

Machine Learning Foundations

**CAPSTONE-PROJECT**  
**TEACHING-ASSISTANT-PERFORMANCE-EVALUATION**

BY

K M L Karunanayake

DSA\_0308

# Contents

1. Introduction..... 2

2. Data ..... 2

3. Methodology ..... 3

4. Results ..... 4

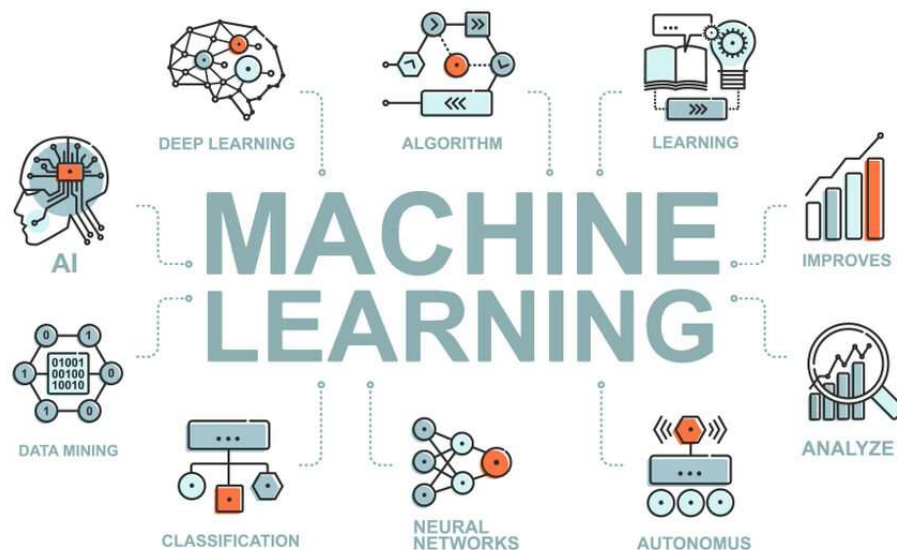
5. Conclusion ..... 5

6. Discussion ..... 5

## 1. Introduction

Machine learning is a popular technique to predict outputs of software applications using machine learning algorithm trained over past experience and historical data. Supervised learning, Unsupervised learning, Reinforcement learning are the main approaches of the machine learning algorithms. Regression, Classification, Clustering are the main three type of machine learning algorithms.

Building of a machine learning model to predict evaluation of the data set teaching assistant performance is the objective of this project. Supervised machine learning algorithm is built to make predictions from this classification problem.



<https://www.eurixgroup.com/eurix-machine-learning-2/>

## 2. Data

Teaching assistant performance data set is got from the UCI data repository. This data set consists five attributes of English Speaking, Course instructor, Course, Semester and Class size data consist of evaluations of teaching performance over three regular semesters and two summer semesters of 151 teaching assistant (TA) assignments at the Statistics Department of the University of Wisconsin-Madison. The scores were divided into three roughly equal-sized categories ("low", "medium", and "high") to form the Performance variable.

<b>Data Set Characteristics:</b>	Multivariate	<b>Number of Instances:</b>	151	<b>Area:</b>	N/A
<b>Attribute Characteristics:</b>	Categorical, Integer	<b>Number of Attributes:</b>	5	<b>Date Donated</b>	1997-06-07
<b>Associated Tasks:</b>	Classification	<b>Missing Values?</b>	No	<b>Number of Web Hits:</b>	193770

<https://archive.ics.uci.edu/ml/datasets/Teaching+Assistant+Evaluation>

## Attribute Information:

- Whether or not the TA is a native English speaker (binary); 1=English speaker, 2=non-English speaker
- Course instructor (categorical, 25 categories)
- Course (categorical, 26 categories)
- Summer or regular semester (binary) 1=Summer, 2=Regular
- Class size (numerical)
- Class attribute (categorical) 1=Low, 2=Medium, 3=High

### 3. Methodology

Supervised machine learning algorithm approach is used to make predictions from this classification problem. In this machine learning algorithm predictions are made using logistic regression and random forest classifier. Following steps are used in this solution to make machine learning model.

- Load data to Google Collab
- Pre-process Data for Training
- Data Pre-processing
- Train Test Split
- Model Training Function
- Saving Best Model
- Predict on a Sample Data
- Score Function
- Post-processing
- Inference Pipeline
- Get Prediction
- Made model server
- Load model server and test model serving API to Jupyter Notebook
- Test model serving API

In the Pre-process Data for Training, X variables and Y variables are selected from the teaching assistant performance data set. Adding new column of " id " using index and return selected X variables after correcting if there are mistakes when input data done in the data pre-processing. After training data and test data splits model training function returns model name, model, accuracy, precision and f1\_score to evaluate the model. Save Selected best model as a pickle and joblib files. And made predictions on a sample data. Score function returns probability of prediction on input data. Then post-processing and inference pipe line for the model. Get predictions of the input data. After successful pipelines made, made model server API and tested model serving API.

Below Tools used to build the machine learning model.



## 4. Results

### Model server API

```
(base) PS C:\Users\lahir\Desktop\MLFoundations> C:\Users\lahir\anaconda3\python C:\Users\lahir\Desktop\MLFoundations\model_server.py
C:\Users\lahir\anaconda3\lib\site-packages\sklearn\base.py:310: UserWarning: Trying to unpickle estimator DecisionTreeClassifier from version 1.0.2 when using version 0.24.2. This might lead to breaking code or invalid results. Use at your own risk.
  warnings.warn(
C:\Users\lahir\anaconda3\lib\site-packages\sklearn\base.py:310: UserWarning: Trying to unpickle estimator RandomForestClassifier from version 1.0.2 when using version 0.24.2. This might lead to breaking code or invalid results. Use at your own risk.
  warnings.warn(
* Serving Flask app "model_server" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with watchdog (windowsapi)
C:\Users\lahir\anaconda3\lib\site-packages\sklearn\base.py:310: UserWarning: Trying to unpickle estimator DecisionTreeClassifier from version 1.0.2 when using version 0.24.2. This might lead to breaking code or invalid results. Use at your own risk.
  warnings.warn(
C:\Users\lahir\anaconda3\lib\site-packages\sklearn\base.py:310: UserWarning: Trying to unpickle estimator RandomForestClassifier from version 1.0.2 when using version 0.24.2. This might lead to breaking code or invalid results. Use at your own risk.
  warnings.warn(
* Debugger is active!
* Debugger PIN: 699-707-837
* Running on http://127.0.0.1:5001/ (Press CTRL+C to quit)
```

Model serving API running on http://127.0.0.1:5001

### Result 1

```
In [11]: import requests

url = 'http://127.0.0.1:5001/classifier'

payload = {'English Speaking': 1,
           'Course instructor': 10,
           'Course': 21,
           'Semester': 2,
           'Class size': 29,}

headers = {}

files = [

]

response = requests.request("POST", url, headers=headers, data=payload, files=files)

print(response.text)

{
  "prediction": 0.442
}
```

According to the machine learning model output - prediction of 0.442

### Result 2

```
In [16]: import requests

url = 'http://127.0.0.1:5001/classifier'

payload = {'English Speaking': 0,
           'Course instructor': 3,
           'Course': 3,
           'Semester': 2,
           'Class size': 29,}

headers = {}

files = [

]

response = requests.request("POST", url, headers=headers, data=payload, files=files)

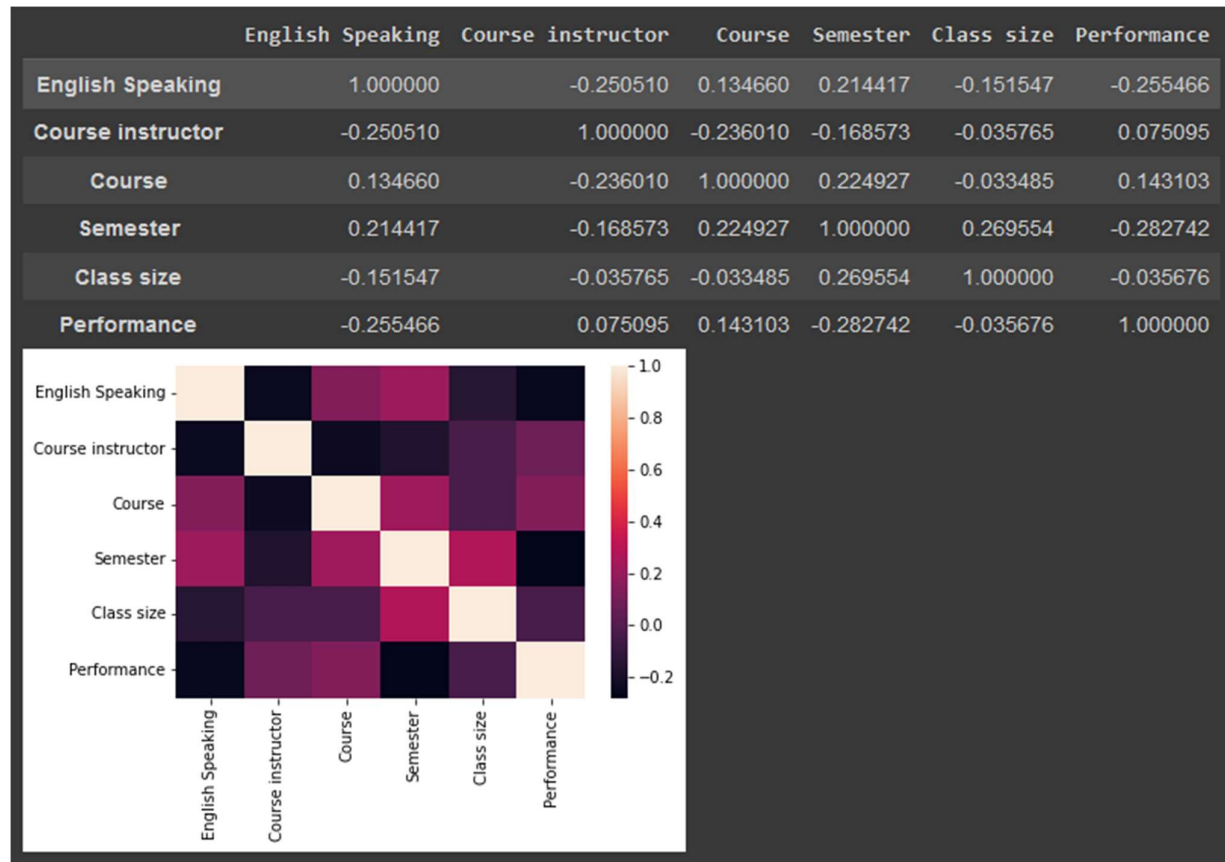
print(response.text)

{
  "prediction": 0.078
}
```

According to the Inputs machine learning model output prediction value of 0.078

## 5. Conclusion

Correlation matrix of the data set



Predictions of the machine learning model depends according to the training data. Relationship between X variables and the Y variable shows by the correlation matrix. According to the inputs predictions of the model changes as in the above “Result 1 ” and “Result 2 ”.

## 6. Discussion

Use of machine learning enable applications to make predictions of feature or classification accurate than traditional software algorithm. Historical data and past experience enhance the machine learning model's accuracy and perform tasks without explicit instructions. Machine learning is a subset of Artificial intelligence (AI) supported the thought that systems can learn from data, identify patterns and make decisions with minimal human intervention.