Machine Learning Project Report

Submitted to:

Dr. Yahia Zakaria

&

Eng. Mohamed Shawky Sabae

Team Members:

Name	Section	Bench
Karim Mahmoud Saqer	2	11
Karim Mohammed Abd-Elhameed	2	10
Mustafa Mahmoud Hamada	2	25
Mahmoud Reda Sayed	2	21

Contribution of each team member:

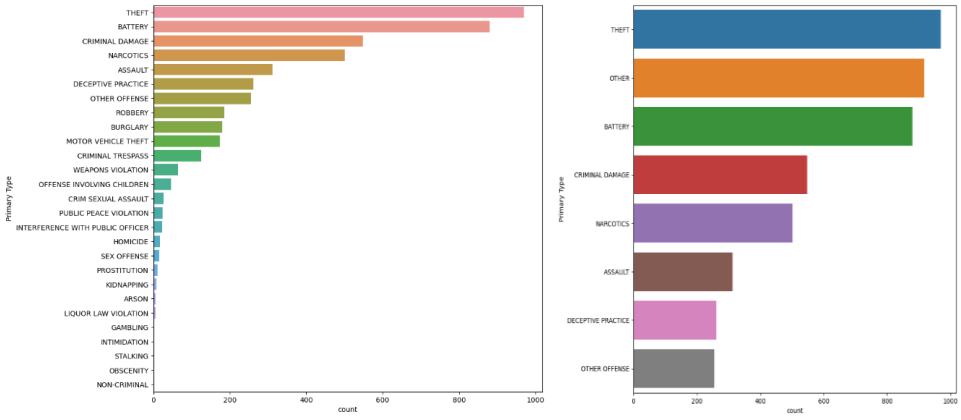
Name	Module
Mustafa Mahmoud & Mahmoud Reda	Data Analysis
Karim Mahmoud & Karim Mohammed	Machine Learning Models

Problem: Predict Crime Type

- **Problem Definition**: The problem involves predicting the type of crime based on various features such as location, time, and possibly other contextual information. Given a
  dataset containing historical records of crimes along with relevant attributes,
  the task is to build a machine learning model that can classify each instance
  into one of several predefined crime types.
- **Motivation**: Predicting the type of crime can assist law enforcement agencies in allocating
  resources more efficiently by focusing on areas or times where certain types of crimes are more likely to occur. This can lead to better crime prevention strategies, quicker response times, and ultimately safer communities.
- **Evaluation Metrics**: Accuracy – Precision – Recall – F1 Score - ....
- **Dataset link**:
  https://www.kaggle.com/code/heng8835/classification-with-ml-predict-crime-type/input

Results:

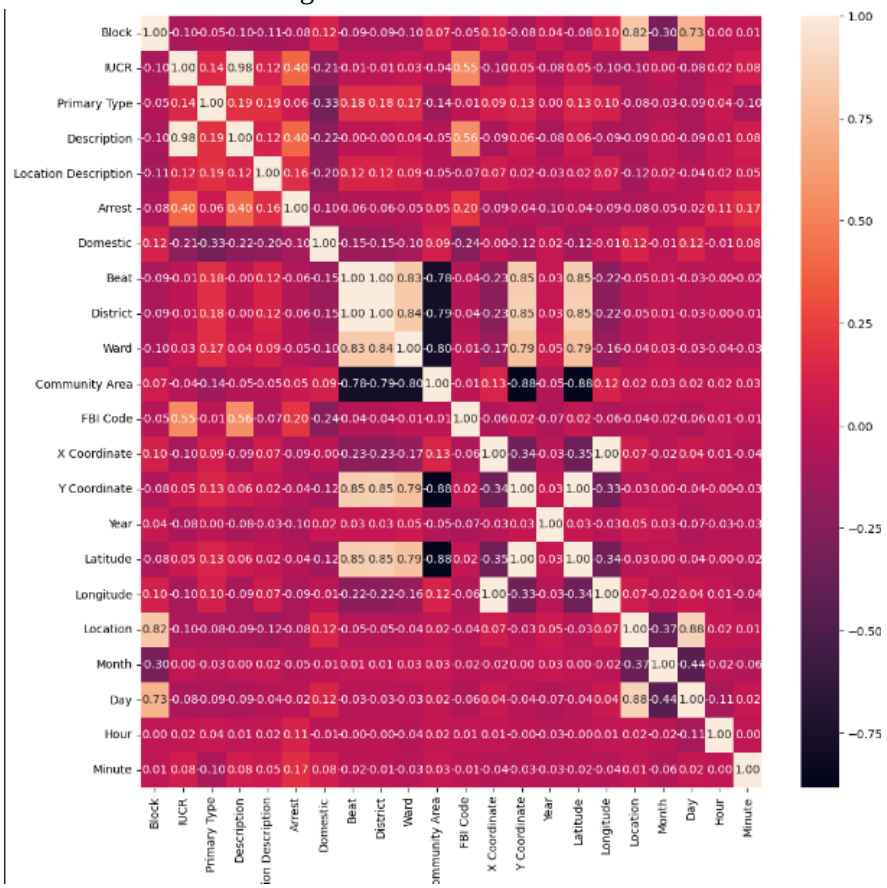
Initially, we loaded the dataset and proceeded to remove any rows containing missing or null data. Subsequently, we eliminated irrelevant columns such as ID and case numbers. Next, we assessed the number of classes to target and observed that there were numerous classes. To streamline the analysis, we set a threshold for less frequent classes and grouped them into a single class labeled as 'OTHER'.



Following this, we encoded the classes and converted categorical columns into numerical ones. Finally, we removed outliers from the dataset.

After cleaning the data, we generated a correlation matrix to examine the relationships between the features and target variables. Notably, we found no significant correlations, leading us to

retain all features unchanged.



Subsequently, we applied four different models: Support Vector Machine (SVM), Logistic Regression, Random Forest, and Perceptron. For each model, we conducted grid search to identify the optimal parameters. Among these models, Random Forest exhibited the highest accuracy.

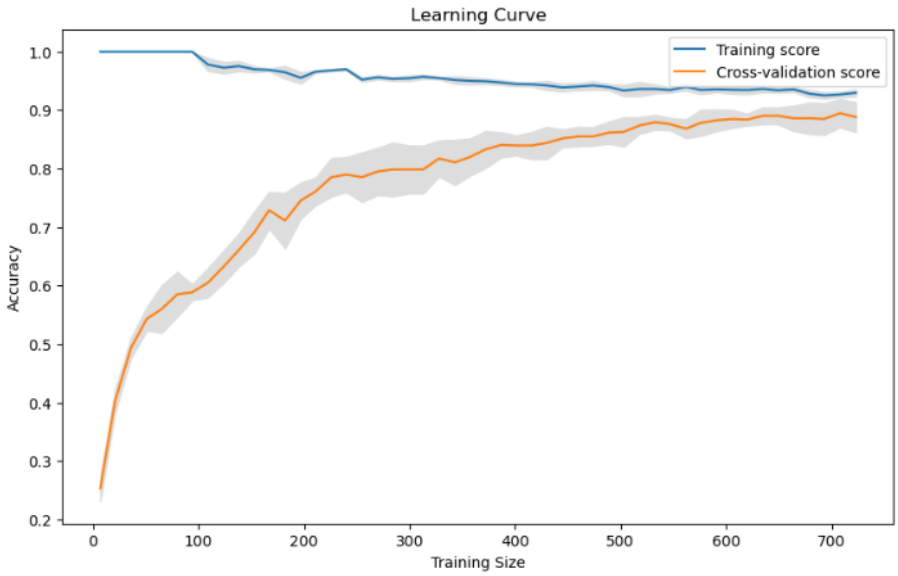
Experimental Results:

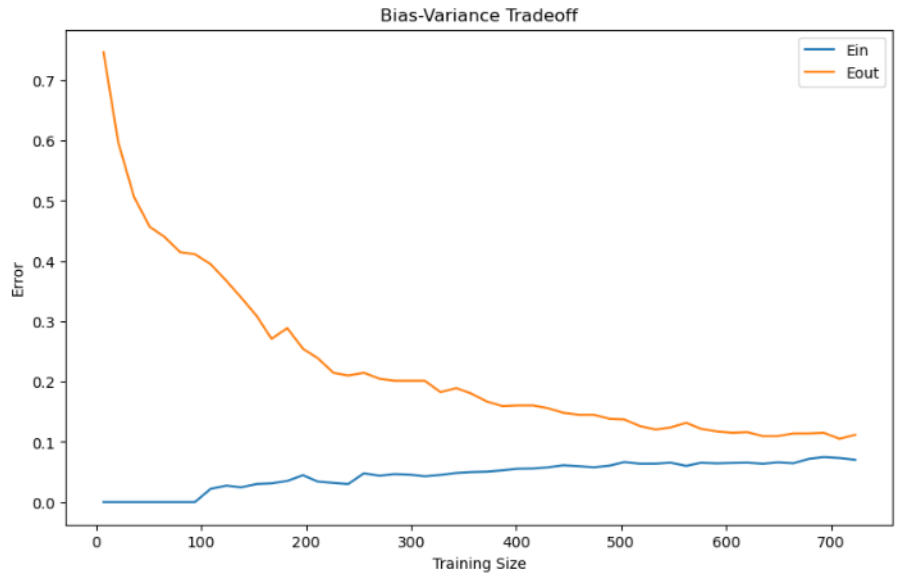
I will discuss each model in separate.

- **ZeroR (Baseline Model)**
  - ZeroR (most frequent class) serves as a baseline model to compare the performance of other classifiers.
  - Accuracy achieved by ZeroR is the simplest metric to evaluate performance.
  - It provides a benchmark against which the other models can be compared.
- **Support Vector Machine (SVM)**
  - SVM with a linear kernel was used with grid search for hyperparameter tuning (C, gamma and kernel)
    - C: Controls the tradeoff between maximizing the margin and minimizing the classification error. A higher value of C means the classifier will prioritize classification accuracy over the margin width
    - Kernel: Transform the input data into a higher-dimensional feature space where a linear boundary can be used to separate the data.
    - Gamma: It controls the shape of the decision boundary and how tightly the algorithm fits the data. A small gamma value means the decision boundary is relatively smooth.
  - The grid search helps to find the optimal combination of hyperparameters.
  - Precision, recall, and F1 score can provide insights into the performance of each class.
    -

	precision	recall	f1-score	support
0	1.00	0.91	0.95	11
1	1.00	0.91	0.96	47
2	0.94	1.00	0.97	30
3	0.75	0.60	0.67	5
4	1.00	1.00	1.00	14
5	0.72	0.94	0.81	47
6	1.00	0.33	0.50	15
7	0.91	0.91	0.91	58
accuracy			0.89	227
macro avg	0.92	0.83	0.85	227
weighted avg	0.91	0.89	0.88	227

  - Learning Analysis: this curve shows how the performance of a model improves with increasing amounts of training data. As can be seen the training and validation scores are converging at high values as the number of training examples increases and that means the model neither overfits nor underfitting the data, and it generalizes well to unseen data.





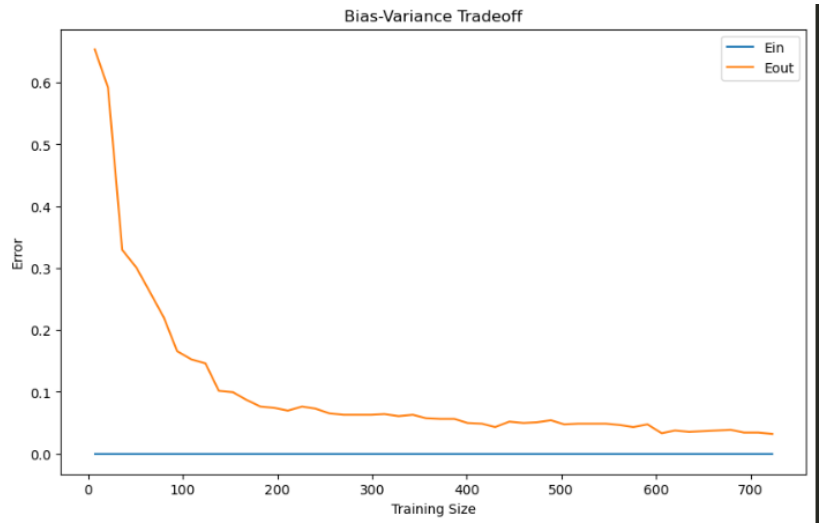
Conclusions:

- It is a complex model, characterized by rapid changes in Eout and achiving final small Eout.
- Initially, Eout is significantly higher for the complex model due to the small number of points, resulting in large variance.

➢ **<u>Random Forest Classifier</u>**

- Random Forest Classifier was utilized with grid search to tune the number of estimators and the maximum depth of trees.
- The parameters of the best model indicate the optimal configuration for achieving high accuracy.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	0.98	1.00	0.99	47
2	0.97	1.00	0.98	30
3	1.00	0.80	0.89	5
4	0.93	1.00	0.97	14
5	0.94	0.98	0.96	47
6	0.92	0.73	0.81	15
7	1.00	0.98	0.99	58
accuracy			0.97	227
macro avg	0.97	0.94	0.95	227
weighted avg	0.97	0.97	0.97	227

- 

Conclusions:

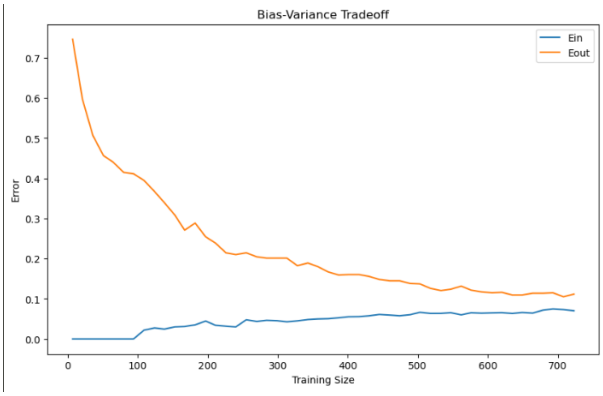
- The model is complex.
- Initially, Eout is significantly higher for the complex model due to the small number of points, resulting in large variance.
- Ein starts at a low point as the complex model can easily find a hypothesis that fits the small sample.
- Eout and and Ein become the same as $N \rightarrow \infty$ which is true by Hoeffding's inequality.
- With $N \rightarrow \infty$ and variance (var) diminishing, the final Eout becomes very low, containing only the bias component.

➢ **<u>Logistic Regression</u>**

- Logistic Regression was employed with grid search to optimize (C, Solver, Penalty).
  - Penalty: determines the type of regularization to be used in the logistic regression model (L1 / L2)
  - C: controls the strength of the regularization in the model. A smaller value of C results in stronger regularization

    - Solver: determines the algorithm to be used for optimization in the logistic regression model
- Classification report provides a detailed breakdown of precision, recall, and F1 score for each class.

	precision	recall	f1-score	support
0	1.00	0.91	0.95	11
1	0.93	0.85	0.89	47
2	0.86	1.00	0.92	30
3	0.67	0.40	0.50	5
4	0.93	1.00	0.97	14
5	0.73	0.79	0.76	47
6	0.67	0.40	0.50	15
7	0.87	0.91	0.89	58
accuracy			0.85	227
macro avg	0.83	0.78	0.80	227
weighted avg	0.84	0.85	0.84	227



Conclusions:

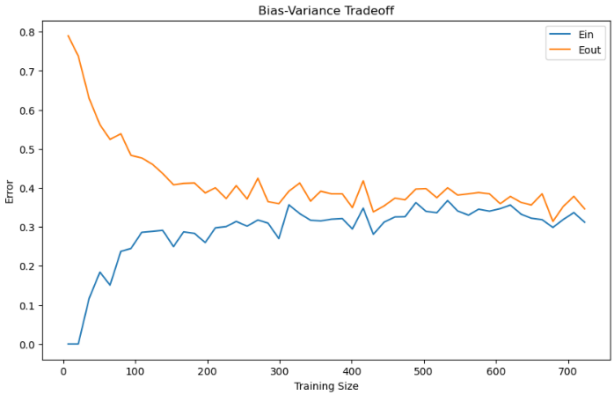
- It is a complex model, as Eout changes rapidly and the final Eout is small
- Eout decreases as the variance decreases and Ein increases as it becomes harder to fit the data points

- Initially, Eout is significantly higher for the complex model due to the small number of points, resulting in large variance.

➢ **Perceptron**

- Perceptron was trained with grid search to find the best values for regularization parameter (alpha), maximum iterations, and learning rate.
- Insights from the classification report help in understanding the performance across different classes.

	precision	recall	f1-score	support
0	0.71	0.91	0.80	11
1	0.80	0.68	0.74	47
2	0.85	0.73	0.79	30
3	0.14	0.40	0.21	5
4	0.88	1.00	0.93	14
5	0.56	0.57	0.57	47
6	0.38	0.40	0.39	15
7	0.60	0.55	0.58	58
accuracy			0.64	227
macro avg	0.61	0.66	0.62	227
weighted avg	0.66	0.64	0.65	227

- 

Conclusions:

- Simple Model
- Eout decreases as the variance decreases and Ein increases as it becomes harder to fit the data points

Overall Analysis

- Random Forest Classifier appears to perform the best among the tested models, achieving the highest accuracy.
- However, it's essential to consider other metrics such as precision, recall, and F1 score to gain a comprehensive understanding of model performance.
- Interpretation of model parameters can provide insights into the importance of features and their impact on model performance.

Evaluation

Model	Accuracy	Weighted F1	Weighted precision	Weighted recall
SVM	88.38%	0.89	0.91	0.88
Logistic Regression	84.63%	0.84	0.84	0.85
Random Forest	**97.1%**	**0.97**	**0.97**	**0.97**
Perceptron	65.56%	0.58	0.58	0.58