HW20 – Report

# Analysis Overview

The purpose of this analysis is to determine the creditworthiness of borrowers based on their loan risk using data from a peer-to-peer lending services company. We will be using various techniques to train and evaluate the model to predict loan status.

After some inspection of the data, we found that there was an imbalance as most people do not default their loans.

We started by creating a logistic regression model using the original data and, even though there were no signs of overfitting, the false positive and negatives were a little high (15 and 9% respectively). The model also violated the assumptions of linear models so we concluded that if we used a PCA or a scaler, it would be more accurate and therefore a much better model.

We then did a SV and a KNN and, even though they both reduced false negatives, they either did not reduce or even increase the false positives, so we concluded that they were not the best models either.

# Results (Testing Metrics)

## Log Regression

* Accuracy score: 0.94
* Precision score: 0.92
* Recall score: 0. 95

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

## SV

* Accuracy score: 0.95
* Precision score: 0.92
* Recall score: 0.99

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

## KNN

* Accuracy score: 0. 95
* Precision score: 0.92
* Recall score: 0.99

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

## Decision Tree Classifier

* Accuracy score: 0.92
* Precision score: 0.92
* Recall score: 0.92

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

## Random Forest

* Accuracy score: 0.93
* Precision score: 0.95
* Recall score: 0.92

A screenshot of a computer

Description automatically generated A screenshot of a computer screen

Description automatically generated

## ADA Boost Classifier

* Accuracy score: 0.95
* Precision score: 0.99
* Recall score: 0.92

A screenshot of a computer screen

Description automatically generated A screenshot of a computer screen

Description automatically generated

## Gradient Boost Classifier

* Accuracy score: 0.95
* Precision score: 0.99
* Recall score: 0.92

A screenshot of a computer screen

Description automatically generated A screenshot of a computer screen

Description automatically generated

# Summary

In summary, either AdaBoost, GradientBoost or KNN are good and viable options. Even though KNN is the most explainable of the three, Ada or Gradient Boost are probably the most adaptable models. My final pick would be GradienBoost because it has the best distribution of features.