****

**MODULE 2**

**AIDI 1003 – CAPSTONE TERM 1**

**AMIT MARAJ**

AHNCH BALA 100424062

SONAKSHI KARKERA 100720763

SURBHI THAKUR 100732335

ARUN KALAESWARAN 100771700

**November 9, 2019**

# ***Executive Summary***

The following report will outline the results of various algorithms we used to test against our dataset. Our dataset contains reviews of restaurants made by customers and has been provided to us by Yelp. As such, NLP has been a huge part of our analysis. The following are some steps we took to examine the data:

1. Data Cleaning
2. Feature extraction: Attributes column consisted of dictionary of important features which we converted into individual columns and dropped attributes column.
3. Feature engineering: Combined all reviews for each restaurant as one and appended it with restaurant data using business\_id.
4. Datatype conversions for relevant features.
5. Dropped columns which consisted of information of the business-like business ID, name, address, state and location
6. We replace columns with Boolean values with 0 and 1 and null values with most frequent values
7. Handling missing values.

• If the missing values of greater than 50% than we drop the columns.

• Replace the null values with most frequent values of the column.

1. Sentiment Analysis:

* We used nltk to preprocess as well as created a function to remove punctuation, stop words, converting to lower case, removing nouns and adjectives using Lemmatization and returns the list of words. We then used Text Blob to get sentiment score for each review in the dataset.

1. Once we have the final dataset, we then split it into 80% test and 20% train.

We have handled imbalance in the dataset using imbalance data handling technique SMOTE where we add data for samples for minority class.

Model Building

Traditional train test method suffers from high variance test problem. We usually the data to train and test set. We will not touch test set until the end of the computation and the final performance evaluation. Then, we can divide the train set to train and validation sets. We use the validation data set to tune the model.

It means that by changing the test set the result of the prediction changes. To overcome this problem, we use k-fold validation method in our train and validation set. We used K-fold cross validation procedure is used to evaluate each algorithm from which we can choose from, each configured with the same random seed to ensure that the same splits to the training data are performed and that each algorithm is evaluated in precisely the same way. We choose 10 folds for our project.

We have used 6 different classification algorithms to compare from:

1. Logistic Regression

2. Linear Discriminant Analysis

3. K-Nearest Neighbors

4. Classification and Regression Trees

5. LDA

6. Support Vector Machines

The plot gives us the spread of the accuracy scores across each cross-validation fold for each algorithm.

Limitations: We faced a few limitations with out data when examining. We realized that in order to achieve even more accurate results we needed more information, particularly the financial information of the restaurants, which were not provided by Yelp. We also noticed that some reviews could be biased or have been made by bots. Yelp did not state whether that was covered or not.

Future Improvements: We will scale the data and do more of feature extractions and engineering in order to get more accurate results.

# ***Algorithm Evaluation***

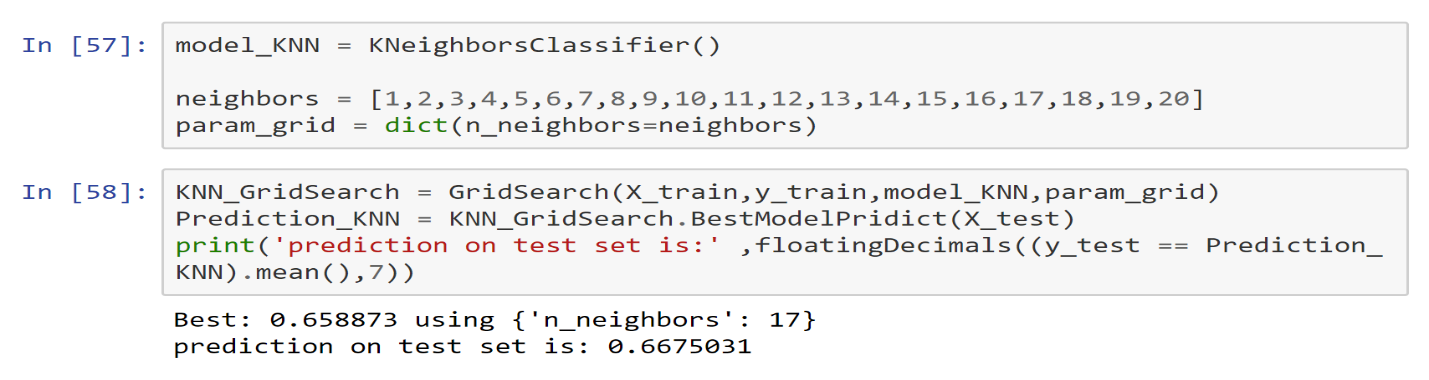
## K-NN

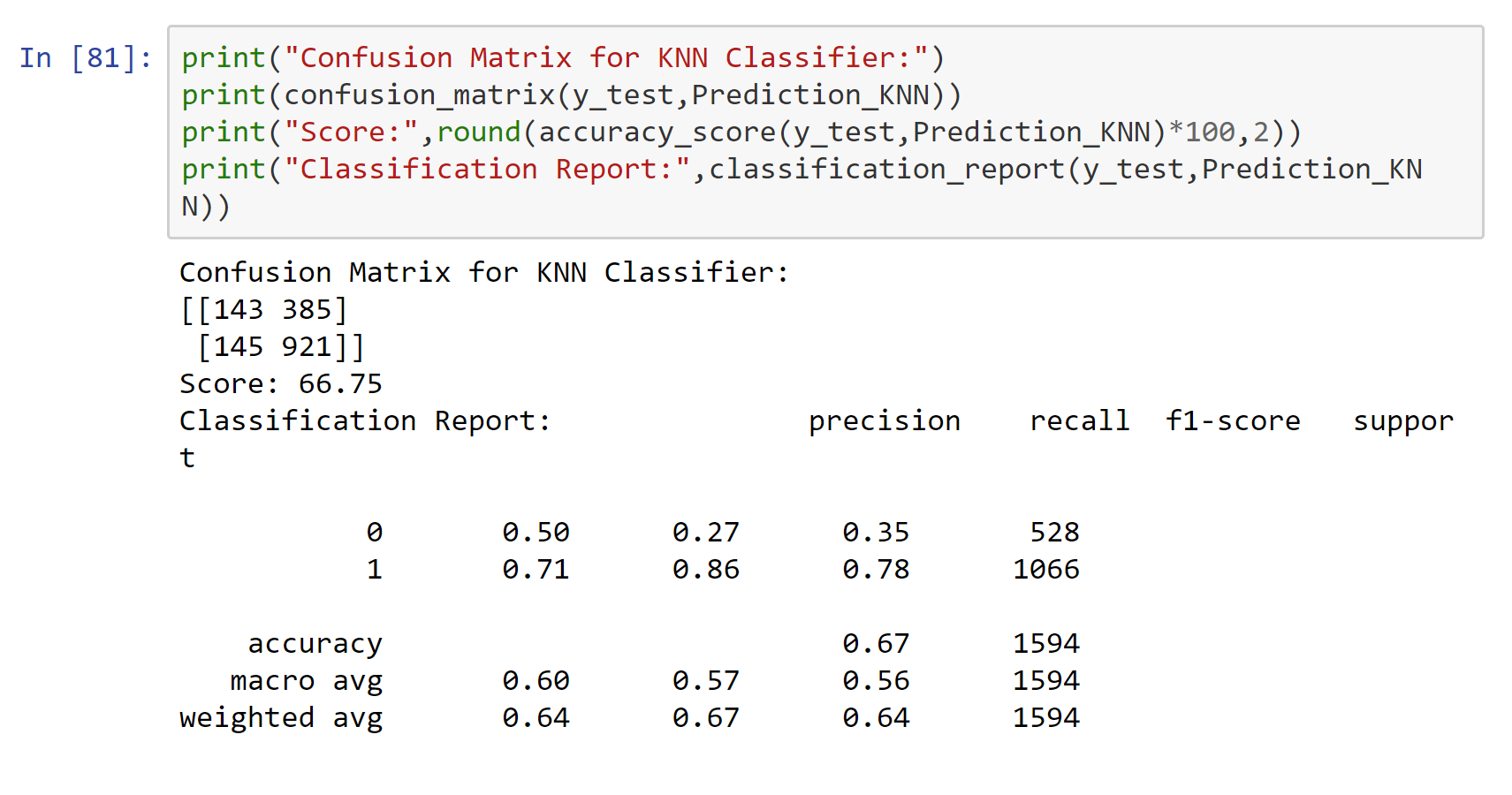
The K Nearest Neighbor algorithm is a robust classifier and is very easy to understand and implement. It is know as the “lazy” learning algorithm. It clusters the data into several classes in order to predict how a new point will be classified.

Pros

* Simple and Intuitive
* Makes no assumptions
* Constantly Evolving and widely used
* Easy to implement

Cons

* Very slow algorithm
* Curse of dimensionality (works well with a small number of input variables)
* Very sensitive to outliers



# Support Vector Machines

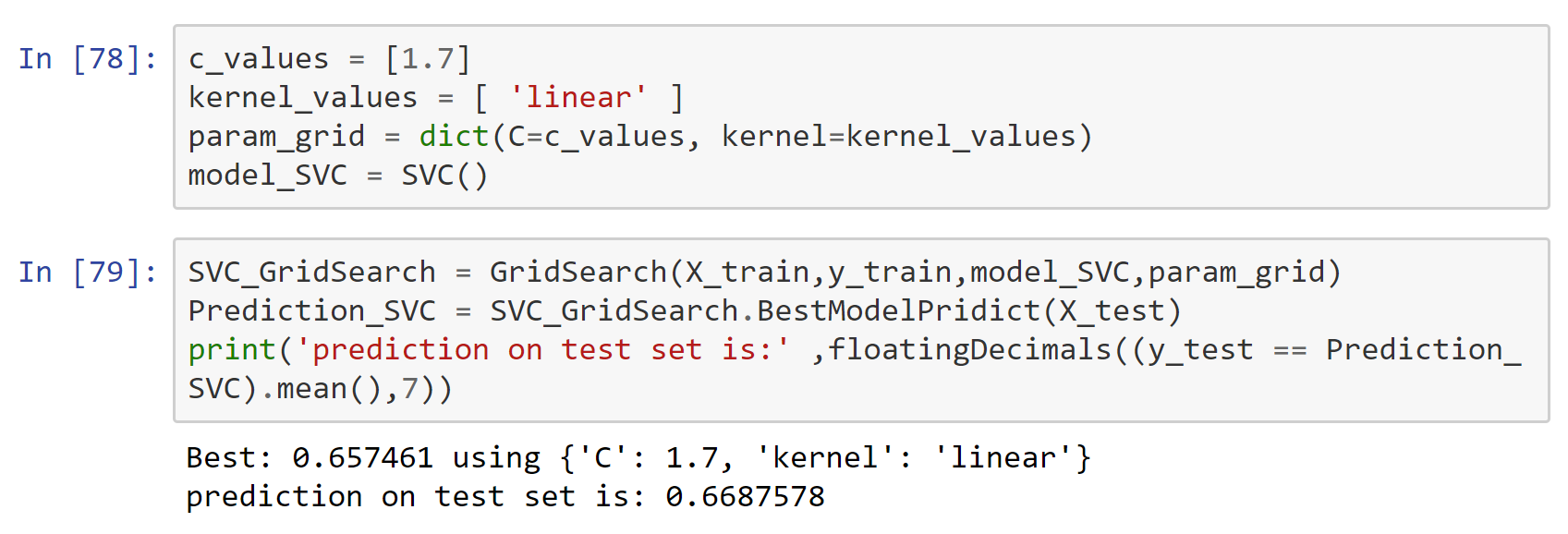
Support vector machines (SVM) use kernels to calculate the distance between two classes. The SVM algorithm then finds the boundary that ensures the greatest distance between those two classes. It acts similarly to linear and logistic regression.

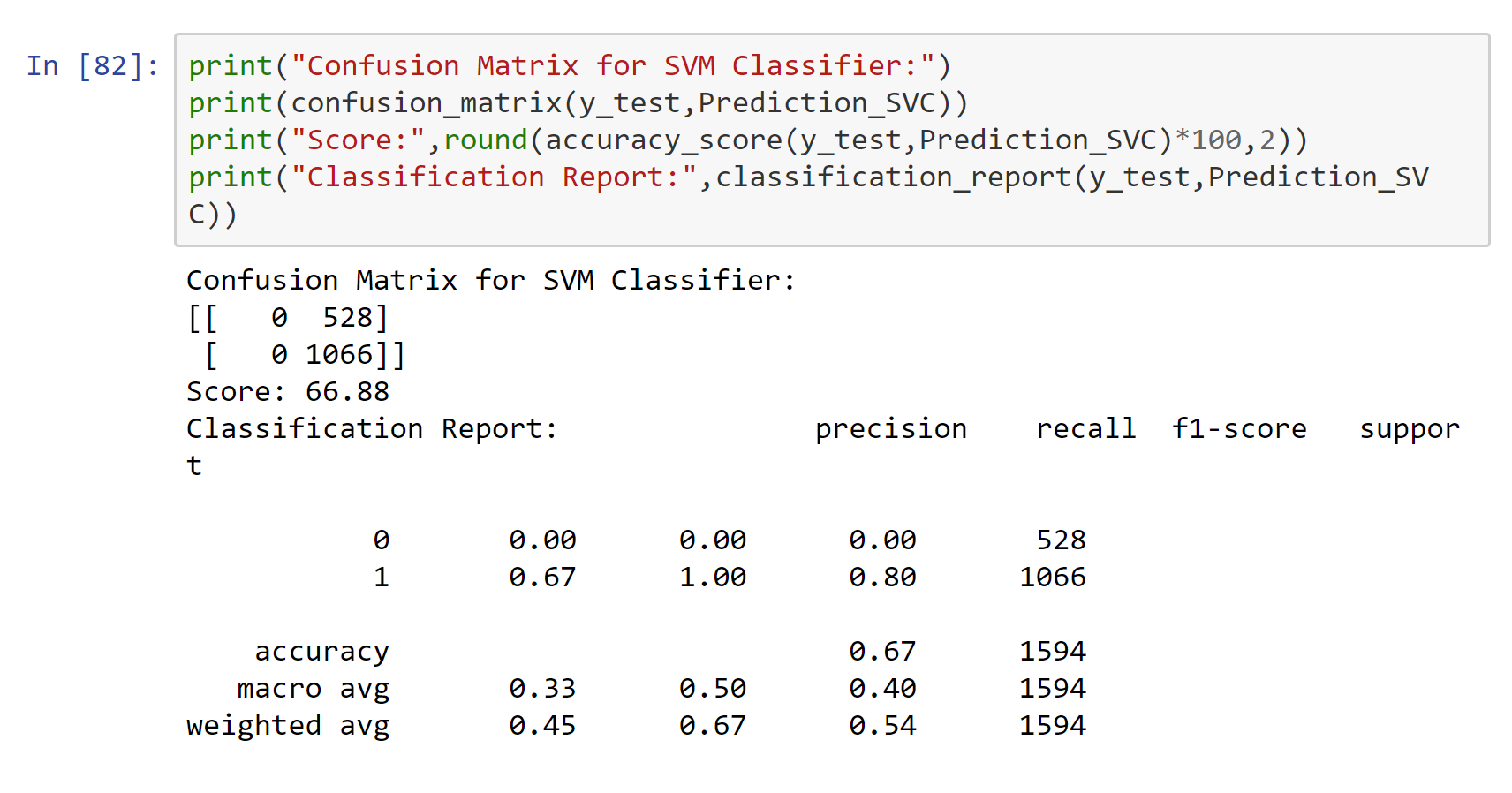
Pros

* Performs will with nonlinear boundary
* Many kernels to choose from to ensure best output
* Robust against overfitting

Cons

* Memory Intensive
* Choosing the correct kernel is a must to ensure highest accuracy





# Random Forest

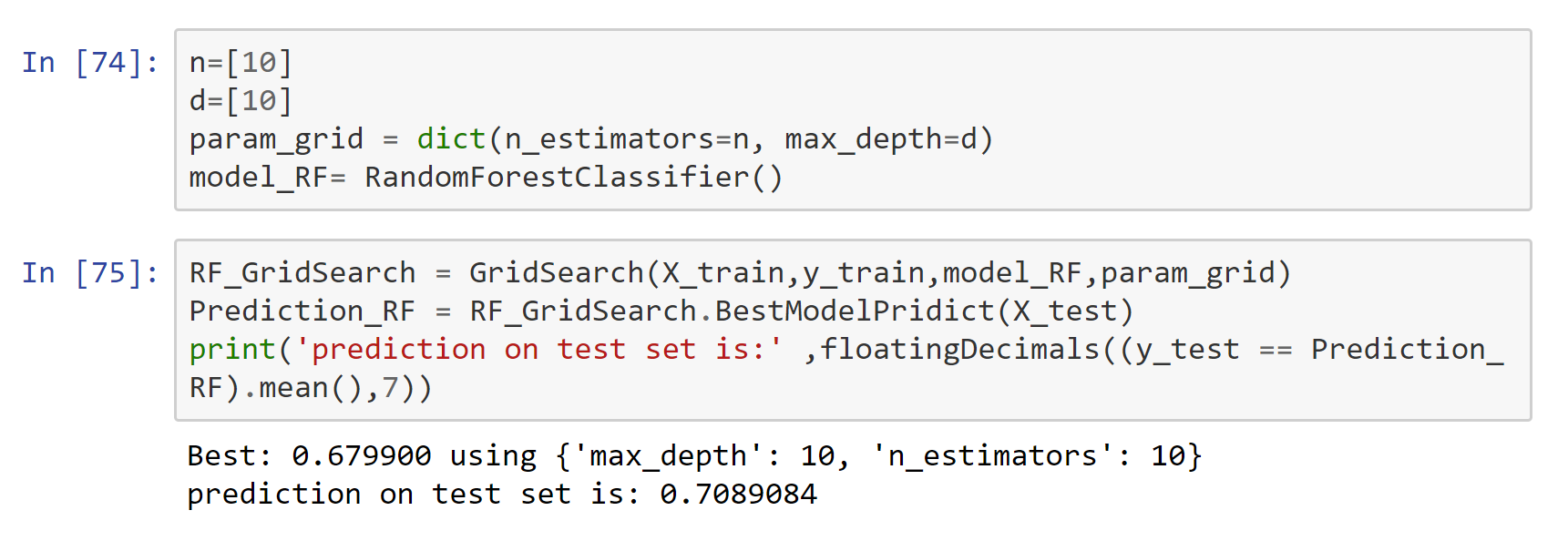
This algorithm follows the same idea as decision trees, but instead of using one tree, there are multiple decision trees used to produce an outcome.

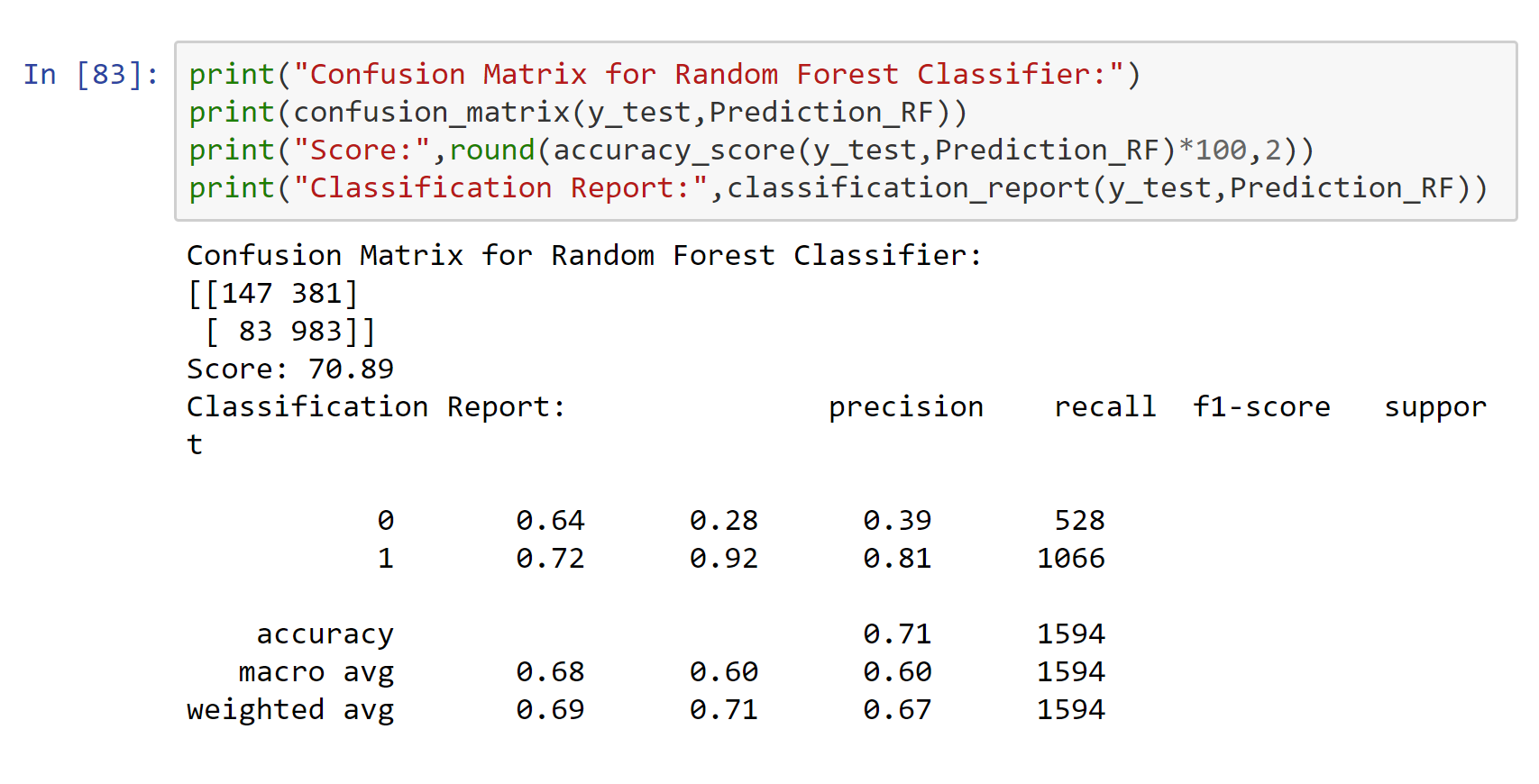
Pros

* Removes correlation between trees
* Reduced variance compared to singular decision tree

Cons

* Harder to visualize compared to regular decision trees
* May require heavy computation





# Decision Tree

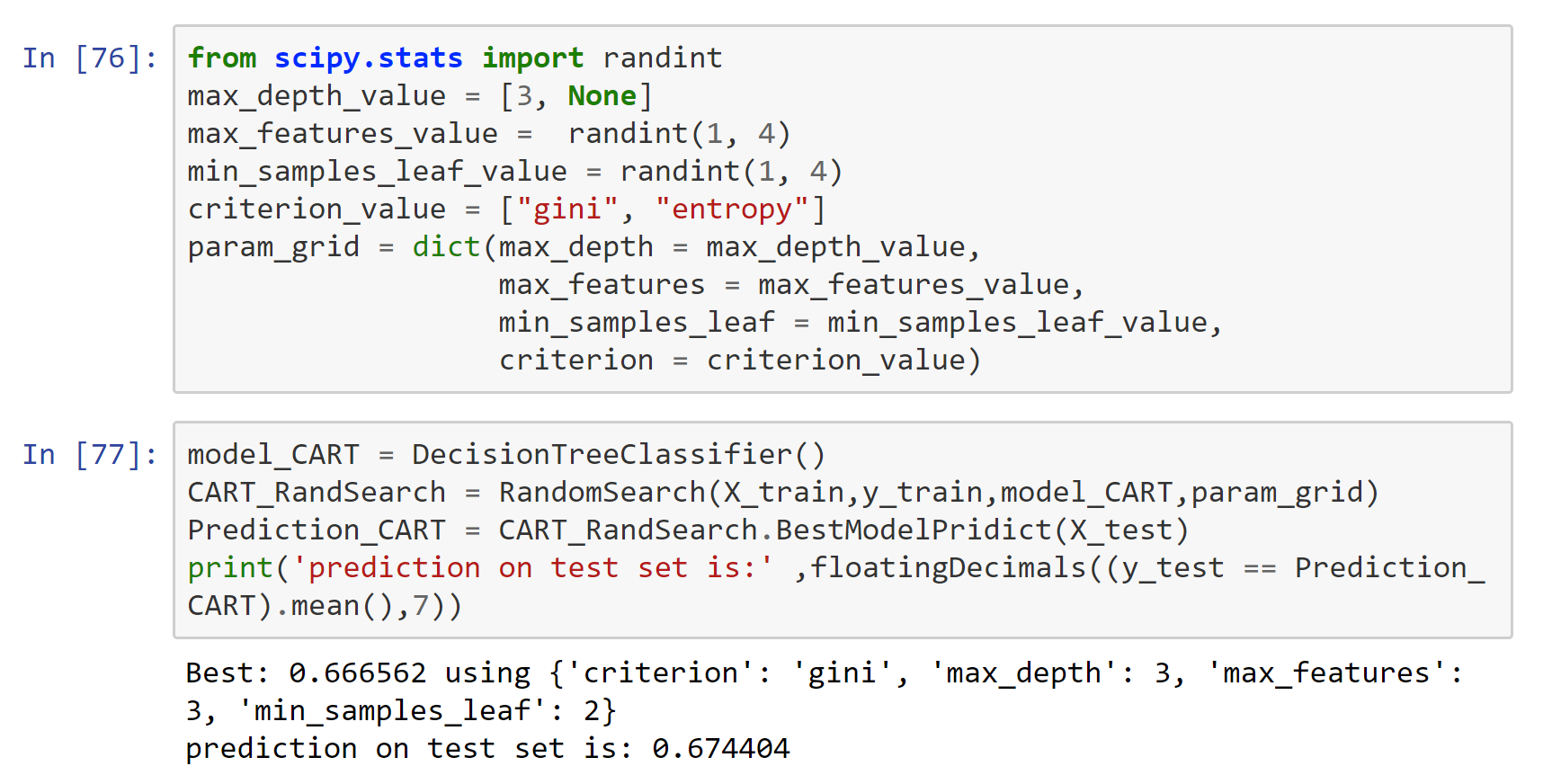
Decision trees is a supervised algorithm that consistently split based on the given parameters until there is an output. It consists of decision nodes and leaves.

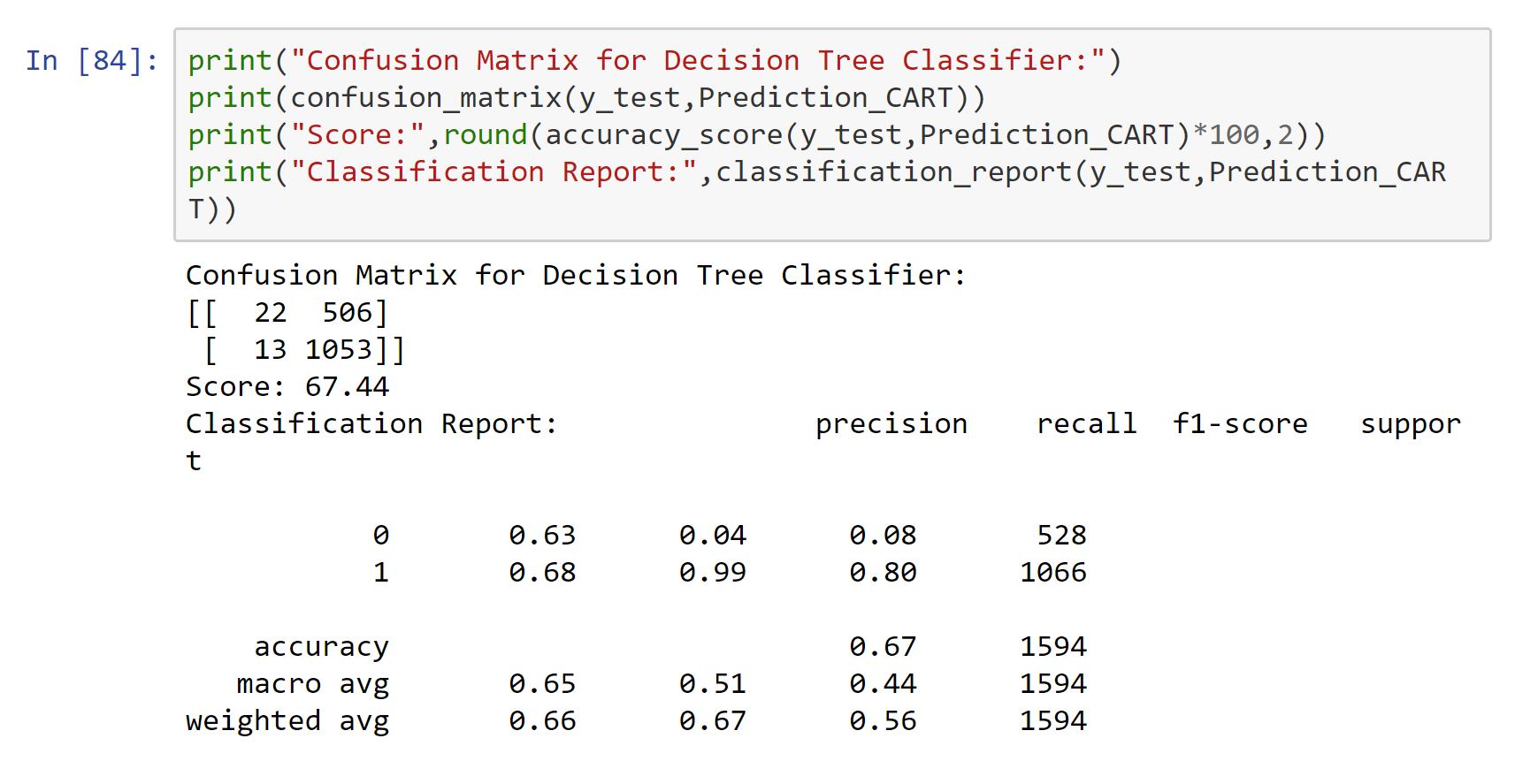
Pros

* Easy to interpret and visually represent
* Mimics human decision making
* Can be used for regression or classification
* Feature selection happens automatically

Cons

* Tends to overfit
* Only axis aligned splits of data





# Logistic Regression

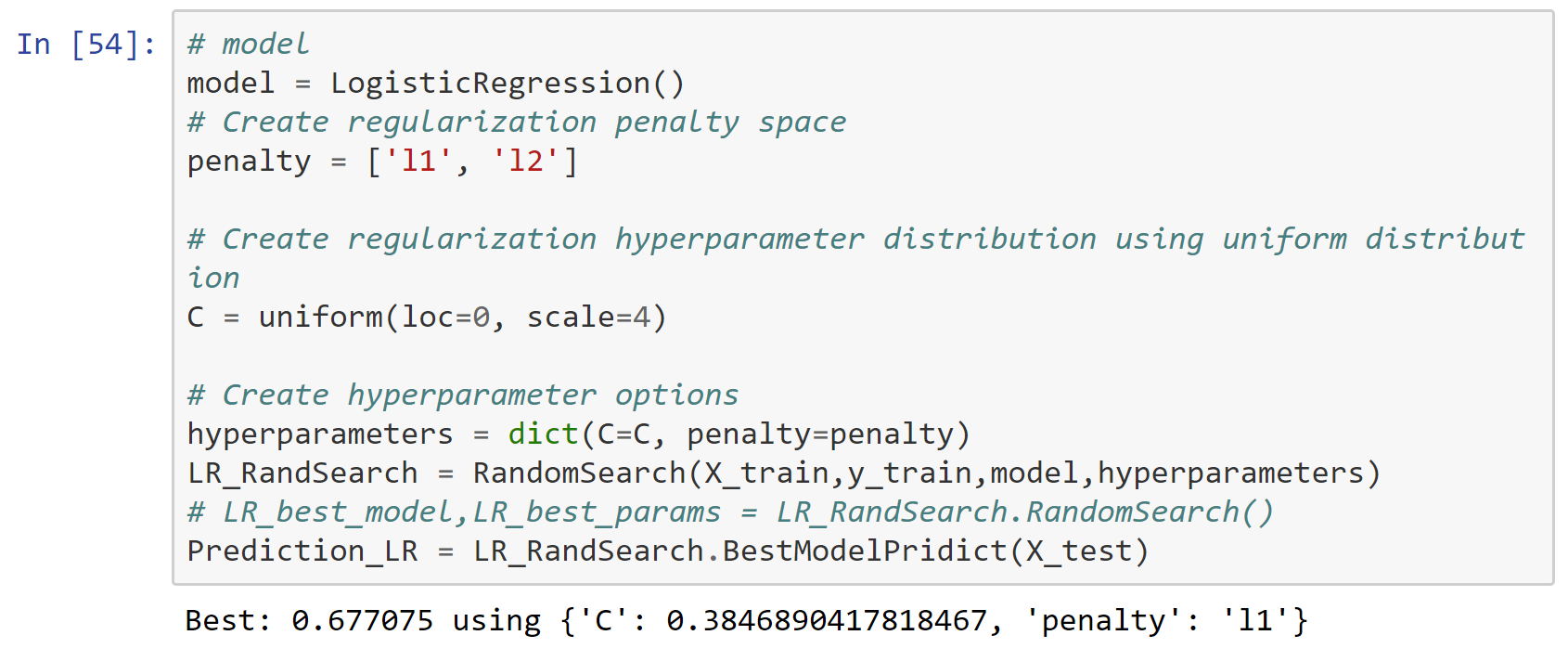
Logistic regression is the classification counterpart of linear regression, used to make predictions between 0 and 1. In our case, we used this to predict whether a restaurant will close or not.

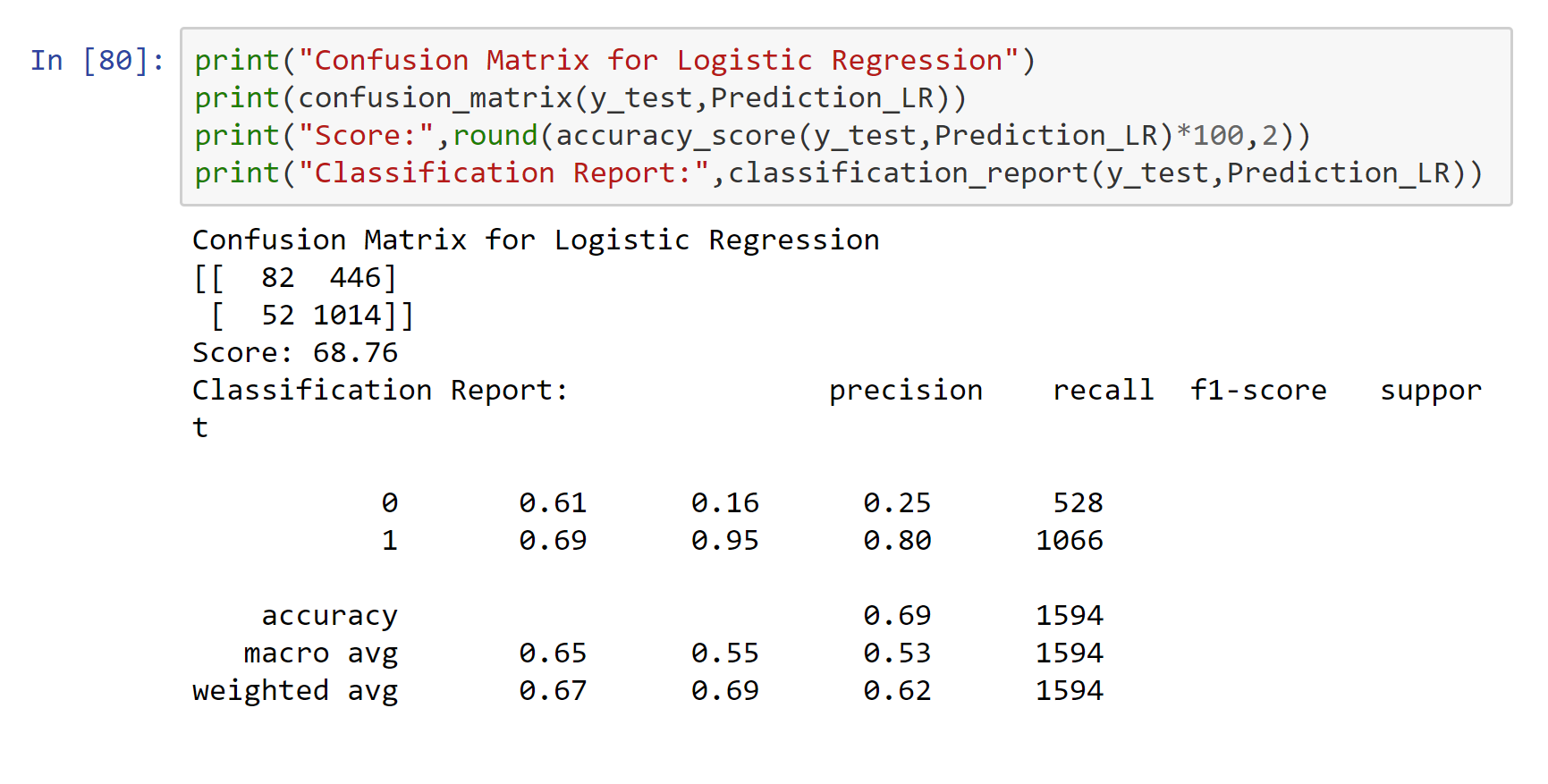
Pros:

* Can be regularized to avoid overfitting
* Easy to implement
* Outputs are easy to interpret
* Adding new data will not cause problems

Cons

* Hard to predict more complex problems
* Will underperform when the decision boundaries are non-linear or are plentiful





# LDA

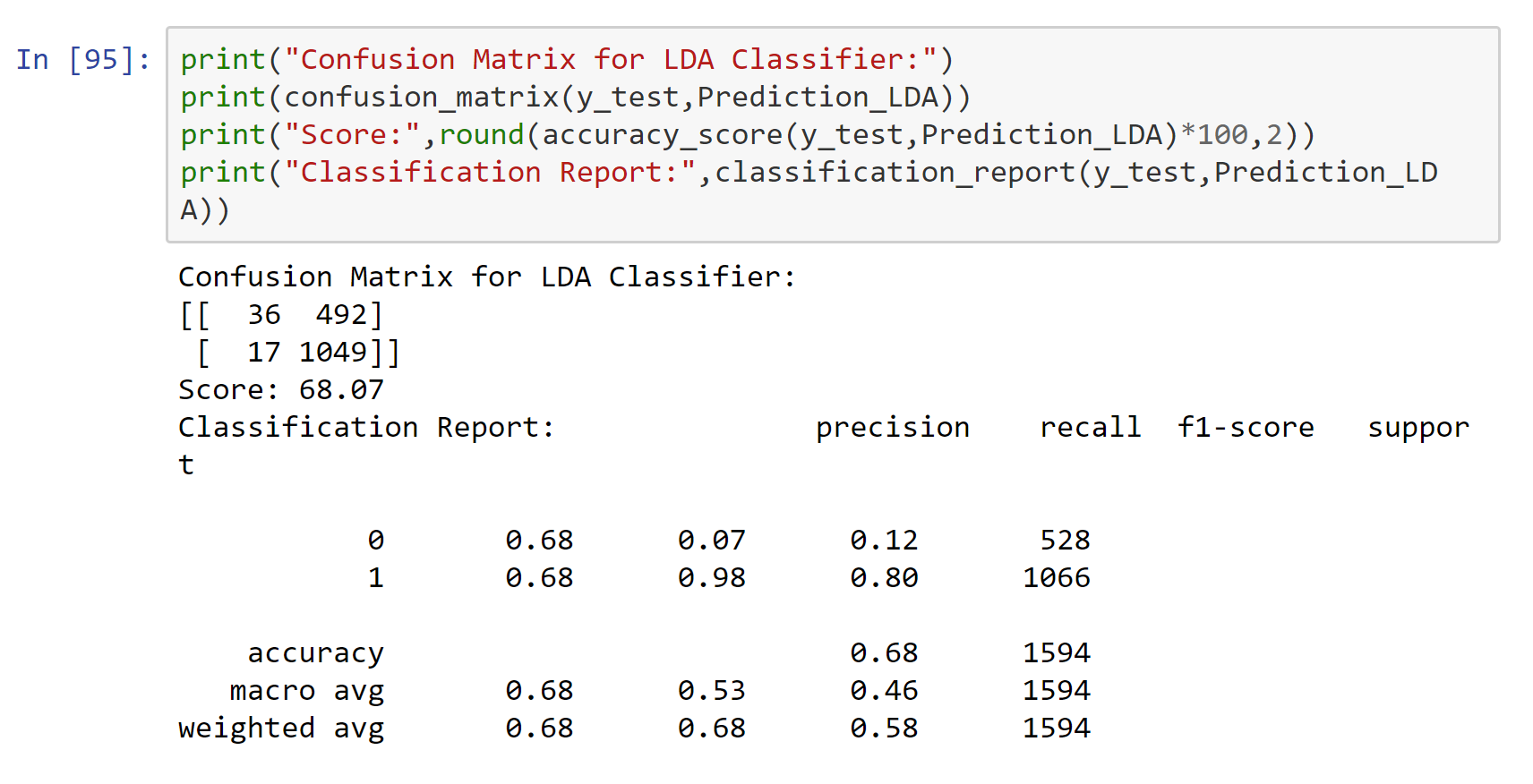
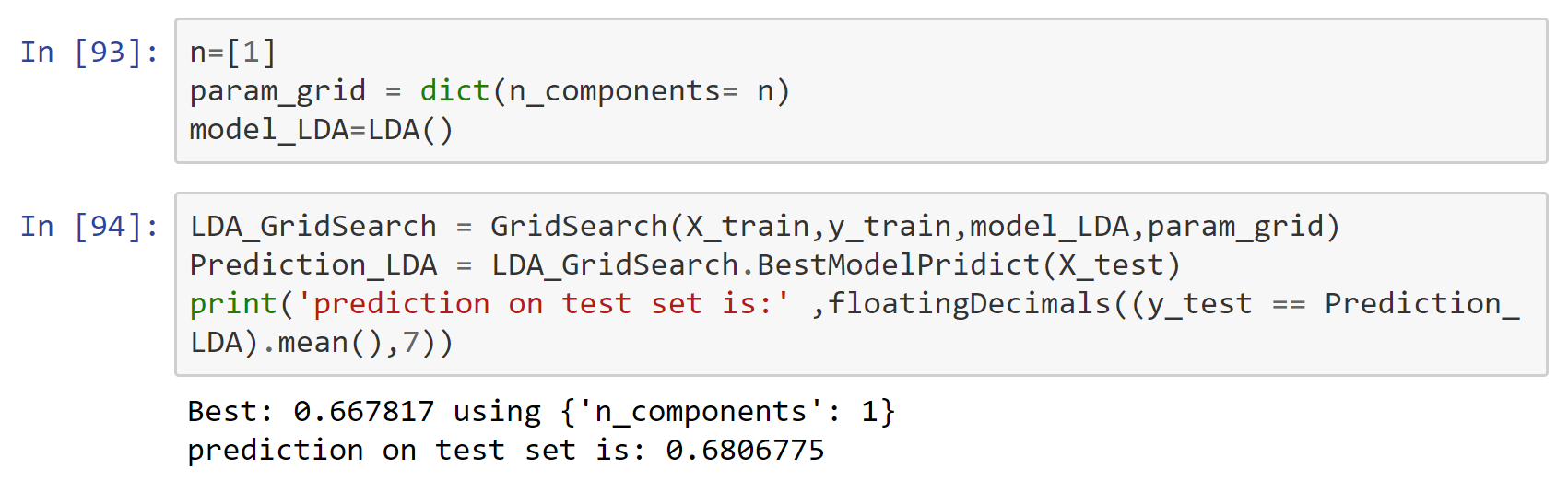
The Linear Discriminant Analysis is a topic model and is used to classify text to a particular topic. We thought this algorithm would work well with our reviews from Yelp, as it is text.

Pros

* Linear decision boundary
* Fast classification
* Easy to implement

Cons

* Training time is long
* Complex matrix
* Less accurate compared to QDA



# ***Candidate Algorithm Selection and Rationale***

Using the above information in regard to the algorithm evaluation, we can come to the conclusion that the top 3 algorithms are Random Forest, LDA and Logistic Regression respectively. Our target prediction is to determine whether a restaurant will close or not, hence we rely on the precision score. The more precise our predictions, the more confident we can be when providing our predictions to our clients. The weighted average of the precision metric for each were as follows:  
  
 **Random Forest : 0.69**

**LDA: 0.68**

**Logistic Regression: 0.67**

**Random Forest**: Compared to decision trees, the difference is minimal, but using Random Forest and removing the excess variance caused by Decision Tree we are given a more precise prediction.

**LDA**: Focused on text analysis, LDA is a strong candidate algorithm to use for our predictions. The algorithm was easy to implement, and the results were strong compared to the bottom 3.

**Logistic Regression**: Logistic Regression comes in as the bottom of the top 3 candidates with a precision result of 0.67. A strong reason to choose this is that the output is easy to understand and visualize.

Going forward, we will prioritize data scaling, feature selection and engineering to ensure that the we get better results from the selected candidate algorithms.