SUMMARY AND BRIEF REPORT

A Comparative Study on Predicting High-School Student Performance Using Neural Networks and SVM

Learning From Data Course (DS341)

Assistant Professor: Dr. Elshimaa Elgendi
Teacher Assistant: Eng. Ahmed Fouad

By

Abdullah Elafifi: 20221238

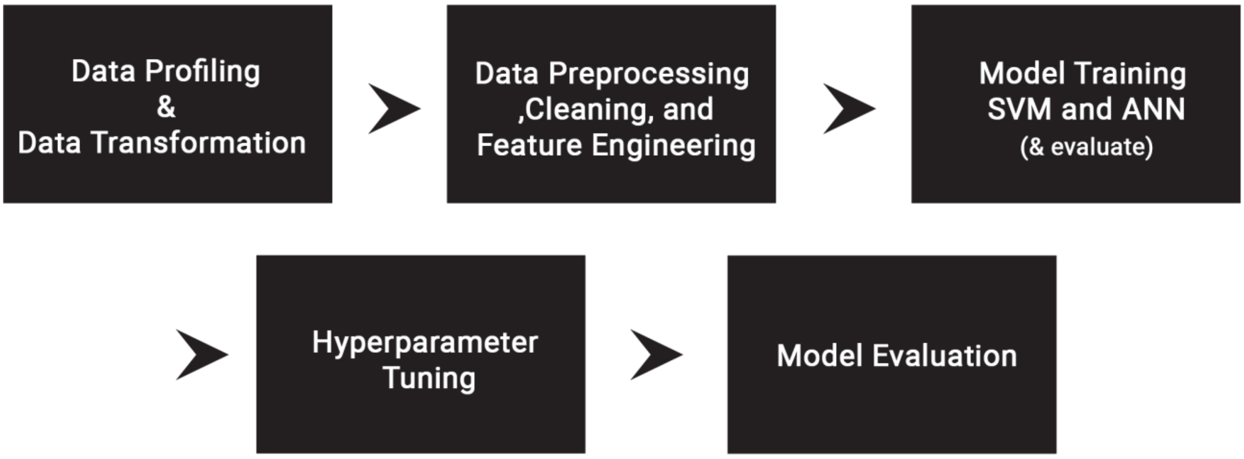
Mostafa Mahmoud: 20200549

Youssef Khaled: 20221244

Osama Husseny: 20221224

Mohammed Hassan: 20221239

The project Pipeline:



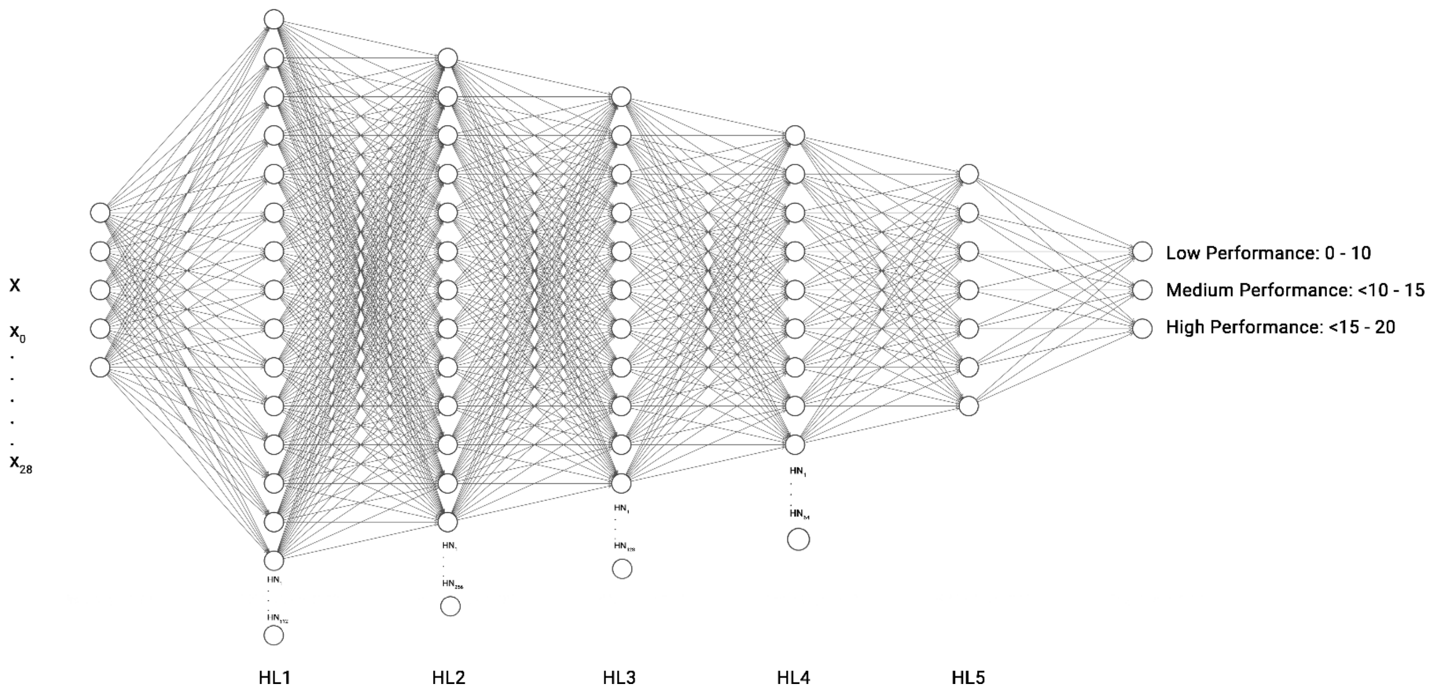
Technological Tools:

- **Development Environment:** Google Colab
- **Programming Language:** Python
- **Packages and Libraries:**
  - **For Data Manipulation and cleaning:** Numpy, Pandas, ydata profiling, sklearn with its sub libraries.
  - **Data Visualization:** Matplotlib and Seaborn.
  - **Neural Network Development:** Pytorch (Torch), sklearn and its swith its sub libraries.
  - **Support Vector Machine Algorithm:** SVM of sklearn.

Sources for enrichment:

We reached out to different resources including **Medium** articles, **GeeksforGeeks,** and some technical articles on **Reddit.**

The Artificial Neural Network Architecture:



Brief Technical Report:

We are chosen to work in bands of ranges as a classification problem. These bands are separated in 3 ranges that represented in **Low Performance [0,10[, Medium Performance [10,15[and High Performance [15,20].** These ranges are based on the 28 features (including G1, G2) and make the target is the final grade (G3) and follow our categorized guideline:

- Bin 1: Values from 0 to 10
- Bin 2: Values from 10 to 15
- Bin 3: Values from 15 to 20

The values in G3 will be categorized based on which range they fall into and follow the labels in the final new **Performance** Column.

- (0, 10) will be labeled 0
- (10, 15) will be labeled 1
- (15, 20) will be labeled 2

Moreover, we a lot of data preprocessing steps in order to make the input more balanced and less correlated, you can take a look at
Data Preprocessing Notebook

Summary of the two models:

First of all, based on the experts and based on the dataset we are looked at, the SVM will be suitable for this mission; due to the less counted entries (records) of the dataset and have the C parameter that control the overfitting, which in this case is more likely to happened.

After building the models and follow the project's guidelines of evaluating the performance, we can simply say that **SVM** is more suitable for classification of the project's problem, the coming lines will have a more technical explanation for why we decided to use **SVM** in the future with similar problems.

Model Evaluation:

The evaluation metrics shows that the **SVM** is better in **Accuracy** which is the overall correctness of the model by calculating the percentage of correct predictions out of all predictions made; gets a percentage of **87.34,** comparison with the result of the ANN gets a percentage of **83.50.**

Precision which is the percentage of true positive predictions out of all positive predictions made by the model; gets a percentage of **87.90,** comparison with the result of the ANN gets a percentage of **85.06.**

Recall which is the percentage of actual positives that the model successfully identified; gets a percentage of **87.34,** comparison with the result of the ANN gets a percentage of **83.50.**

F1 -Score which is the mean of Precision and Recall, providing a single score that balances the two; gets a percentage of **87.34,** comparison with the result of the ANN gets a percentage of **83.64.**

Moreover, the Area Under Curve (AUC) and Receiver operating characteristics (ROC) analysis in both models which they got a very great results but again, the SVM is more slightly better in this case.

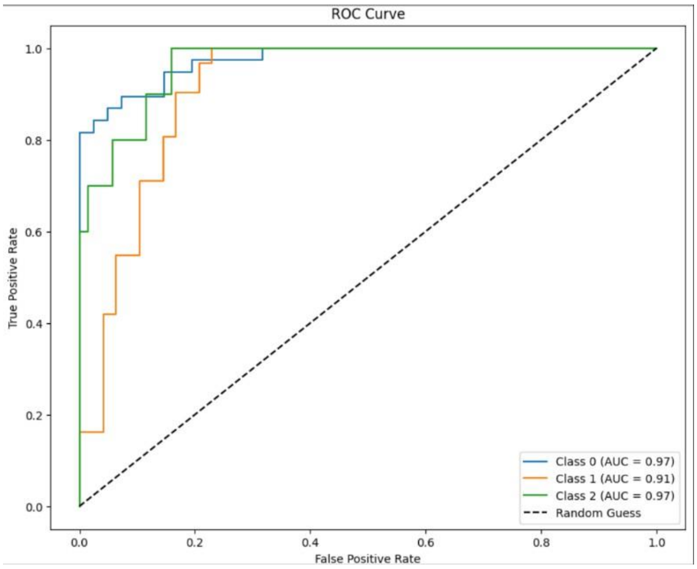


Fig.1 The ROC curve for the SVM model

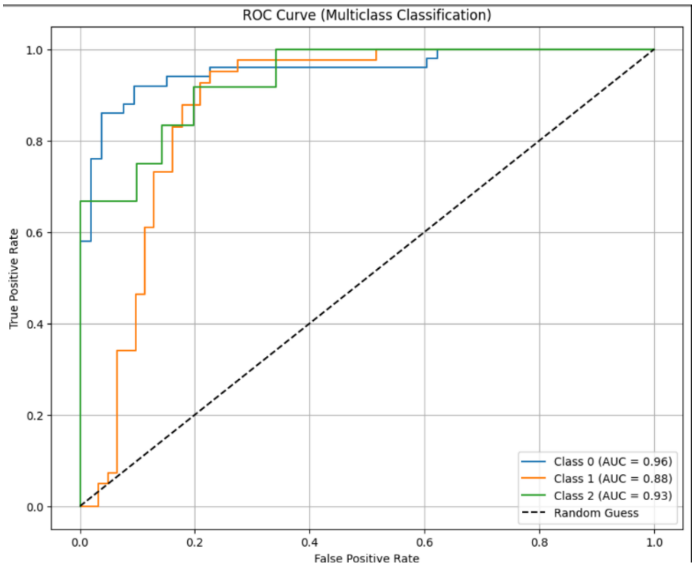


Fig.1 The ROC curve for the ANN model

***NOTE: there is a problem in the key results for both models**

In addition, the error for each epoch in the 5 required epochs shows a slightly better results for SVM Model, which has a **train error of 9.9,** and **test error of 16.5.**

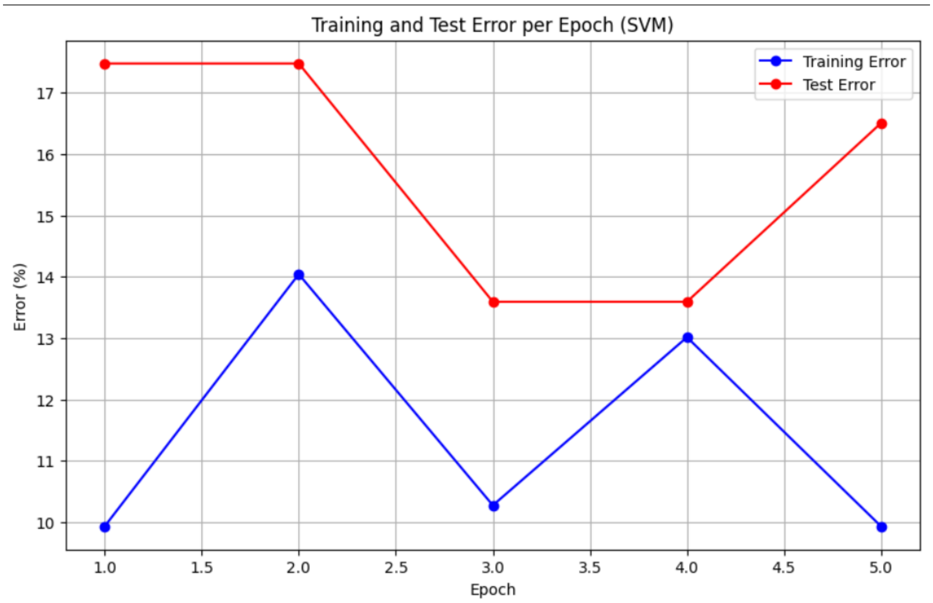


Fig.3 The train and test error per epoch for the SVM model

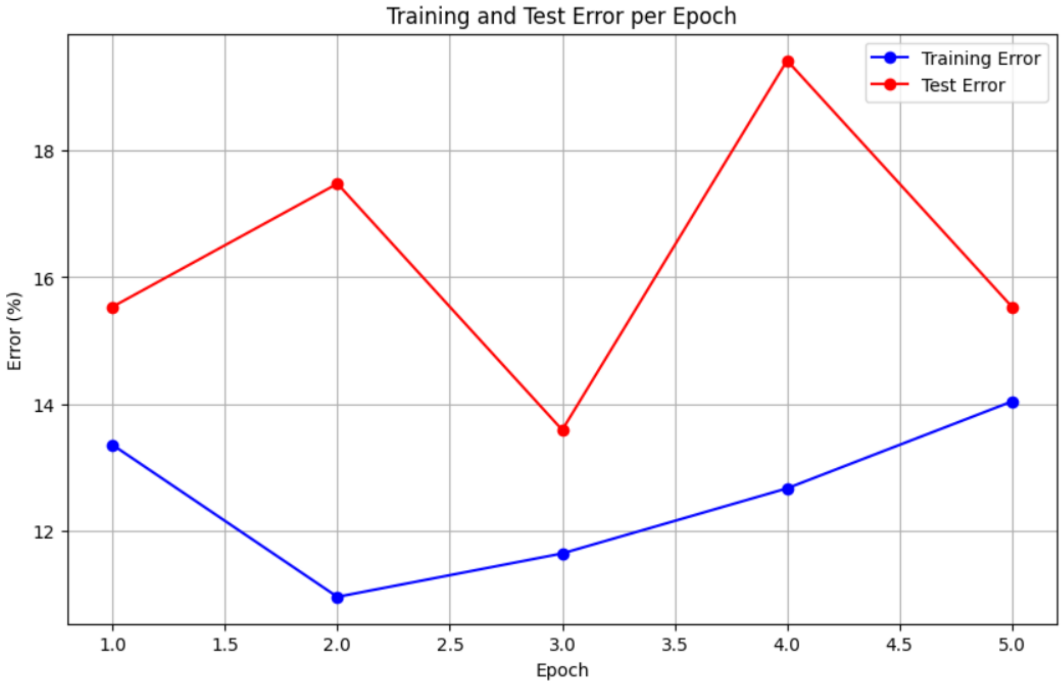


Fig.4 The train and test error per epoch for the ANN model

Conclusion and Future Plan:

The two models perform very well, but the scales are tilted towards the SVM model using the Radial Basis Function (RBF) kernel based on the clear appearance of non-linearity of the data, we can say that the result can be enhanced but we have a restriction in the project guidelines.

For a similar project problem in the future, we are suggesting taking into consideration some essential points:

- Increasing the number of epochs more than the 5 restricted epochs, we suggest that use 50 epochs or more (we do that in the ANN and get accuracy of 98%) with no overfitting.
- Increasing the number of hidden layers will get better results than increasing the hidden neurons with a fixed number of hidden layers.
- Try with polynomial kernel of SVM may it has improved results.
- Increasing the usage of hyperparameters tuning will have improved results for sure as well as shuffle your data at the beginning of the model’s training.

Acknowledgement:

We would like to express our sincere gratitude to **Professor Elshimaa Elgendi** for her invaluable guidance and support throughout this semester. Your expertise, constructive feedback, and encouragement have greatly contributed to the success of this work.

We would also like to extend our appreciation to Eng. Ahmed Fouad, for his continuous assistance, insightful suggestions, and patience in helping us overcome challenges during the course and labs.

Their collective efforts have been crucial in shaping the direction and quality of this work, and we are deeply grateful for their support.

Written by the team members,

FINISHED

***All visuals are used here are from our creation**