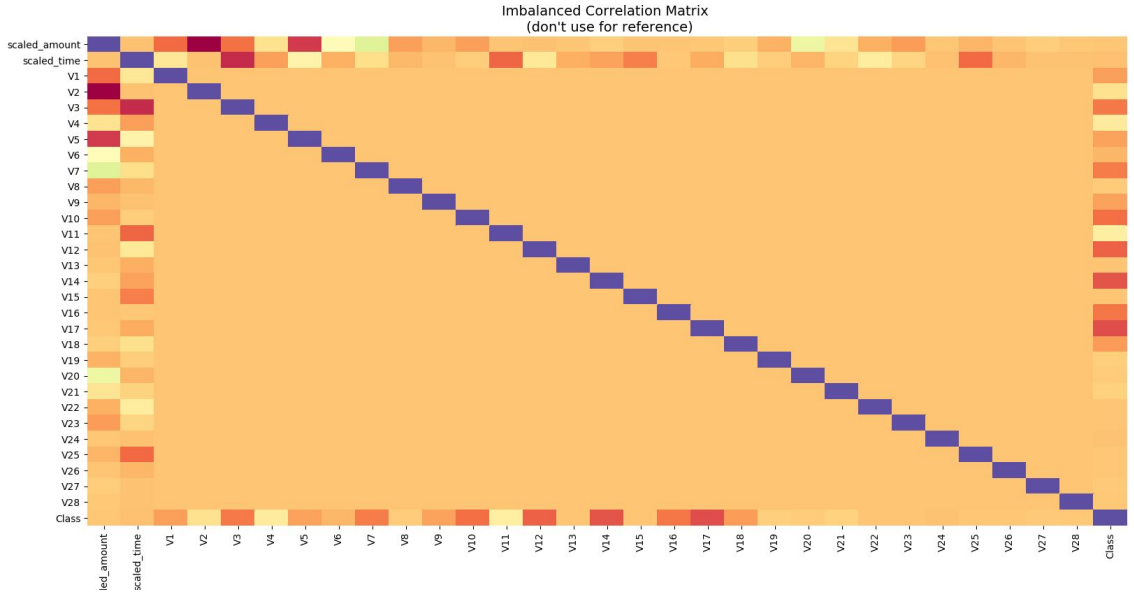
	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...
count	284807.000000	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	...
mean	94813.859575	3.918649e-15	5.682686e-16	-8.761736e-15	2.811118e-15	-1.552103e-15	2.040130e-15	-1.698953e-15	-1.893285e-16	-3.147640e-15	...
std	47488.145955	1.958696e+00	1.651309e+00	1.516255e+00	1.415869e+00	1.380247e+00	1.332271e+00	1.237094e+00	1.194353e+00	1.098632e+00	...
min	0.000000	-5.640751e+01	-7.271573e+01	-4.832559e+01	-5.683171e+00	-1.137433e+02	-2.616051e+01	-4.355724e+01	-7.321672e+01	-1.343407e+01	...
25%	54201.500000	-9.203734e-01	-5.985499e-01	-8.903648e-01	-8.486401e-01	-6.915971e-01	-7.682956e-01	-5.540759e-01	-2.086297e-01	-6.430976e-01	...
50%	84692.000000	1.810880e-02	6.548556e-02	1.798463e-01	-1.984653e-02	-5.433583e-02	-2.741871e-01	4.010308e-02	2.235804e-02	-5.142873e-02	...
75%	139320.500000	1.315642e+00	8.037239e-01	1.027196e+00	7.433413e-01	6.119264e-01	3.985649e-01	5.704361e-01	3.273459e-01	5.971390e-01	...
max	172792.000000	2.454930e+00	2.205773e+01	9.382558e+00	1.687534e+01	3.480167e+01	7.330163e+01	1.205895e+02	2.000721e+01	1.559499e+01	...

8 rows × 31 columns

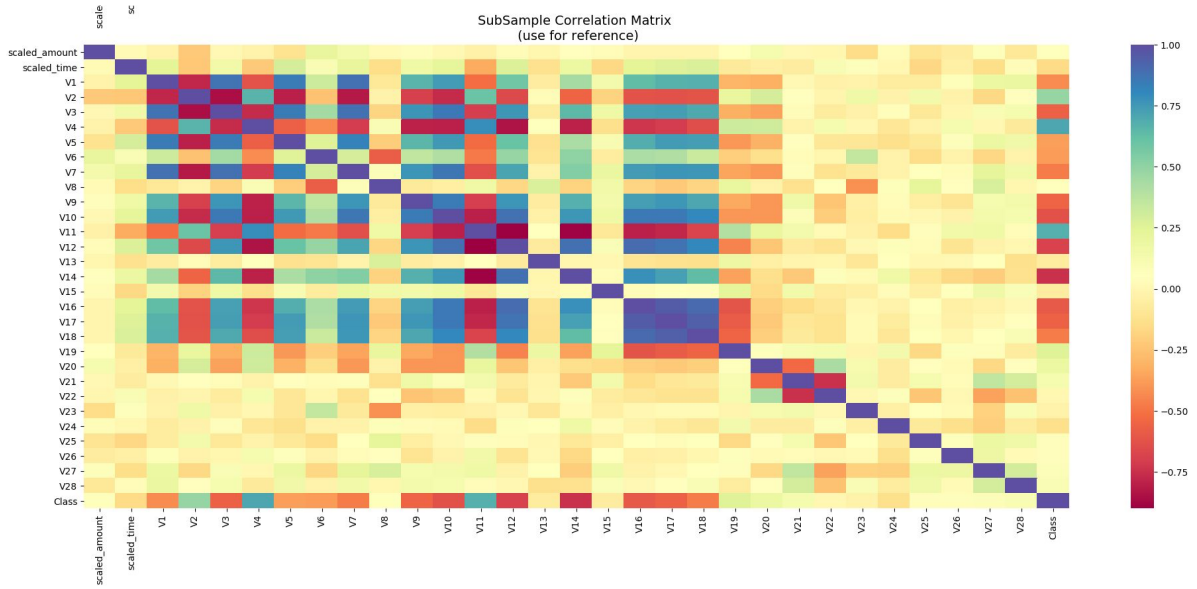
Exploratory Visualization

- Correlation matrices are integral to gaining insights into our data. They enable us to identify features that significantly sway the likelihood of a transaction being fraudulent. However, it's crucial that we utilize the appropriate dataframe (in this case, the subsample) to accurately observe which features exhibit strong positive or negative correlations in relation to fraudulent transactions.

Exploratory Visualization



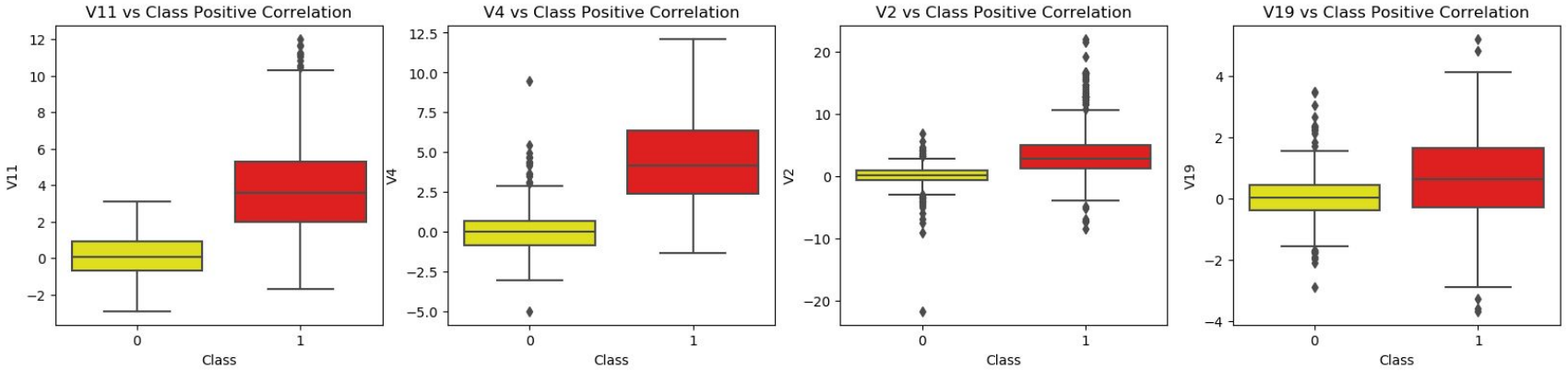
Exploratory Visualization



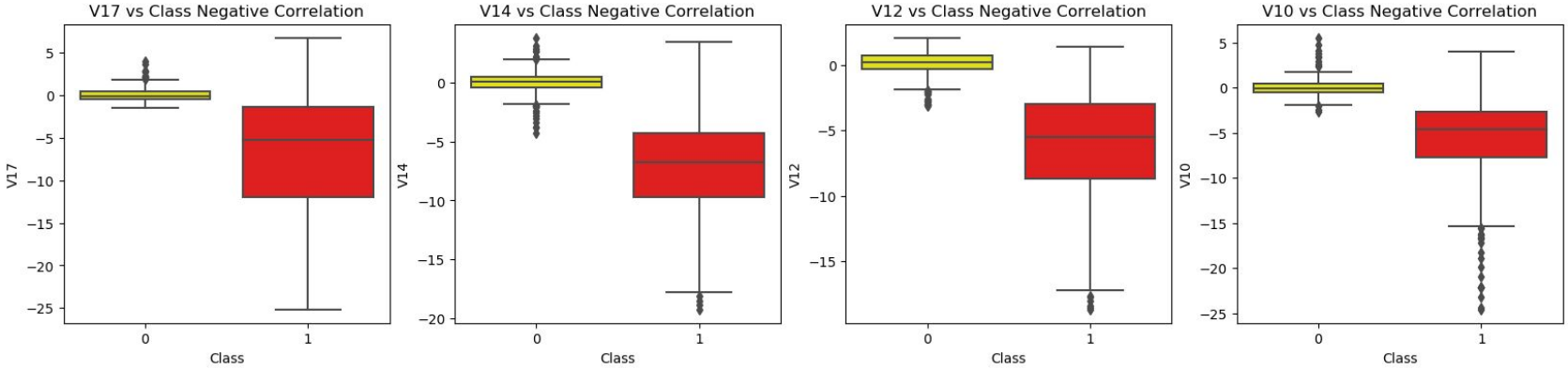
Exploratory Visualization

- Negative Correlations: V17, V14, V12 and V10 are negatively correlated. Notice how the lower these values are, the more likely the end result will be a fraud transaction.
- Positive Correlations: V2, V4, V11, and V19 are positively correlated. Notice how the higher these values are, the more likely the end result will be a fraud transaction.
- BoxPlots: We will use boxplots to have a better understanding of the distribution of these features in fradulent and non fradulent transactions.

Exploratory Visualization



Exploratory Visualization



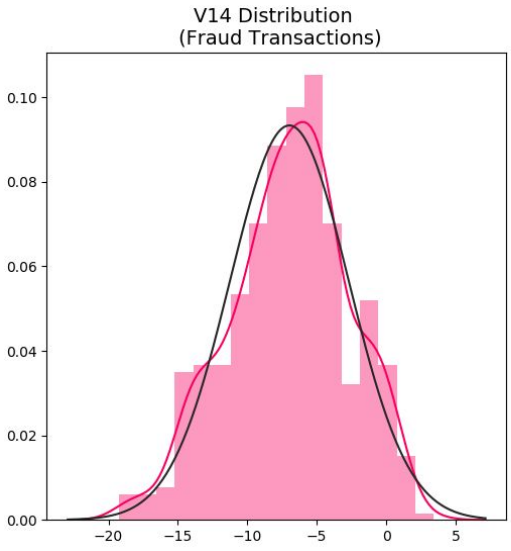
Data Processing

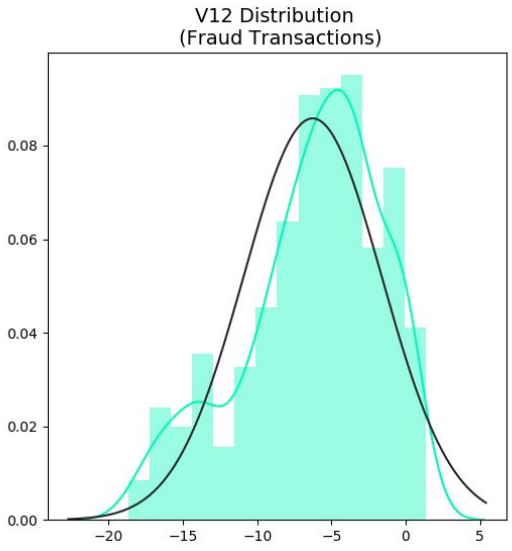
- The primary goal in this segment is to eliminate "extreme outliers" from features exhibiting a strong correlation with our classes. This action will contribute beneficially to enhancing the precision of our models.
- Interquartile Range Method:
  - Interquartile Range (IQR): Our approach involves calculating the interquartile range, which is the difference between the 75th and 25th percentiles. Our objective is to establish a threshold that goes beyond these percentiles, such that if any instance crosses this boundary, it gets eliminated.
  - Boxplots: Besides readily identifying the 25th and 75th percentiles, represented by the ends of the boxes, it's also straightforward to spot extreme outliers, which are the points falling beyond the upper and lower extremes.

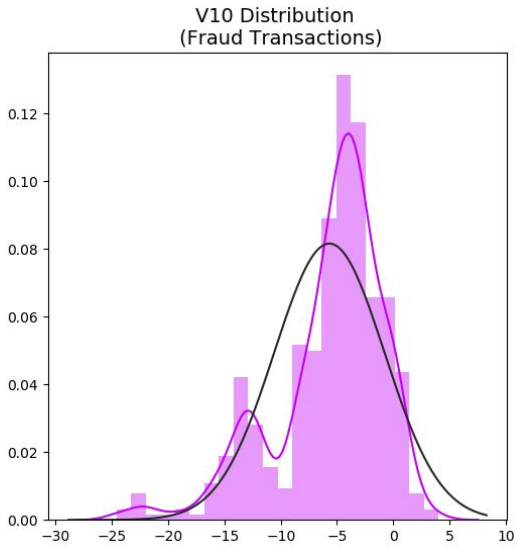
Data Processing

- Visualize Distributions: We first start by visualizing the distribution of the feature we are going to use to eliminate some of the outliers. V14 is the only feature that has a Gaussian distribution compared to features V12 and V10.
- Determining the threshold: After we decide which number we will use to multiply with the iqr (the lower more outliers removed), we will proceed in determining the upper and lower thresholds by substrating q25 - threshold (lower extreme threshold) and adding q75 + threshold (upper extreme threshold).
- - Boxplot Representation: Visualize through the boxplot that the number of "extreme outliers" have been reduced to a considerable amount.

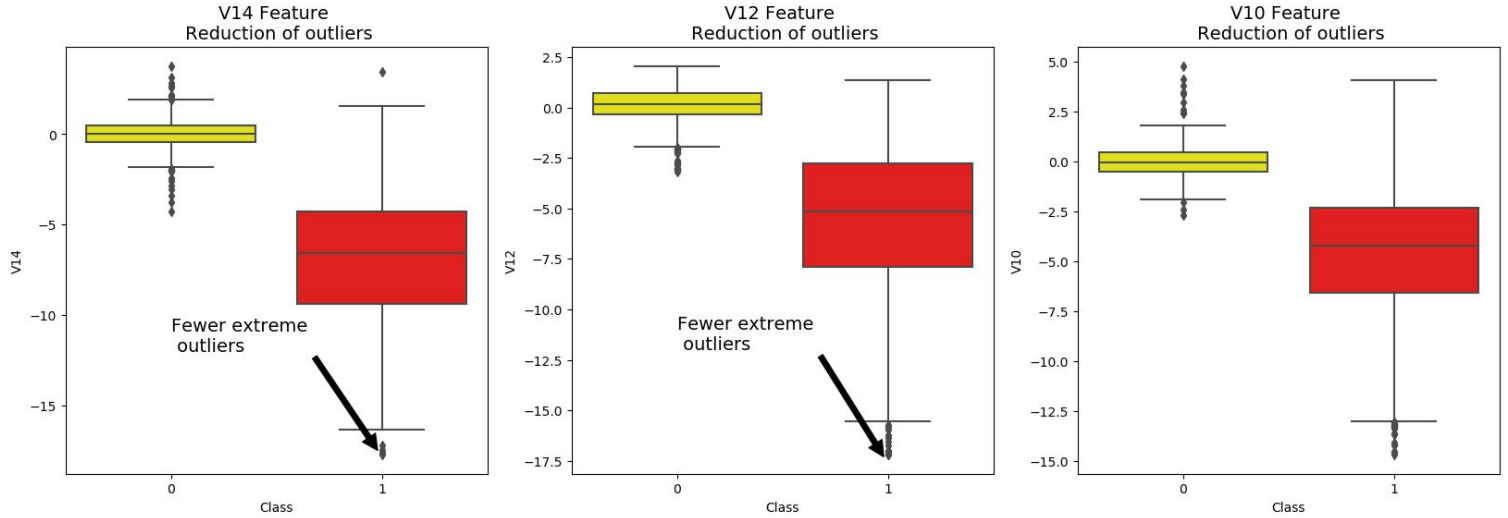
Data Processing
Dist Plot







Data Processing
Boxplot

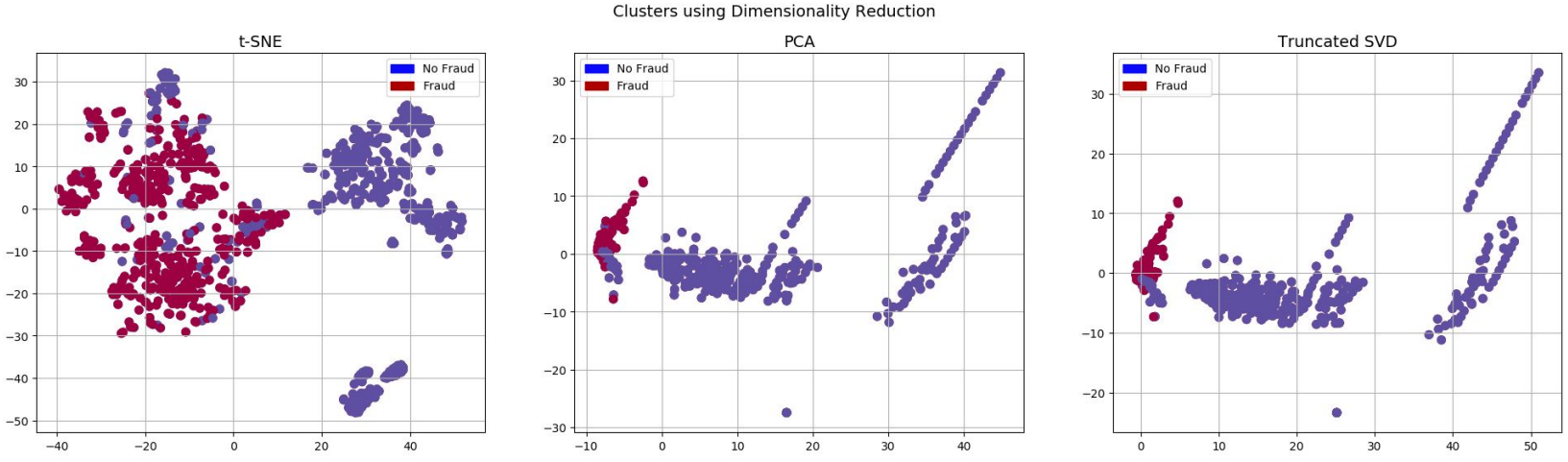


Data Processing

Dimensionality Reduction and Clustering:

- t-SNE algorithm can pretty accurately cluster the cases that were fraud and non-fraud in our dataset.
- Although the subsample is pretty small, the t-SNE algorithm is able to detect clusters pretty accurately in every scenario (I shuffle the dataset before running t-SNE)
- This gives us an indication that further predictive models will perform pretty well in separating fraud cases from non-fraud cases.

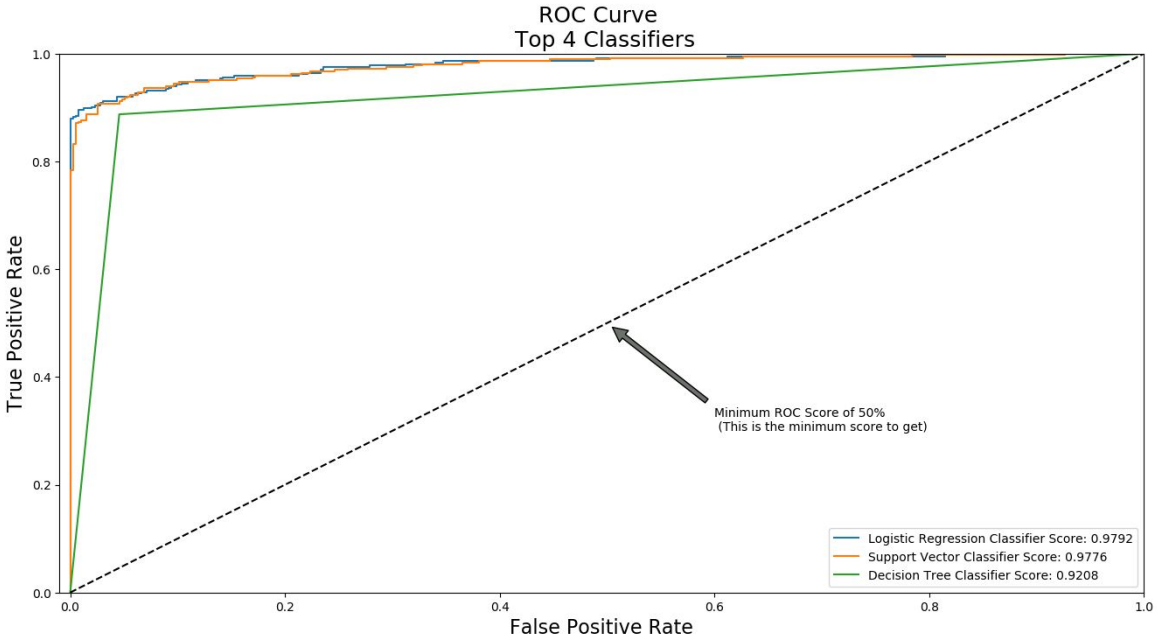
Data Processing
Dimensionality Reduction and Clustering



Algorithms and Techniques

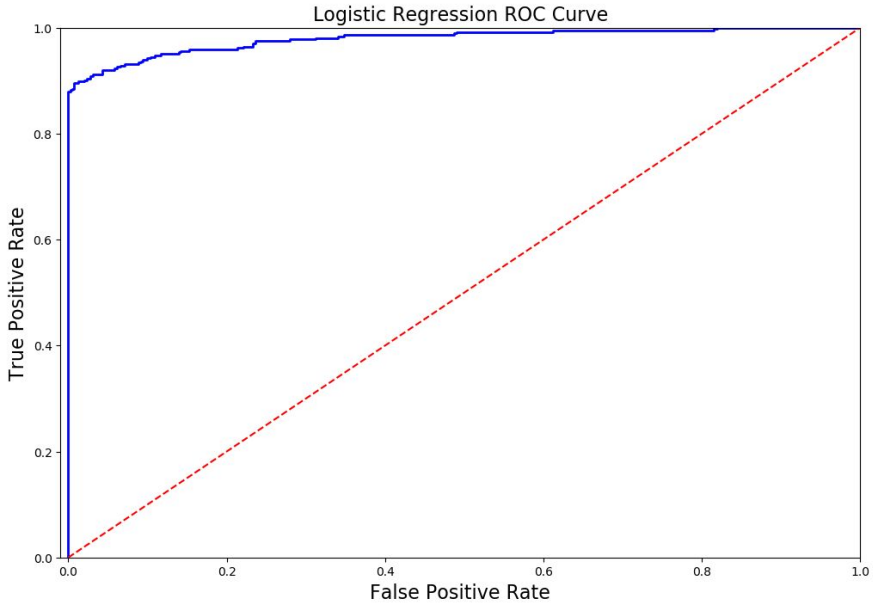
- UnderSampling
  - Logistic Regression classifier is more accurate than the other three classifiers in most cases. (We will further analyze Logistic Regression)
  - GridSearchCV is used to determine the paremeters that gives the best predictive score for the classifiers.
  - Logistic Regression has the best Receiving Operating Characteristic score (ROC), meaning that LogisticRegression pretty accurately separates fraud and non-fraud transactions.
- SMOTE Technique (Over-Sampling)
  - Solving the Class Imbalance: SMOTE creates synthetic points from the minority class in order to reach an equal balance between the minority and majority class.
  - Location of the synthetic points: SMOTE picks the distance between the closest neighbors of the minority class, in between these distances it creates synthetic points.
  - Final Effect: More information is retained since we didn't have to delete any rows unlike in random undersampling.
  - Accuracy || Time Tradeoff: Although it is likely that SMOTE will be more accurate than random under-sampling, it will take more time to train since no rows are eliminated as previously stated.

Benchmark

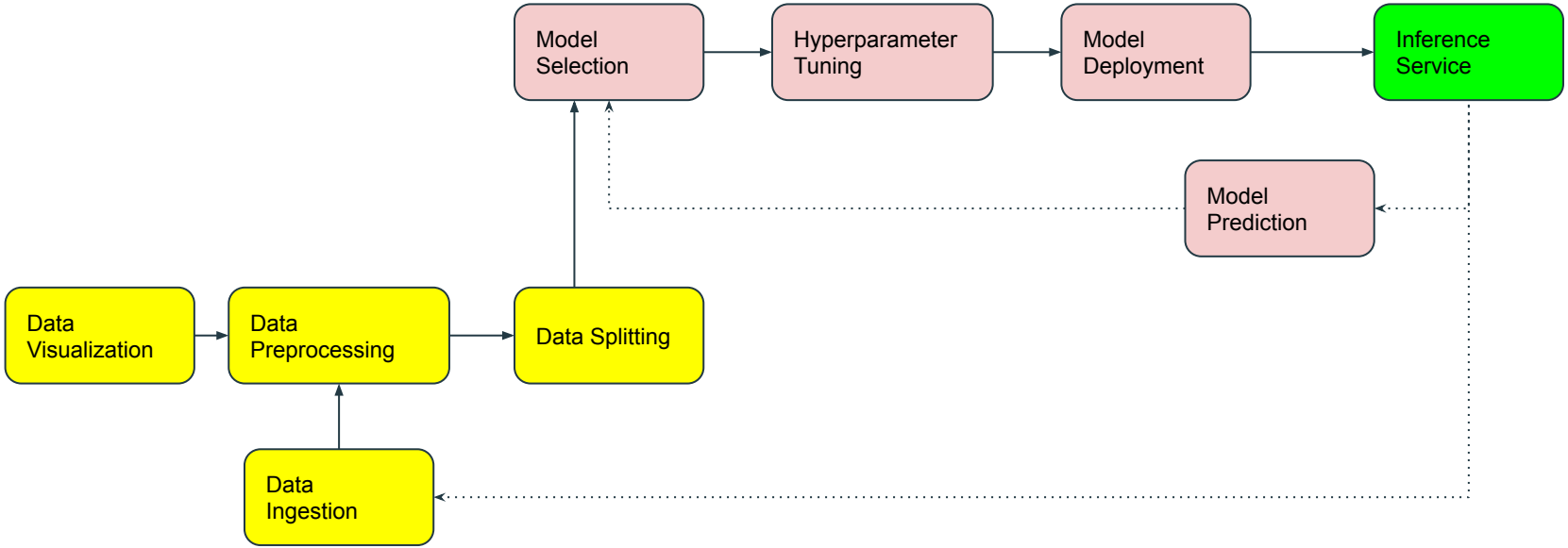

ROC Curve
Top 4 Classifiers
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Minimum ROC Score of 50%
(This is the minimum score to get)
Logistic Regression Classifier Score: 0.9792
Support Vector Classifier Score: 0.9776
Decision Tree Classifier Score: 0.9208

Benchmark
Logistic regression

	Technique	Score
0	Random UnderSampling	0.921053
1	Oversampling (SMOTE)	**0.987869**



Workflow


Model Selection
Hyperparameter Tuning
Model Deployment
Inference Service
Model Prediction
Data Visualization
Data Preprocessing
Data Splitting
Data Ingestion