Report

By:-

- Mohamed Khaled Ismail - 7000463
- Yahia Ehab - 7037125
- Mariam Amr elDeeb - 7001053

Steps Done for Each Dataset:-

1. Importing the needed imports:-

   - pandas
   - numpy
   - StandardScaler
   - LabelEncoder
   - RandomForestClassifier
   - AdaBoostClassifier
   - GradientBoostingClassifier
   - GridSearchCV
   - RandomizedSearchCV
   - accuracy_score
   - precision_score
   - recall_score
   - f1_score
   - matplotlib.pyplot
   - seaborn

2. Dataset selection is done for each one so that we can work on each dataset seperately.

3. Then preprocessing is done:-

   1. Dataset then has no nulls
   2. Choosing only columns that we will use
   3. Scaling of the numerical features
   4. Encoding the categorical features
   5. Removing outliers using `IQR` method

4. Then the model is impleneted and the hyperparameters are then tuned for better model performance.

   1. Train, test and aplit the data using the predifined method.

   2. Apply models and hyperparameter tuning.

      1. **Hyperparameters**

         - We chose a range upwards of 2 parameters to experiment with and tune with some values that are given manually and see what values are best for the parameters we experiment with.

      2. **Random Forest**

         - Random Search and/or Grid Search
         - Get the model using the best parameters

3. **AdaBoost**

   - Random Search and/or Grid Search
   - Get the model using the best parameters

4. **Gradient Boost**

   - Random Search and/or Grid Search
   - Get the model using the best parameters

3. Final Comparison between models throght the:-

```
from sklearn.metrics import accuracy_score, precision_score,
recall_score, f1_score
```

Visualizations & Results

1. **Bankloan Dataset - Yahia Ehab**:-

   1. **Random Forest**

      - When using the *default* Random Forest classifier, these are the results:-
        - *Accuracy: 0.989*
        - *Precision: 0.9795918367346939*
        - *Recall: 0.9142857142857143*
        - *F1 Score: 0.9458128078817734*
      - When trying to tune the parameters with Randomized Search, these were the best parameter values:-
        - *n_estimators*: 200
        - *max_features*: sqrt
        - *max_depth*: None
      - When trying to tune the parameters with Grid Search, these were the best parameter values:-
        - *n_estimators*: 300
        - *max_features*: sqrt
        - *max_depth*: None
      - When using the *tuned* Random Forest classifier, these are the results:-
        - *Accuracy: 0.989*
        - *Precision: 0.9795918367346939*
        - *Recall: 0.9142857142857143*
        - *F1 Score: 0.9458128078817734*

   2. **AdaBoost**

      - When using the *default* AdaBoost classifier, these were the results:-
        - *Accuracy: 0.971*
        - *Precision: 0.8958333333333334*
        - *Recall: 0.819047619047619*
        - *F1 Score: 0.855721393034826*

- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
  - *n_estimators = 100*
  - *learning_rate = 1*

- When trying to tune the parameters with Grid Search, these were the best parameter values:-
  - *n_estimators = 100*
  - *learning_rate = 1*

- When using the *tuned* Random Forest classifier, these are the results:-
  - *Accuracy: 0.971*
  - *Precision: 0.8958333333333334*
  - *Recall: 0.819047619047619*
 - *F1 Score: 0.855721393034826*

3. **Gradient Boost**

- When using the *default* Gradient Boost classifier, these were the results:-
  - *Accuracy: 0.992*
  - *Precision: 0.98989898989899*
  - *Recall: 0.9333333333333333*
  - *F1 Score: 0.9607843137254903*

- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
  - *n_estimators = 150*
  - *max_depth = 3*
  - *learning_rate = 0.1*

- When trying to tune the parameters with Grid Search, these were the best parameter values:-
  - *n_estimators = 150*
  - *max_depth = 7*
  - *learning_rate = 0.1*

- When using the *tuned* Random Forest classifier, these are the results:-
  - *Accuracy: 0.992*
  - *Precision: 0.98989898989899*
  - *Recall: 0.9333333333333333*
  - *F1 Score: 0.9607843137254903*

4. **Final Comparison**

- Table of comparisons:-

	Model	Accuracy	Precision	Recall	F1 Score
0	Random Forest Classifier	0.989	0.979592	0.914286	0.945813
1	Random Forest Classifier with best hyperparame...	0.989	0.979592	0.914286	0.945813
2	AdaBoost Classifier	0.971	0.895833	0.819048	0.855721
3	AdaBoost Classifier with best hyperparameters	0.970	0.886598	0.819048	0.851485
4	Gradient Boosting Classifier	0.992	0.989899	0.933333	0.960784
5	Gradient Boosting Classifier with best hyperpa...	0.992	0.989899	0.933333	0.960784

- Bar chart of the results:-

BankLoan Dataset Model Comparison

---
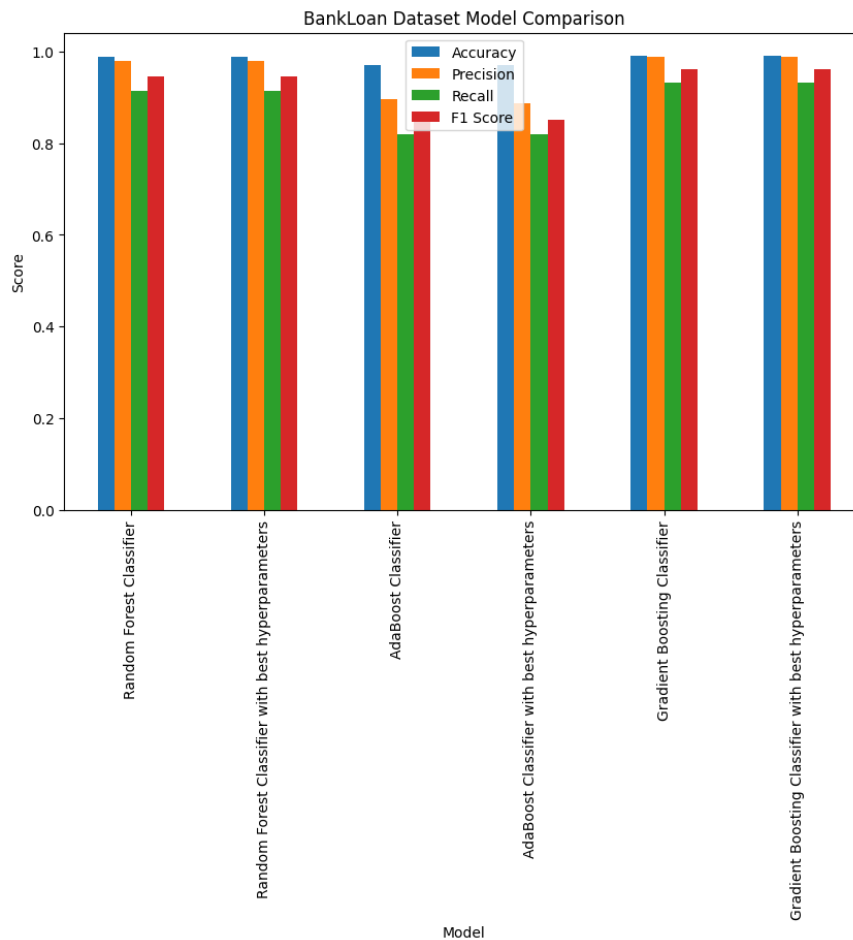
2. **Banknote Dataset - Mohamed Ismail**

   1. **Random Forest**

      - When using the *default* Random Forest classifier, these are the results:-
        - *Accuracy: 0.9927272727272727*
        - *Precision: 1.0*
        - *Recall: 0.984251968503937*
        - *F1 Score: 0.9920634920634921*
      - When trying to tune the parameters with Grid Search, these were the best parameter values:-
        - *bootstrap: True*
        - *max_features: sqrt*
        - *n_estimators: 200*
      - When using the *tuned* Random Forest classifier, these are the results:-
        - *Accuracy: 0.989*
        - *Precision: 1.0*

- Recall: 0.984251968503937
- F1 Score: 0.9920634920634921

2. **AdaBoost**

- When using the *default* AdaBoost classifier, these were the results:-
  - Accuracy: 0.9963636363636363
  - Precision: 1.0
  - Recall: 0.9921259842519685
  - F1 Score: 0.9960474308300395
- When trying to tune the parameters with Grid Search, these were the best parameter values:-
  - algorithm = SAMME
  - learning_rate = 0.5
  - n_estimators = 500
  - random_stat = None
- When using the *tuned* Random Forest classifier, these are the results:-
  - Accuracy: 0.9963636363636363
  - Precision: 1.0
  - Recall: 0.9921259842519685
 - F1 Score: 0.9960474308300395

3. **Gradient Boost**

- When using the *default* Gradient Boost classifier, these were the results:-
  - Accuracy: 1.0
  - Precision: 1.0
  - Recall: 1.0
  - F1 Score: 1.0
- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
  - learning_rate = 2.0
  - loss = exponential
  - n_estimators = 64
- When trying to tune the parameters with Grid Search, these were the best parameter values:-
  - n_estimators = 150
  - max_depth = 7
  - learning_rate = 0.1
- When using the *tuned* Random Forest classifier, these are the results:-
  - Accuracy: 1.0
  - Precision: 1.0
  - Recall: 1.0
  - F1 Score: 1.0

4. **Final Comparison**

- Table of comparisons:-

	Model	Accuracy	Precision	Recall	F1 Score
0	Random Forest Classifier	0.992727	1.0	0.984252	0.992063
1	Random Forest Classifier with best hyperparame...	0.992727	1.0	0.984252	0.992063
2	AdaBoost Classifier	0.996364	1.0	0.992126	0.996047
3	AdaBoost Classifier with best hyperparameters	0.996364	1.0	0.992126	0.996047
4	Gradient Boosting Classifier	1.000000	1.0	1.000000	1.000000
5	Gradient Boosting Classifier with best hyperpa...	1.000000	1.0	1.000000	1.000000

- Bar chart of results:-

BankLoan Dataset Model Comparison

---
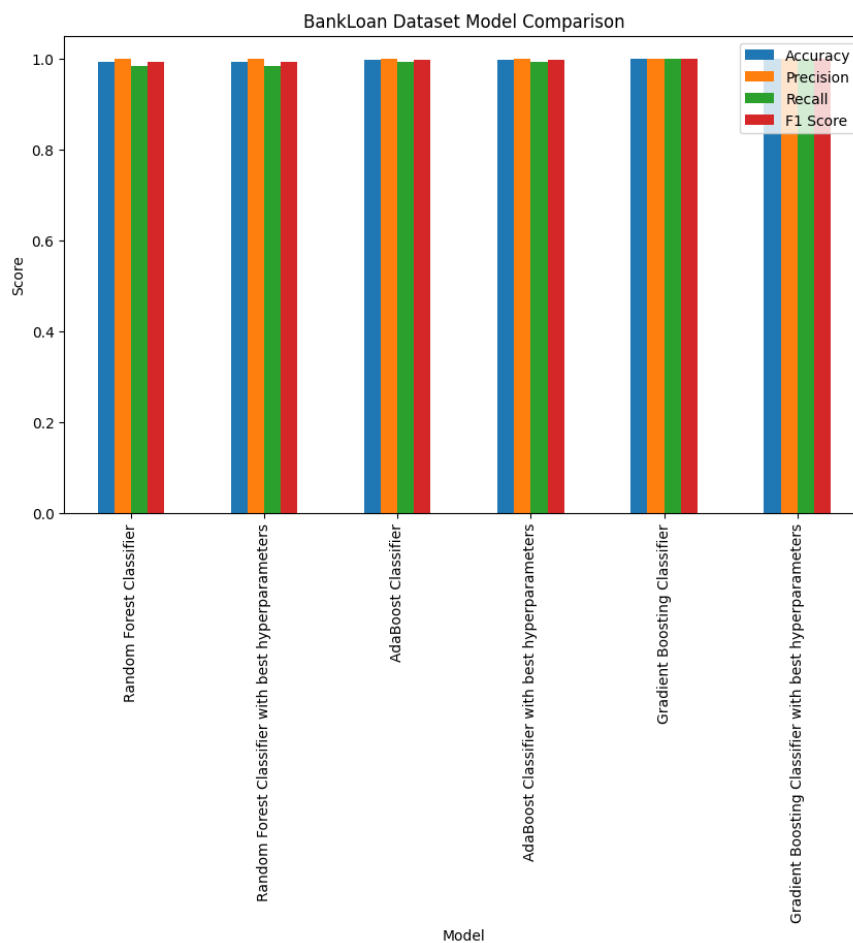
3. GlassType Dataset - Mariam Amr

   1. **Random Forest**

      - When using the *default* Random Forest classifier, these are the results:-
        - *Accuracy: 0.8372093023255814*
        - *Precision: 0.9127314814814816*
        - *Recall: 0.8432539682539683*
        - *F1 Score: 0.8605223570909845*

- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
    - *n_estimators: 200*
    - *min_samples_split = 2*
    - *min_samples_leaf = 2*
    - *max_features: sqrt*
    - *max_depth = 10*
    - *criterion = entropy*
- When trying to tune the parameters with Grid Search, these were the best parameter values:-
    - *n_estimators: 150*
    - *min_samples_split = 5*
    - *min_samples_leaf = 2*
    - *max_features: sqrt*
    - *max_depth = 10*
    - *criterion = gini*
- When using the *tuned* Random Forest classifier, these are the results:-
    - *Accuracy: 0.8604651162790697*
    - *Precision: 0.9299145299145298*
    - *Recall: 0.8253968253968255*
    - *F1 Score: 0.8546058879392212*

2. **AdaBoost**

- When using the *default* AdaBoost classifier, these were the results:-
    - *Accuracy: 0.4883720930232558*
    - *Precision: 0.7671568627450981*
    - *Recall: 0.43127705627705626*
    - *F1 Score: 0.3815668202764977*
- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
    - *n_estimators: 150*
    - *learining_rate = 0.1*
- When trying to tune the parameters with Grid Search, these were the best parameter values:-
    - *n_estimators: 150*
    - *learining_rate = 0.1*
- When using the *tuned* Random Forest classifier, these are the results:-
    - *Accuracy: 0.6976744186046512*
    - *Precision: 0.7705415499533146*
    - *Recall: 0.6006493506493507*
    - *F1 Score: 0.5954224270353302*

3. **Gradient Boost**

- When using the *default* Gradient Boost classifier, these were the results:-
    - *Accuracy: 0.8604651162790697*

- *Precision: 0.8902116402116403*
- *Recall: 0.8115079365079364*
- *F1 Score: 0.813602671837966*

- When trying to tune the parameters with Randomized Search, these were the best parameter values:-
  - *n_estimators = 100*
  - *min_samples_split = 2*
  - *min_samples_leaf = 2*
  - *max_features = sqrt*
  - *max_depth = 5*
  - *learning_rate = 0.1*

- When trying to tune the parameters with Grid Search, these were the best parameter values:-
  - *n_estimators = 150*
  - *min_samples_split = 2*
  - *min_samples_leaf = 2*
  - *max_features = sqrt*
  - *max_depth = 5*
  - *learning_rate = 0.1*

- When using the *tuned* Random Forest classifier, these are the results:-
  - *Accuracy: 0.8837209302325582*
  - *Precision: 0.9386446886446885*
  - *Recall: 0.8809523809523809*
  - *F1 Score: 0.8935802469135803*

4. **Final Comparison**

  - Table of comparisons:-

	Model	Accuracy	Precision	Recall	F1 Score
0	Random Forest Classifier	0.837209	0.912731	0.843254	0.860522
1	Random Forest Classifier with best hyperparame...	0.860465	0.929915	0.825397	0.854606
2	AdaBoost Classifier	0.488372	0.767157	0.431277	0.381567
3	AdaBoost Classifier with best hyperparameters	0.697674	0.770542	0.600649	0.595422
4	Gradient Boosting Classifier	0.860465	0.890212	0.811508	0.813603
5	Gradient Boosting Classifier with best hyperpa...	0.883721	0.938645	0.880952	0.893580

▪ Bar chart of results:-

Glass Type Prediction Model Comparison