Neural network-based time series forecasting of student enrollment with exponential smoothing baseline and statistical analysis of performance.

by

Friday James

B.S., University of Agriculture, Makurdi – Nigeria, 2009
M.S., University of Ilorin, Ilorin – Nigeria, 2015

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computer Science
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2021

Approved by:

Major Professor
Dr. Joshua Levi Weese

Copyright

Abstract

The sustainability of educational institutions generally depends largely on strategic planning, both in terms of optimal allocation of resources/manpower and budgeting for financial aids/scholarships to incoming students. Hence, forecasting of student enrollment plays a vital role in making crucial decisions based on previous time-bound records. This work demonstrates the power of neural network-based time series forecast over a traditional time series model and recommends the better network architecture between deep and shallow neural networks based on 25-year historical records of student enrollment in Programming Fundamentals from 1995 – 2020 at Kansas State University, Manhattan Campus. The study reveals that Vanilla Long Short-Term Memory (LSTM) model performs better than the deep neural network with Root Mean Square Errors (RMSE) of 0.11 and 0.24 respectively – both of which produced better results than the Single Exponential Smoothing baseline having a RMSE of 0.27. The study also carries out a statistical analysis of 5-year student performance based on weekly Labs, Projects and Mid-Terms using Analysis of Variance (ANOVA). The result shows the existence of differences in the yearly average performance of students. Post Hoc Tukey's pairwise multiple comparison tests reveals consistency in performance up to the period of the semester where possible dropouts would have occurred. Students' delay in tackling challenging projects also accounts for the significant differences in the mean scores.

Table of Contents

List of Figures

List of Tables

List of Equations

Acknowledgements

My profound gratitude goes to my major Dr. Josh Weese for his support and for always being available to guide me amidst his busy schedules. I also appreciate the support of my committee members Dr. William Hsu and Dr. Doina Caragea for their immense contributions to this work.

Dedication

I dedicate this work to my late Uncle, Mr. Sunday J. Ali through whose effort I am where I am today – having brought me up as a kid and never gave up on my dreams. I also dedicate this work to my friends and well-wishers Mr. Kehinde Osho and his wife Mrs. Abiola Osho as well as Adedolapo Okanlawon for their encouragement and support. Finally, I also dedicate this work to the all members of the Cyber Physical Systems Lab and the Lab Head, Dr. Mitchell Nielsen who gave me a befitting space for my study and research.

CHAPTER ONE - INTRODUCTION

Time plays an important role in decision making in every human endeavor – from personal to business/management decisions. In these, the education sector is not left out. Hence, it is note-worthy that strategic decisions are a function of time. The sustainability of educational institutions generally depends largely on strategic planning both in terms of optimal allocation of resources/manpower and budgeting for financial aids/scholarships to incoming students. Similarly, it is imperative that the institutions are furnished with information that would lead to improved teaching techniques and determine the aspects of courses that might require more attention from the professors. Hence, forecasting of student enrollment and performance plays a vital role in making crucial decisions based on previous time-bound records. The aim of this work is to solve a two-pronged problem viz:

1. Forecast the enrollment trend of students and analyze the performance of students based on historical records.
2. By leveraging the power of neural networks, we compare its performance with baseline traditional time series models and compare different neural network architectures.

The experiments performed in this work are targeted towards addressing the following research questions:

- ❖ What would be the future enrollment pattern of students in the fundamentals of programming class?
- ❖ Is there any significant difference in the student performance in fundamentals of programming class?
- ❖ Which is better for neural network time series forecasting between Deep neural networks and Shallow neural networks?

Classical Time Series Models are effective in performing the tasks of forecasting future values given sequence of historical records. However, they come with unwarranted assumptions and complexities. In this work, we leverage the power of LSTM (Long Short-Term Memory) models, which are memory-effective kind of Recurrent Neural Networks (RNN) to build forecasting models based on student records collected over time. The Exponential Smoothing model is used as baseline to demonstrate the robustness of the Neural Networks in handling Univariate Time Series data. By varying the sizes of the hidden layers, a comparison of Deep and Shallow Neural Networks is made to get an understanding of which would be better in forecasting student enrollment.

With the historical data on student performances in weekly lab exercises, projects and midterm exams, an in-depth statistical analysis is carried out to examine if there exist significant differences in the performances of students. This would provide an idea of the weak and strong areas of the course curriculum as well as identify possible factors for any differences that exist in the performances. This is done with the use of analysis of variance (ANOVA) statistical tests.

This report is organized as follows: Chapter two contains the background and related literature review for the research. Specifically, it gives an analysis of previous work done on student enrollment and forecasting using time series analysis models, comparison of the conventional time series models with the neural network counterpart as well as shallow versus deep neural networks. Chapter three is a demonstration of the research methods used in achieving the objectives and providing answers to the research questions. It describes the baseline models with which the neural network models are compared and the different architectures used for the neural networks. Essentially, the neural network algorithms used for the forecast as well as the parameters of the baseline models are described therein. Chapter four reveals the results after performing the time series analysis based on the traditional baselines and outcomes of the neural network experiments performed. Chapters five and six contains the discussion of the results and the recommendations, respectively.

CHAPTER TWO - BACKGROUND

2.1 Forecasting Enrollment /Performance

Several research projects have been carried out in a bid to predict the enrollments and performances of students with the aim of achieving effective decision making, which forms a great source of motivation for this research. Having an electronic school management system helps in the management of the university in terms of admissions, registrations, gradings and effective record management; however, without enrollment planning and student projection, there would be unavoidable problems of student complaints about course unavailability, merged courses, or dissolved sections [6]. Statistical modeling techniques used included Simple Moving Average (SMA), Simple Exponential Smoothing (SES) and Double Exponential Smoothing (DES). Using Mean Absolute Percentage Error (MAPE), results showed that the DES was the best among the models in projecting the number of student enrollments in Mindanao State University – Ilegan Institute of Technology (MSU – IIT) [6].

Forecasting the performance of students can be useful in taking early precautions and essential for educators to obtain early feedbacks and take necessary actions for improvement as well as gives instructors the opportunity to select aa student that is fit for a specific task, hence there is a need to explore better models to achieve better performance [1]. Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) algorithms are applied on the University of Minlo (Portugal) dataset to estimate student performance in final Math examination. Results show that the SVM outperformed the KNN by a slight margin with SVM having a correlation coefficient of 0.96 over the 0.95 coefficient of the KNN counterpart.

There is a growing interest among colleges in attempting to identify the factors that maximizes student enrollment, which helps administrators to identify potential students and make better allocation of their financial rewards in form of scholarships and financial aids according to (Slim, 2018)[8]. Machine Learning methods can be used to statistically achieve the enrollment predictability of such factors. A case study of the University of New Mexico (UNM) – whose results could be widely applicable to other universities was used to validate models built by Logistic Regression (LR), Support Vector Machine (SVM) and Semi-supervised probability methods. It was discovered that the LR and SVM predict the enrollment of applicants at an individual level while at the cohort level, the Semi-supervised model predicts the applicant enrollment successfully.

It is important for nations to develop and enhance student and teacher potentials if the nation is to achieve global competitiveness [14]. In a study conducted in Taiwan, the researchers opined that low birthrates have impacted the schools negatively thereby reducing its educational competitiveness – hence the need to monitor the decrease in the enrollments of students and overpopulation of the teachers. These formed the motivation to research on student enrollment and teacher statistics forecasting based on time series analysis. A combination of Whale Optimization Algorithm (WOA) and Support Vector Regression (SVR) was invented (WOASVR) to determine the trends of students and teachers in Taiwan so as to get good accuracy in time series forecasting analysis. In comparison with other common models, the WOASVR proved better with the lowest Mean Absolute Percentage Error (MAPE) and this provides accurate information for developing educational policies and responses.

2.2 Classical vs Neural Network forecasting

Existing literature have placed emphasis on only two forecasting methods – conventional time series and regression analysis which have been used in many applications with methods ranging from moving averages and Autoregressive Integrated Moving Averages (ARIMA) to Exponential Smoothing amongst others. The power and usefulness of Artificial Neural Networks (ANNs) for forecasting was employed by forecasting the future container throughput at Bangkok port [4]. Specifically, the Multi-Layer perceptron (MLP) was explored together with Linear regression and their performances measured by Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). It was noted that the Neural Network Multilayer Perceptron proves better at predicting the container throughput which is vital for Thailand's economics.

In furtherance to research on demonstrating the power of neural networks in forecasting, a study by [3] on student enrollment forecasts and a case study of Federal University of Technology, Owerri – Nigeria, reveals that student enrollment provides information for decision making and budget planning. To achieve the objective of good forecasting of enrollment, a Multi-Layer Feed-Forward Artificial Neural Network (MLFF ANN) and an Ordinary Least Square (OLS) models were built and performance measures of Mean Square Error (MSE), Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) were used to assess the performances of the two models. The results suggested that the MLFF provides better prediction of the student enrollment, having the lowest error values.

The superiority of deep learning-based algorithms over traditional time series models is demonstrated by [9]. The researchers concur that while several conventional time series models such as ARIMA has demonstrated to be very useful in performing forecasts, recent advancement in the computational power of computers and the coming of advanced machine learning algorithms proves useful in analyzing time series data as well. Applying the ARIMA and LSTM models on historical monthly financial time series from yahoo finance website, they attempted to determine whether and how the deep learning algorithms supersedes the traditional-based algorithms. The results show that the LSTM outperforms the ARIMA model with the former having an average reduction in error between 84 – 87% compared to the later. This advocates the usefulness of applying machine/deep learning algorithms and techniques on time series data which forms a huge motivation for this research in forecasting student enrollment and performance on historical student records.

A research performed by [11] on a comprehensive analysis of time series forecasting using neural networks, revealed that although classical methods like ARIMA and Hidden Markov Models (HMM) proved to be successful in modeling time series data, they are however subject to certain limitations [11]. One key issue identified is that the HMM follows a one-step Markov assumption that the current/future states of a time series data are independent of the past states. The assumption that there exist some hidden states responsible for generating the observations also makes it a weak model. The ARIMA is not short of flaws as well because it is only suitable for univariate time series analysis and the data must be stationary coupled with the fact that it cannot be effectively used to model nonlinear time series accurately. It was discovered that the massive volume of data, together with recent progress in the processing powers of computers has enabled researchers to develop more sophisticated algorithms using neural networks.

2.3 Deep Versus Shallow Neural Networks

While the robust nature of the neural network time series forecasting makes it a great choice for solving forecasting problems, it is imperative to pay attention to the kind of network architectures and configurations that would bring about a model with the minimum loss values and greater efficiency. Having to determine the architecture that best answer the questions that pertains to this research, it is therefore necessary to study a couple of work that has been done involving comparison of the performance of deep versus shallow neural networks.

An investigation on the performance of different neural network architectures was carried out on the task of automatic music genre classification using four datasets of different sizes [8]. Specifically, they explored the effect of the depth of network architectures for automatic music genre classification. It was discovered that shallow neural networks are more suitable for use when the dataset is small. The experiments were performed on different categories of datasets including the GTZAN datasets. The results show an accuracy of 73.2% in favor of the shallow models for the GTZAN dataset. However, the deep model was better in the presence of data augmentation using time stretching and pitch shifting. Hence, for applications that would require small size of data, a deep neural network architecture is not recommended.

The power of neural networks in accurate gear defect detection in induction machine-based system varied deep and shallow neural networks [2]. It was discovered that the use of shallow neural network instead of a deep one produced the better results. This is in congruence with [6] and provides a great deal of motivation for this work in general and a part of the research questions/objectives.

CHAPTER THREE - Methodology

This chapter describes the methods used in achieving the research objectives and answering the research questions. Source and description of dataset used, time series and the baseline models together with the neural network model and architecture used in the research are described here.

3.1 Source and Description of Dataset

This research uses:

1. 25 years historical records of student enrollment from 1995 to 2020 in fundamentals of programming course obtained from the office of the registrar.
2. 5 years student performance records with annual average scores in weekly Labs, Projects, three mid-term exams and final exams.

The first category is a univariate time series data used to forecast future enrollment of students whereas the second category is a multivariate time series data having the final exams as the target variable.

3.1.1 Test for Stationarity of Data

Before using the traditional time series model as a baseline, it is important to determine if the data is stationary. A time series is said to be stationary if its statistical properties – mean, variance or standard deviation does not vary with time. The stationarity of the dataset for this research is determined using:

- Plotting rolling mean and rolling standard deviation

Using Dicky-Fuller Statistics test

- **Hypothesis:**
  - $H_0: The\ dataset\ is\ stationary$
  - $H_1: The\ dataset\ is\ NOT\ stationary$

Decision: Reject the null hypothesis, H_0 if the $P - value\ statistic$ is greater than 0.01 level of significance

3.2 Time Series Analysis Described

Time Series is a series or sequence of observations/data points over a certain period of time or simply indexed with time. Usually, it is taking a series of observations $y_1, y_2, \ldots\ y_t$ and telling in advance what y_{t+1} would be.

The objective of the time series is to extract information from historical data and forecast future behavior of the series based on past pattern. The future hence depends on the past or simply, we can get the future as a function of the past:

$$y_{t+1} = f(y_t, y_{t-1}, y_{t-2}, \ldots) \tag{1}$$

A graph of the time series describes the pattern of the historical data. That is, a movement of a point with the passage of time. This movement may be affected by economic, sociological, psychological, or other factors, bringing about different kinds of patterns in the data. Thus, there are various versions of patters that a time series data might have.

3.2 Time Series Patterns

A time series could be a long or short term upward or downward trend of the series. An illustration of a time series that follows a trend pattern is shown in Figure 1.

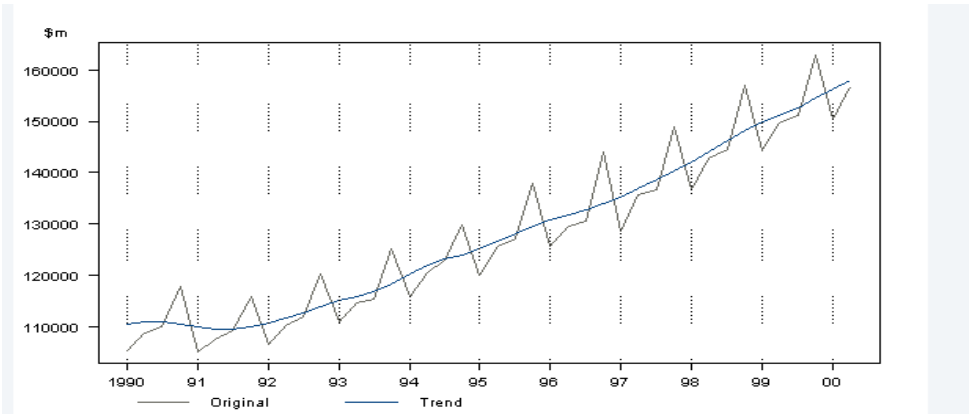

Figure 1: Example of Time Series Trend Pattern

A time series could be an oscillation or swings about a trend line or curve. This is easily seen in business cycles where there are moments or intervals of prosperity, recession, depression, or recovery. A time series cyclic pattern is exemplified in figure 2.

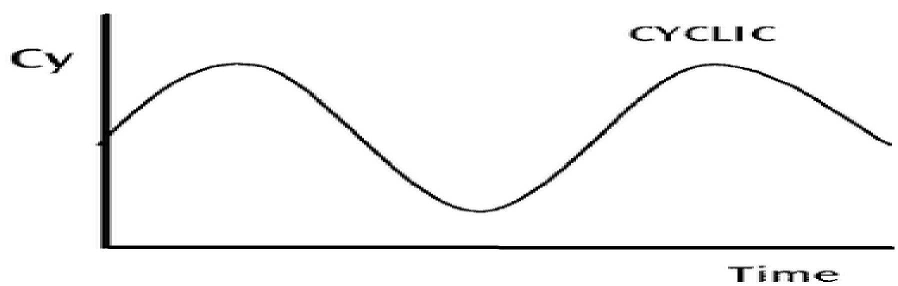

Figure 2: Example of Time Series Cyclic Pattern

A time series can also have identical patterns during particular periods of time usually months or quarters of successive years as shown in Figure 3 below between second and third quarters.

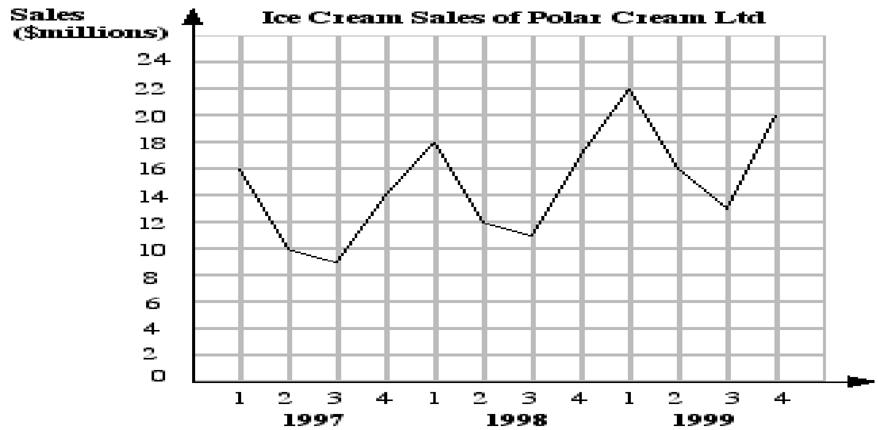

Figure 3: Example of Time Series Seasonal Pattern

A time series can have random or sporadic movement of the series. As shown in Figure 4, the series does not follow a particular trend and lacks cyclic or seasonal pattern.

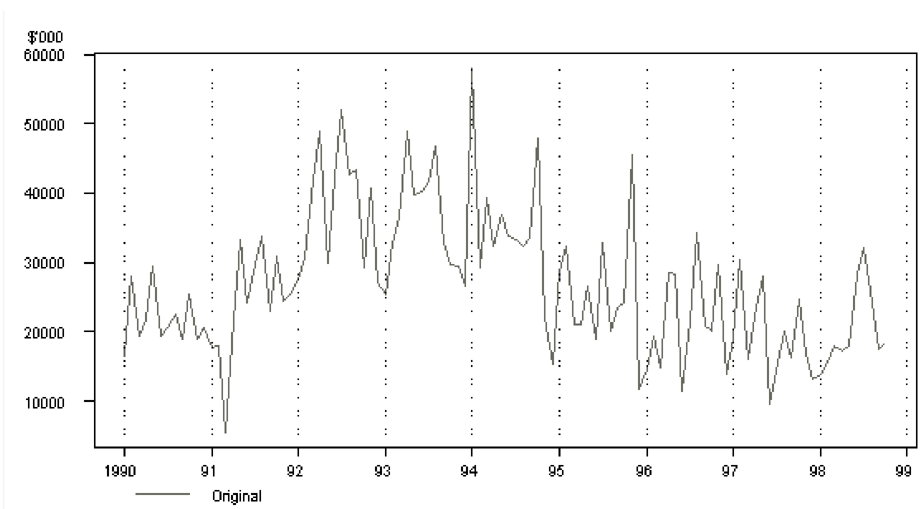

Figure 4: Example of Time Series Irregular Pattern

The first step towards building forecasting models based on time series historical data is to visualize the data and have an idea of the type of pattern it has. That would determine the type of model that is best suitable for the data. In this work, the python programming language is used to both visualize the dataset obtained and building the time series baseline and neural network models.

3.3 Simple Exponential Smoothing

Simple Exponential Smoothing (SES) is useful to forecast future values using a weighted average of all previous values in the series. It is useful for forecasting a series with no trend and no seasonality as well as series with no stationarity. Since the data for this research is not possess the aforementioned properties, classical univariate models like ARIMA, ARMA, SARIMA, etc. are not suitable for use as baselines.

For a SES, the forecast is the estimated level at the most recent time point. That is:

$$F_{t+i} = L_t = \alpha y_t + (1 - \alpha)L_{t-1} \text{ .... } (2)$$

where L_t is the estimated level at time, t

This implies taking the level at the previous time step and integrating with the information from the most recent data point.

This is a weighted average with α as the weights called the smoothing constant where $0 < \alpha < 1$. It is called exponential smoothing because the weights reduce exponentially into the past. Suppose L_{t-1} is being replaced by its own expression in ......(2):

$$L_t = \alpha y_t + (1 - \alpha)L_{t-1} = \alpha y_t + (1 - \alpha)[\alpha y_{t-1} + (1 - \alpha)L_{t-2}]$$

$$= \alpha y_t + \alpha(1 - \alpha)y_{t-1} + (1 - \alpha)^2 L_{t-2} \text{ .... } (3)$$

Replacing L_{t-2} with its own expression further reduces the value of the weights hence the name exponential smoothing.

The value of the smoothing constant α determines how much weight is given to the past. There are two key scenarios to watch out for:

- When $\alpha = 1$: The past values will have no effect on the forecast and the algorithm does not learn anything – leading to under-smoothing.
- When $\alpha = 0$: The past values have equal effect on the forecast values and leads to over-smoothing.

Thus, the exponential smoothing constant would need to be chosen wisely between zero and one.

3.4 Analysis of Variance (ANOVA) for Student Performance

The Analysis of Variance (ANOVA) is used to analyze the differences that exist among means of data. In this research, a one-way ANOVA which usually consist of a single factor having several levels and observations at each level, is used to analyze the students' performance. The model equation is given by:

$y_{ij} = \alpha + \beta_i + \epsilon$……………………………………………………………………………(4)

Where y_{ij} is the jth data point at level i, α is the grand mean and ϵ is the residual or error term.

The ANOVA is used to test the following hypothesis:

H_o: $All\ the\ means\ are\ equal\ or\ there\ is\ not\ significant\ differnce\ in\ the\ population\ mean$

H_1: $There\ exist\ a\ significant\ difference\ among\ the\ population\ means$

Decision Rule: Reject the null hypothesis if the p-value is greater than the level of significance.

3.5 Long Short – Term Memory (LSTM) Model Architecture

LSTM models are special type of Recurrent Neural Networks (RNN). RNNs are robust algorithms used in modeling of sequential data. Figure 5 shows a typical RNN model. For example, it takes the first input from the student enrollment dataset in the year $x_0 = 1995$ with $h_0 = 427$ students, keeps a copy of it in its memory and takes in the year $x_1 = 1996$ with $h_1 = 660$ students, keeps a copy of it in memory, and so on.

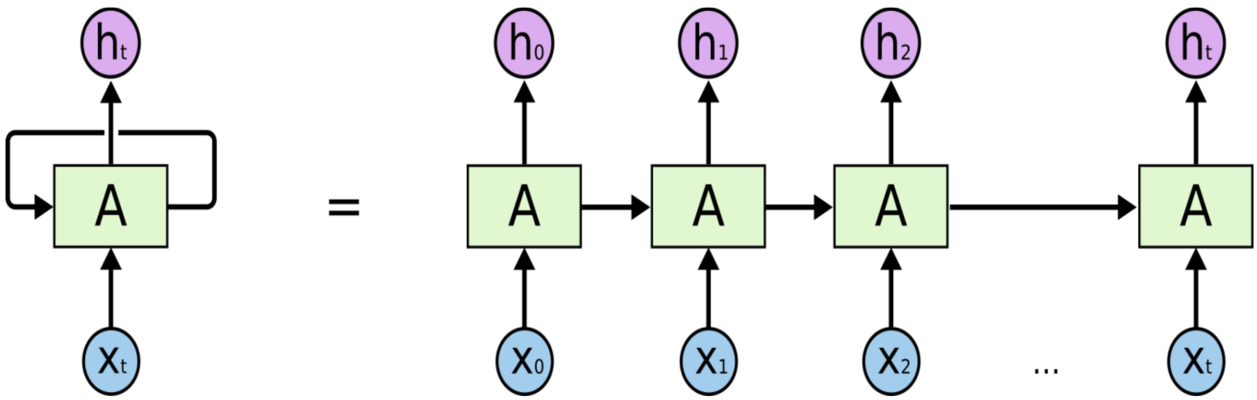

Figure 5: RNN Architecture

Its ability to remember its input because of the possession of an internal memory makes it great for processing sequential data – hence a strong motivation to use in forecasting historical time series data on student enrollment and performance.

The memorization of the sequential data is made possible through gates (as in figure 6) which forms a part of the inbuilt memory of the LSTM. The LSTM is robust is such a way that it automatically extract features from data and would not require that the time series data pass a stationarity test unlike the classical time series models like ARMA, ARIMA or SARIMA which requires stationarity.

In this work, the Root Mean Square Error (RMSE) is an assessment metric often used in measuring model performance. It works by measuring the differences between the predicted and actual values, otherwise called the residuals. Mathematically,

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}{N}} \quad \ldots\ldots(6)$$

Where $N = Total\ number\ of\ observations,$

$x_i = actual\ values\ and\ \hat{x}_i = \ predicted\ values$

There are three gates in the LSTM architecture:

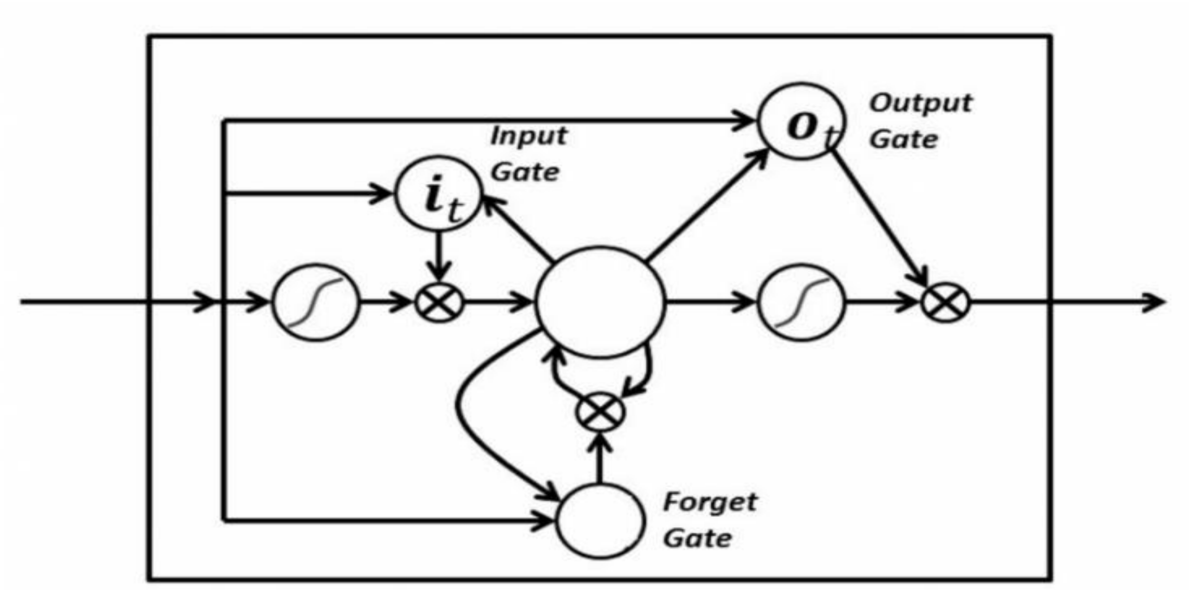

Figure 6: LSTM Processing Gates

1. The forget gate outputs a number between 0 and 1and have a sigmoid activation function. Here, a 0 tells the gate to completely forget the input and a 1 signifies keeping all the information.

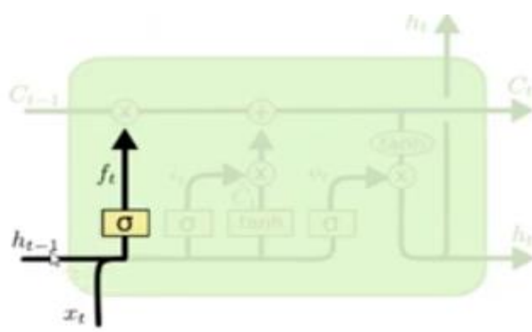

Figure 7: Forget Gate of LSTM Architecture

2. The input gate has a sigmoid and Tanh activation functions for deciding values to be updated and giving weights to the values added to the state respectively.

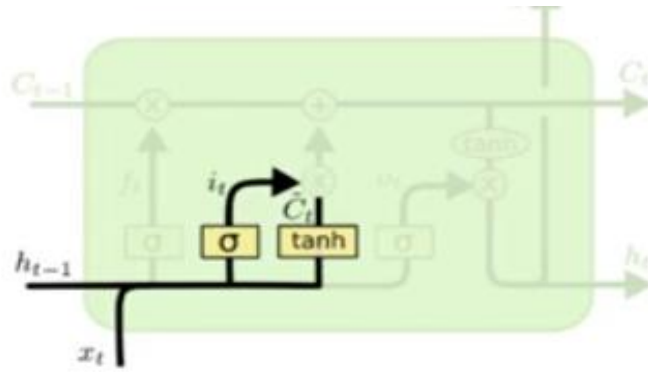

Figure 8: Input Gate of LSTM Architecture

3. The output gate possesses sigmoid activation for deciding the values for output and Tanh for giving weights to the values of the output.

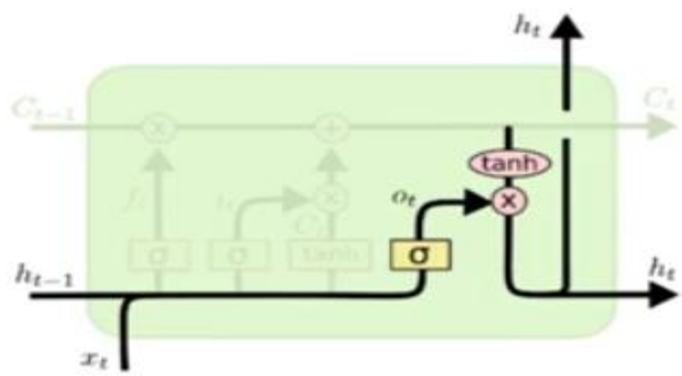

Figure 9: Output Gate of LSTM Architecture

CHAPTER FOUR - Results

This chapter discusses the results of the baseline time series models and the neural network models by first presenting the visualization pattern of the dataset used for the research.

4.1 Data Visualization of Student Enrollment

Figure 10 shows the pattern of the 25-year historical record of student enrollment in CIS 200 from 1995 – 2020. The data does not follow a particular trend, neither is there presence of seasonality nor cycles. It can be at best, described as irregular. From the visual point of it, it lacks stationarity – the points do not seem to move in parallel with the x-axis. This would be confirmed through the result of performing stationarity tests which would in turn determine the classical time series baseline models to apply.

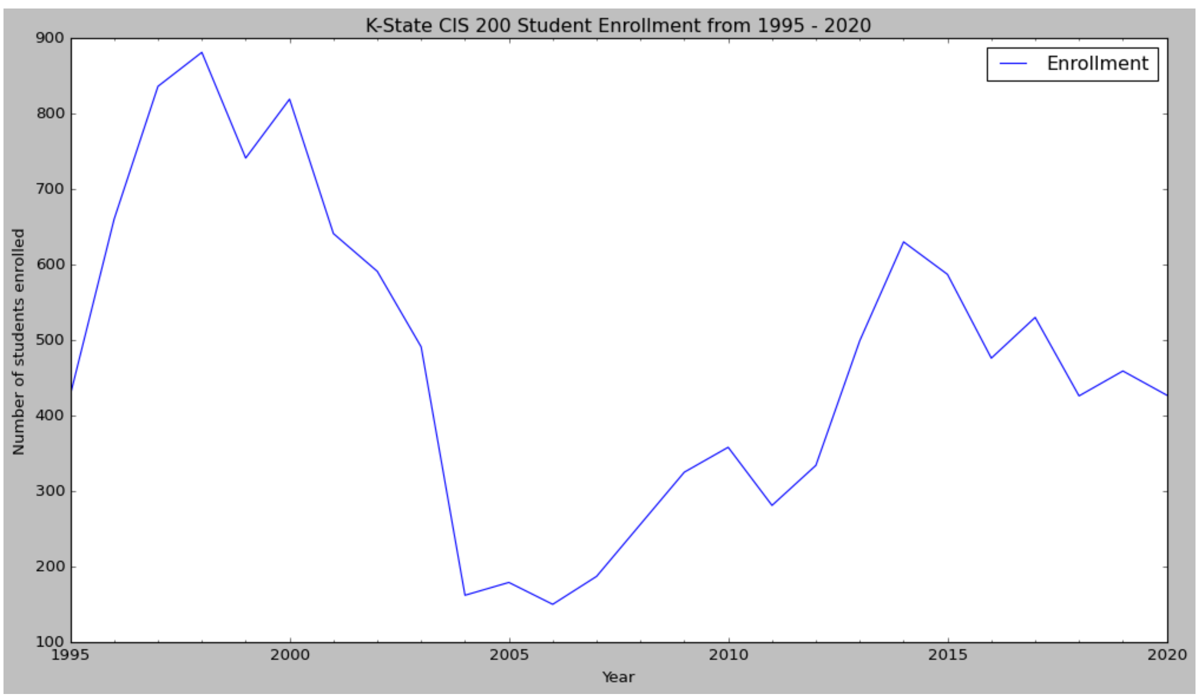

Figure 10: K-State CIS 200 Student Enrollment (1995 - 2020)

4.2 Stationarity

Time series data would be stationary if the statistical properties are constant with time. This is checked by using rolling statistics (plotting the rolling mean and the rolling standard deviations) as well as via statistical tests - Using Augmented Dicky-Fuller (ADF) test. The plot (figure 11) shows that the mean and standard deviation are not constant over time and hence not stationary. The result of the ADF test also shows a high p-value of 0.026 (less than a 0.01 significant level) and thus not stationary.

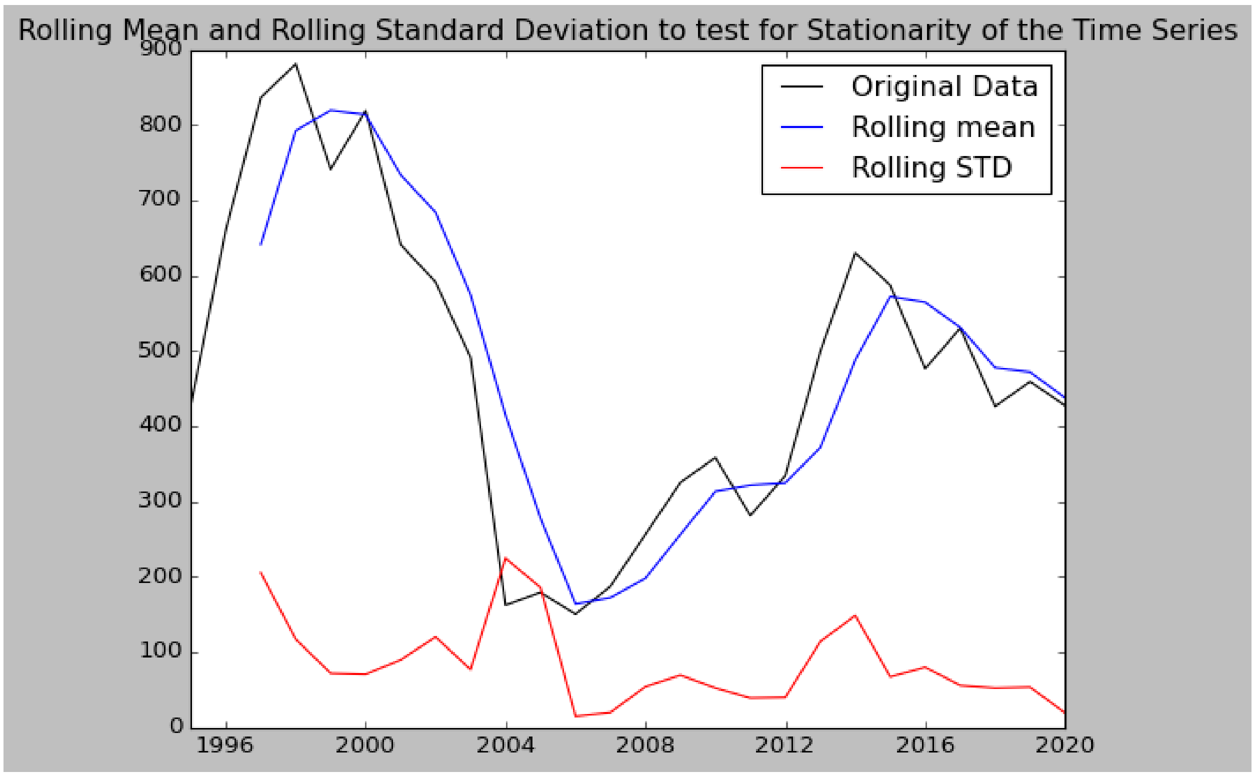

Figure 11: Stationarity Test using Rolling Mean and Rolling Standard Deviation

4.3 Exponential Smoothing Baseline Forecast

The baseline Single Exponential Smoothing model shows an irregular pattern in future enrollment of the students at a smoothing constant of 0.4 as shown in figure 13 obtained using python's time series libraries and in-built packages. A performance metric of Root Mean Square Error of 0.27 which is an indication that the model is a good fit for forecasting. This performance would be compared to the neural network result to demonstrate the efficacy of the later.

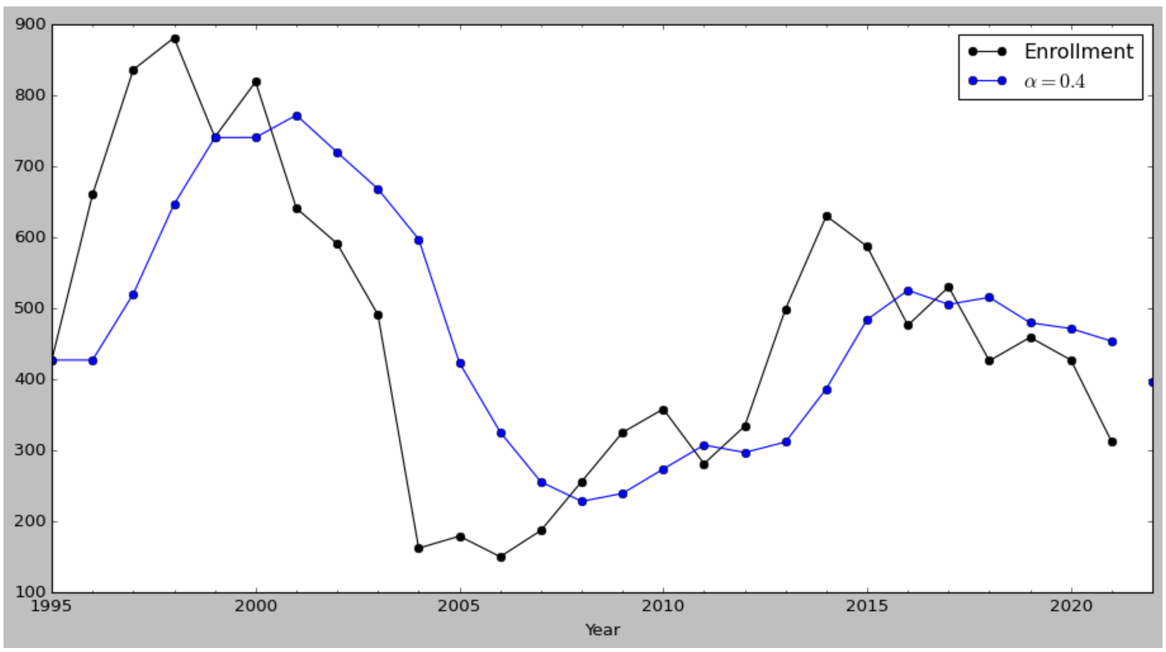

Figure 12: Exponential Smoothing 0.4 Smoothing Constant

4.4 Result from the LSTM Forecasting Models

Figure 14 displays the plot for the forecast pattern using Tensor Flow and a single input, single hidden and single output layers which forms a Vanilla or Shallow network architecture. This is necessary as it would be compared with the result from using a deep neural network architecture shown in figure 15. The RMSE for the shallow and deep networks are 0.11 and 0.24 respectively. A lower RMSE value indicates how well the model fits the data and makes good prediction. Thus, the Vanilla neural network outperforms the deep neural network.

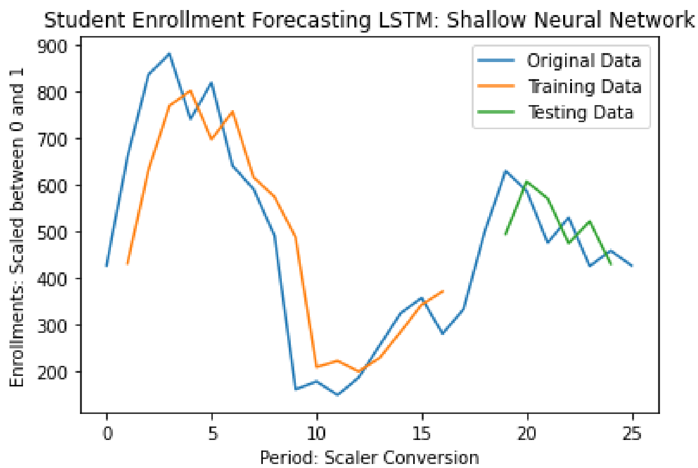

Figure 13: Shallow Neural Network

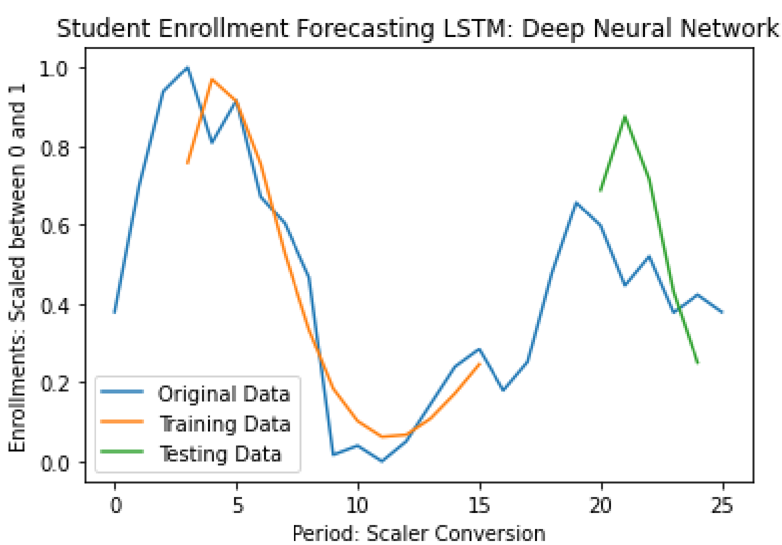

Figure 14: Deep Neural Network

4.5 Result from Student Performance Analysis

The following are the outcomes of the statistical analysis of the student performances from the 5-year historical records.

4.5.1 Summary Statistics of Student Performances

The following figures show the descriptive statistics of the student performances in the labs, projects, and midterm exams. This reveals the average and percentile scores on the assessment sections over the five-year period and it is relevant in providing and insight into the overall distribution of the performances to have a sense of judgment over a drop or increase in the scores, which would form the bases for discussion and recommendation in subsequent chapters.

The first two labs prove to have the best mean performance with 75% of the students performing above average as shown in figure 15.

	Lab1	Lab2	Lab3	Lab4	Lab5	Lab6	Lab7	Lab8	Lab9	Lab10	Lab11
count	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.00000
mean	9.658000	9.658000	9.374000	8.956000	8.672000	8.412000	8.550000	8.338000	7.784000	8.378000	8.00800
std	0.075631	0.074297	0.399475	0.460359	0.650208	0.449077	0.645949	0.635035	0.939377	0.775996	0.89617
min	9.560000	9.550000	8.820000	8.480000	7.870000	7.930000	7.610000	7.430000	6.500000	7.240000	7.19000
25%	9.600000	9.620000	9.220000	8.670000	8.370000	8.060000	8.380000	7.980000	7.180000	8.090000	7.31000
50%	9.680000	9.680000	9.320000	8.730000	8.480000	8.330000	8.550000	8.610000	8.070000	8.430000	7.59000
75%	9.710000	9.700000	9.670000	9.440000	9.120000	8.760000	8.840000	8.620000	8.310000	8.860000	8.90000
max	9.740000	9.740000	9.840000	9.460000	9.520000	8.980000	9.370000	9.050000	8.860000	9.270000	9.05000

Figure 15: Summary of Lab Performances

The first project demonstrates to have the highest average performance with more than three – quarters performing above average as seen in Figure 16. Project 7 has the lowest average performance and likely factors responsible would be discussed later in the work.

	Proj1	Proj2	Proj3	Proj4	Proj5	Proj6	Proj7	Proj8
count	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000	5.000000
mean	46.762000	44.540000	42.198000	39.256000	40.168000	37.310000	36.880000	37.708000
std	2.990965	2.046607	2.209688	5.314201	2.794731	2.839252	2.353072	2.414036
min	41.890000	42.100000	39.500000	31.410000	35.840000	33.800000	33.770000	34.340000
25%	46.300000	42.940000	40.780000	38.120000	40.000000	34.760000	35.690000	37.370000
50%	47.370000	44.540000	41.820000	38.480000	40.240000	38.790000	36.520000	37.710000
75%	48.740000	46.360000	44.440000	43.160000	41.220000	38.990000	39.020000	37.980000
max	49.510000	46.760000	44.450000	45.110000	43.540000	40.210000	39.400000	41.140000

Figure 16: Summary of Project Performances

The summary statistics (Figure 17) for the exams shows a steady decrease in performance across the three mid-terms with a greater margin between the first and the third. This (along with results from the Labs and Projects) would give us an insight into the Analysis of Variance to identify the existence or non-existence of a significant difference in the mean performances.

	Exam1	Exam2	Exam3	FinalExam
count	5.00000	5.000000	5.000000	5.000000
mean	41.53600	38.084000	35.798000	70.688000
std	1.94145	1.675807	2.769182	5.951695
min	39.34000	36.480000	32.430000	64.180000
25%	39.58000	37.260000	33.440000	66.650000
50%	42.47000	37.760000	36.430000	69.900000
75%	42.71000	38.030000	37.900000	73.340000
max	43.58000	40.890000	38.790000	79.370000

Figure 17: Summary of Exam Performances

4.5.2 Normality Plots

To test if the dataset follows a Gaussian distribution across the different sections, probability plots (otherwise referred to as P-P plots) are plotted using the probplot package imported from Python's scipy module as shown in figures 18, 19 and 20. For the labs, the points are evenly distributed across the red line and there are not outliers identified.

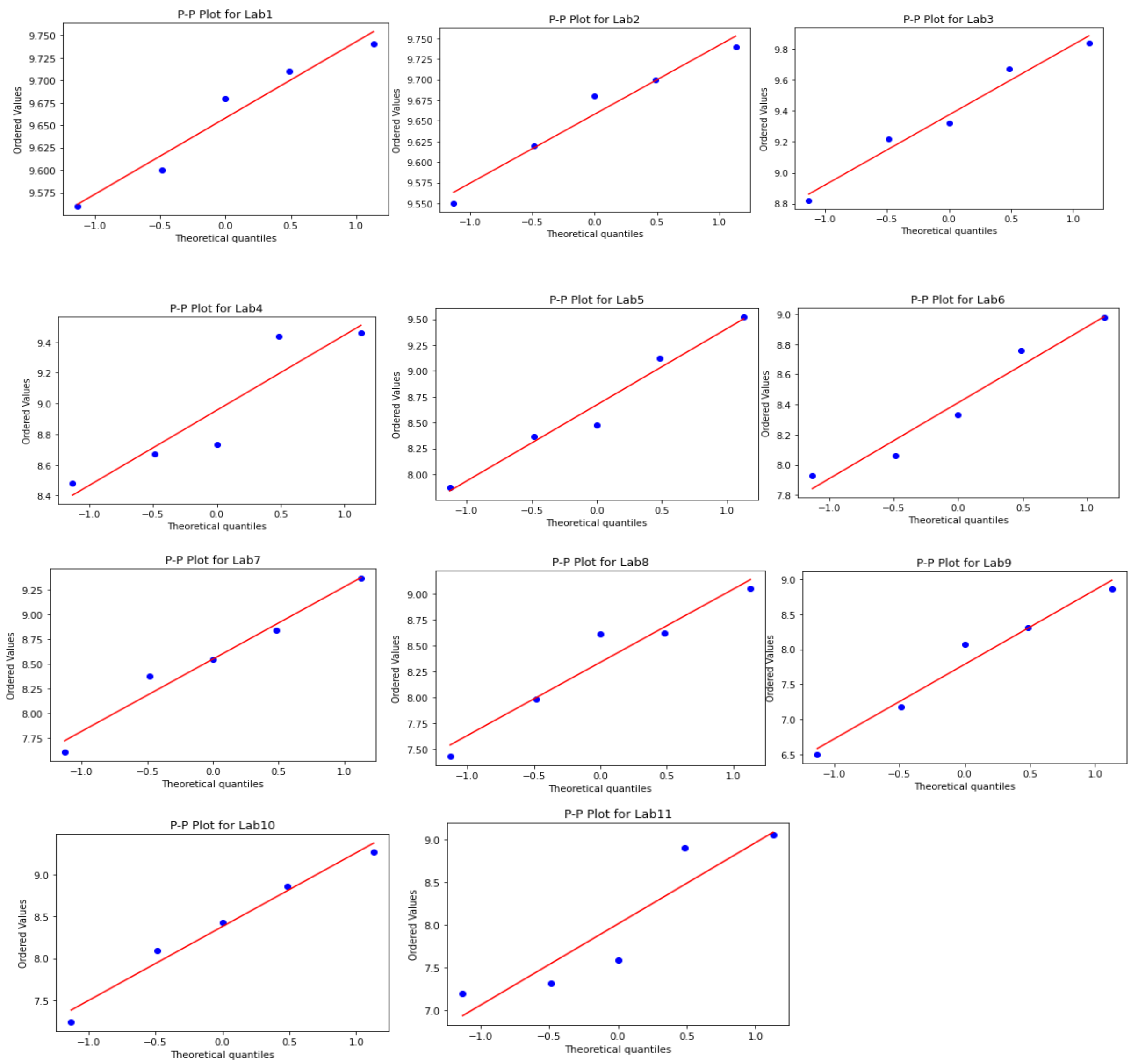

Figure 18: Probability plots for Labs

The P – P plots for the project performances suggests that the data is normally distributed as well. Although some points appear to be off for projects 3, 6 and 8, it would be justified by the Shapiro statistic test of normality.

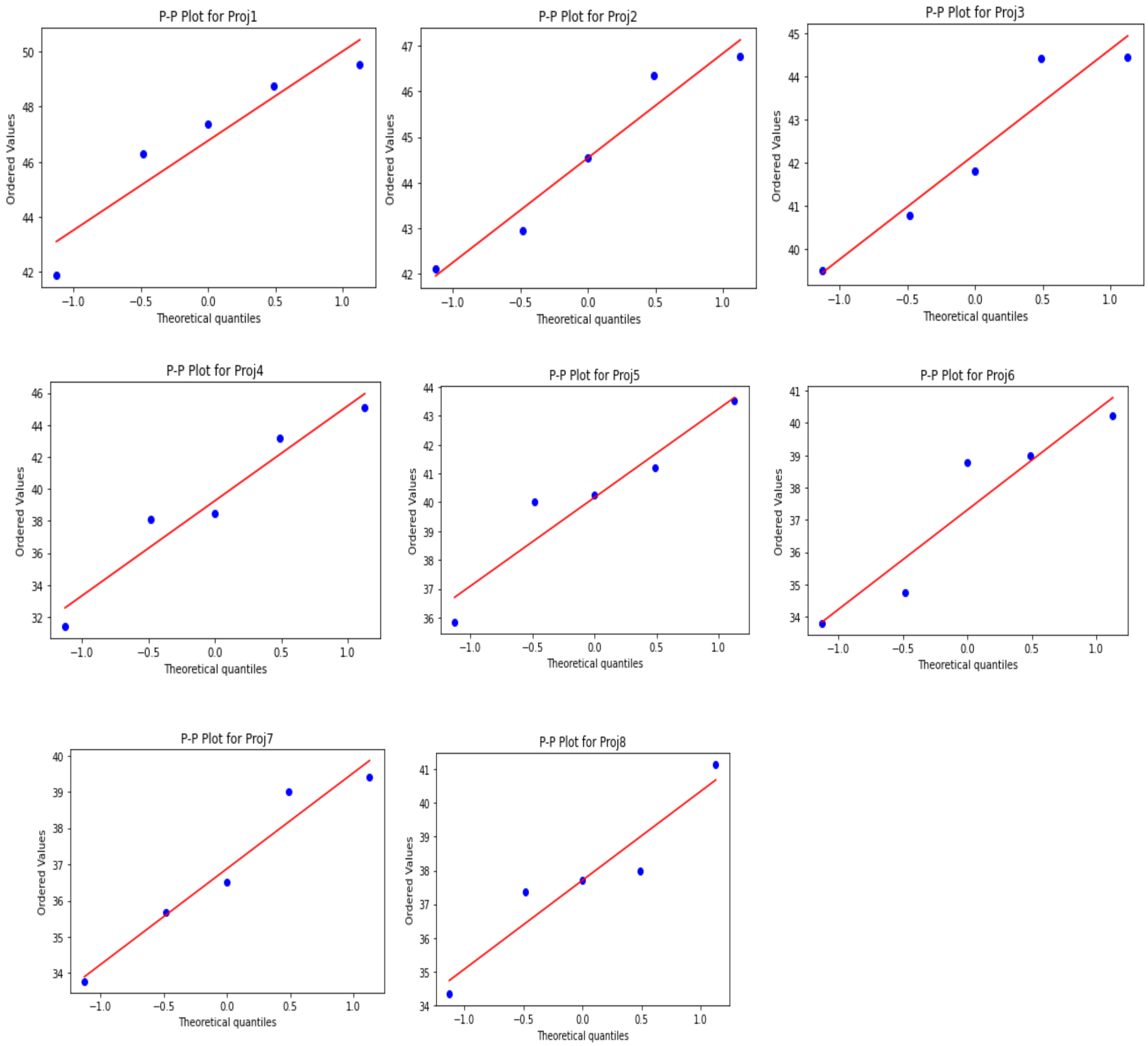

Figure 19: Probability Plot for Projects

The first and second mid-terms have couple of points off the line in each case but not sufficient to conclude non-normality. The supplementary Shapiro Wilk test would provide a better analysis of the model assumption test.

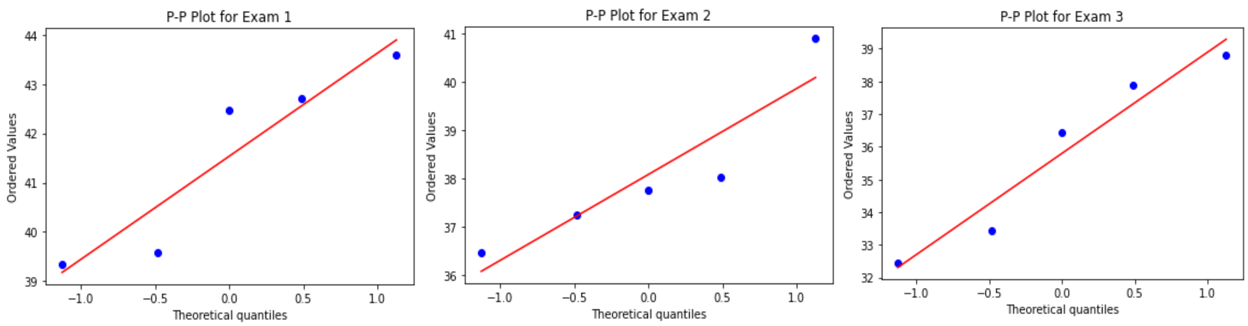

Figure 20: Probability Plots for Mid-Terms

4.5.3 Shapiro test for Normality

The Shapiro Wilk test is used to perform normality tests with the results shown in the tables below. The P-values for the Lab tests are all greater than 0.05 level of significance and shows that the Lab performances are normally distributed.

Lab	**Test – Statistic**	**P-Value**
Lab1	0.9340	0.624
Lab2	0.9600	0.808
Lab3	0.9692	0.870
Lab4	0.8268	0.132
Lab5	0.965	0.845
Lab6	0.931	0.606
Lab7	0.9829	0.949
Lab8	0.940	0.664
Lab9	0.961	0.812
Lab10	0.978	0.925
Lab11	0.817	0.111

Table 1: Shapiro test for Normality - Lab Scores

Table 2 shows the Shapiro test for normality and has all the p-values greater than 0.05 level of significance, depicting that the data is normally distributed and ANOVA model assumption is satisfied.

Projects	**Test – Statistic**	**P-Value**
Proj1	0.893	0.373
Proj2	0.918	0.520
Proj3	0.892	0.366
Proj4	0.943	0.688
Proj5	0.938	0.649
Proj6	0.865	0.247
Proj7	0.933	0.617
Proj8	0.934	0.626

Table 2: Shapiro test for Normality – Projects

Table 3 below shows that the mid-term exam performances also follow a Gaussian distribution having the p-values larger than the test level of significance of 5%.

Mid-Term Exams	**Test – Statistic**	**P-Value**
Exam1	0.847	0.185
Exam2	0.865	0.246
Exam3	0.917	0.511

Table 3: Shapiro test for Normality - Exams

4.5.4 ANOVA Results for Weekly Lab Performances

Figure 21 shows the ANOVA results from the Lab scores. It shows a p-value of 0.000038 which means the null hypothesis of no significance difference among the mean scores would be rejected.

	df	sum_sq	mean_sq	F	PR(>F)
Labs	10.0	20.06756	2.006756	5.363793	0.000038
Residual	44.0	16.46172	0.374130	NaN	NaN

Figure 21: ANOVA Output for Lab Performance

Since there exist significant difference among the means and the ANOVA model does not tell specifically where the differences lie, the Tukey Multiple comparison test (Table 4.1) is performed to show the pairwise datapoints where that are significantly different. This would provide a clue as to the reason that accounts for these differences.

Multiple Comparison of Means - Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
Lab1	Lab10	-1.28	0.0618	-2.5926	0.0326	False
Lab1	Lab11	-1.65	0.0045	-2.9626	-0.3374	True
Lab1	Lab2	0.0	0.9	-1.3126	1.3126	False
Lab1	Lab3	-0.284	0.9	-1.5966	1.0286	False
Lab1	Lab4	-0.702	0.739	-2.0146	0.6106	False
Lab1	Lab5	-0.986	0.3062	-2.2986	0.3266	False
Lab1	Lab6	-1.246	0.0765	-2.5586	0.0666	False
Lab1	Lab7	-1.108	0.1686	-2.4206	0.2046	False
Lab1	Lab8	-1.32	0.0477	-2.6326	-0.0074	True
Lab1	Lab9	-1.874	0.001	-3.1866	-0.5614	True
Lab10	Lab11	-0.37	0.9	-1.6826	0.9426	False
Lab10	Lab2	1.28	0.0618	-0.0326	2.5926	False
Lab10	Lab3	0.996	0.2928	-0.3166	2.3086	False
Lab10	Lab4	0.578	0.9	-0.7346	1.8906	False
Lab10	Lab5	0.294	0.9	-1.0186	1.6066	False
Lab10	Lab6	0.034	0.9	-1.2786	1.3466	False
Lab10	Lab7	0.172	0.9	-1.1406	1.4846	False
Lab10	Lab8	-0.04	0.9	-1.3526	1.2726	False
Lab10	Lab9	-0.594	0.9	-1.9066	0.7186	False
Lab11	Lab2	1.65	0.0045	0.3374	2.9626	True
Lab11	Lab3	1.366	0.0352	0.0534	2.6786	True
Lab11	Lab4	0.948	0.3614	-0.3646	2.2606	False
Lab11	Lab5	0.664	0.7962	-0.6486	1.9766	False
Lab11	Lab6	0.404	0.9	-0.9086	1.7166	False
Lab11	Lab7	0.542	0.9	-0.7706	1.8546	False
Lab11	Lab8	0.33	0.9	-0.9826	1.6426	False
Lab11	Lab9	-0.224	0.9	-1.5366	1.0886	False

Table 4.1: Pairwise Comparison of Lab Mean

Multiple Comparison of Means - Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
Lab2	Lab3	-0.284	0.9	-1.5966	1.0286	False
Lab2	Lab4	-0.702	0.739	-2.0146	0.6106	False
Lab2	Lab5	-0.986	0.3062	-2.2986	0.3266	False
Lab2	Lab6	-1.246	0.0765	-2.5586	0.0666	False
Lab2	Lab7	-1.108	0.1686	-2.4206	0.2046	False
Lab2	Lab8	-1.32	0.0477	-2.6326	-0.0074	True
Lab2	Lab9	-1.874	0.001	-3.1866	-0.5614	True
Lab3	Lab4	-0.418	0.9	-1.7306	0.8946	False
Lab3	Lab5	-0.702	0.739	-2.0146	0.6106	False
Lab3	Lab6	-0.962	0.34	-2.2746	0.3506	False
Lab3	Lab7	-0.824	0.5555	-2.1366	0.4886	False
Lab3	Lab8	-1.036	0.2426	-2.3486	0.2766	False
Lab3	Lab9	-1.59	0.0071	-2.9026	-0.2774	True
Lab4	Lab5	-0.284	0.9	-1.5966	1.0286	False
Lab4	Lab6	-0.544	0.9	-1.8566	0.7686	False
Lab4	Lab7	-0.406	0.9	-1.7186	0.9066	False
Lab4	Lab8	-0.618	0.8654	-1.9306	0.6946	False
Lab4	Lab9	-1.172	0.1186	-2.4846	0.1406	False
Lab5	Lab6	-0.26	0.9	-1.5726	1.0526	False
Lab5	Lab7	-0.122	0.9	-1.4346	1.1906	False
Lab5	Lab8	-0.334	0.9	-1.6466	0.9786	False
Lab5	Lab9	-0.888	0.4577	-2.2006	0.4246	False
Lab6	Lab7	0.138	0.9	-1.1746	1.4506	False
Lab6	Lab8	-0.074	0.9	-1.3866	1.2386	False
Lab6	Lab9	-0.628	0.8503	-1.9406	0.6846	False
Lab7	Lab8	-0.212	0.9	-1.5246	1.1006	False
Lab7	Lab9	-0.766	0.6428	-2.0786	0.5466	False
Lab8	Lab9	-0.554	0.9	-1.8666	0.7586	False

Table 4.2: Pairwise Comparison of Lab Mean Performances (Cont'd)

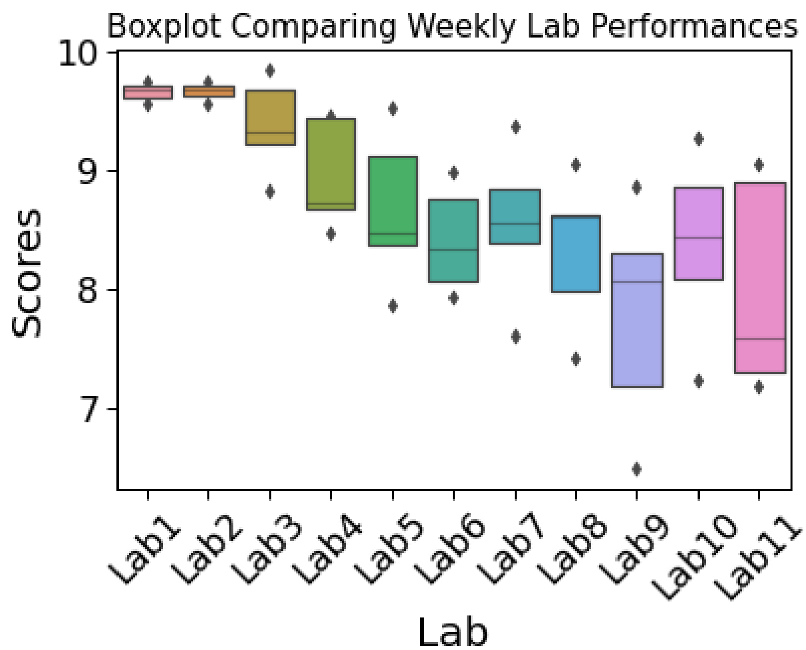

Figure 22: Boxplot for Multiple Comparison of Test for Weekly Labs

The boxplot containing sub-plots of the Labs shows the relationship among the mean scores across the Labs. It gives a visual representation of the overlap or non-overlap that exists between labs of no significant difference and those with significant differences respectively.

4.5.5 ANOVA Results for Project Performances

Figure 24 shows the ANOVA results from the Project scores. It shows a p-value of 0.00004 which means the null hypothesis of no significance difference among the mean scores would be rejected for the project performances.

	df	sum_sq	mean_sq	F	PR(>F)
Projects	7.0	455.332557	65.047508	7.080549	0.00004
Residual	32.0	293.977240	9.186789	NaN	NaN

Figure 23: ANOVA Output for Project Performance

The result is not sufficient to identify the pair of projects that are significantly different from each other, hence the Tukey Multiple comparison test (Table 4.2) is performed to show the pairwise datapoints where that are significantly different.

Multiple Comparison of Means - Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
Proj1	Proj2	-2.222	0.9	-8.4313	3.9873	False
Proj1	Proj3	-4.564	0.2846	-10.7733	1.6453	False
Proj1	Proj4	-7.506	0.0093	-13.7153	-1.2967	True
Proj1	Proj5	-6.594	0.0311	-12.8033	-0.3847	True
Proj1	Proj6	-9.452	0.001	-15.6613	-3.2427	True
Proj1	Proj7	-9.882	0.001	-16.0913	-3.6727	True
Proj1	Proj8	-9.054	0.001	-15.2633	-2.8447	True
Proj2	Proj3	-2.342	0.9	-8.5513	3.8673	False
Proj2	Proj4	-5.284	0.1424	-11.4933	0.9253	False
Proj2	Proj5	-4.372	0.3346	-10.5813	1.8373	False
Proj2	Proj6	-7.23	0.0135	-13.4393	-1.0207	True
Proj2	Proj7	-7.66	0.0075	-13.8693	-1.4507	True
Proj2	Proj8	-6.832	0.0229	-13.0413	-0.6227	True
Proj3	Proj4	-2.942	0.7576	-9.1513	3.2673	False
Proj3	Proj5	-2.03	0.9	-8.2393	4.1793	False
Proj3	Proj6	-4.888	0.2117	-11.0973	1.3213	False
Proj3	Proj7	-5.318	0.1373	-11.5273	0.8913	False
Proj3	Proj8	-4.49	0.3033	-10.6993	1.7193	False
Proj4	Proj5	0.912	0.9	-5.2973	7.1213	False
Proj4	Proj6	-1.946	0.9	-8.1553	4.2633	False
Proj4	Proj7	-2.376	0.9	-8.5853	3.8333	False
Proj4	Proj8	-1.548	0.9	-7.7573	4.6613	False
Proj5	Proj6	-2.858	0.7822	-9.0673	3.3513	False
Proj5	Proj7	-3.288	0.6567	-9.4973	2.9213	False
Proj5	Proj8	-2.46	0.8983	-8.6693	3.7493	False
Proj6	Proj7	-0.43	0.9	-6.6393	5.7793	False
Proj6	Proj8	0.398	0.9	-5.8113	6.6073	False
Proj7	Proj8	0.828	0.9	-5.3813	7.0373	False

Table 5: Pairwise Comparison of Project Performance

The following boxplot in figure 25 depicts a visual relationship among the average performances in the projects.

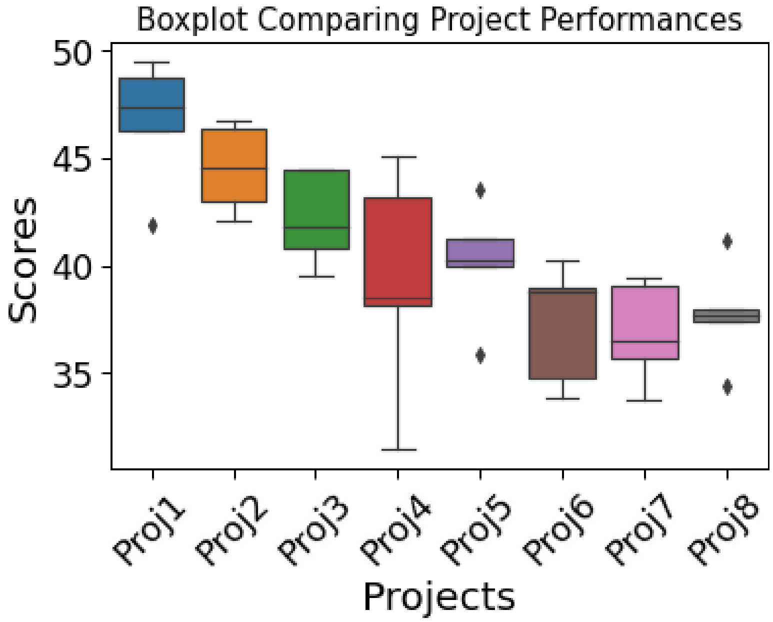

Figure 24: Boxplot for Multiple Comparison for Project scores

4.5.6 ANOVA Results for Mid-Term Performances

Figure 26 shows the ANOVA results from the midterm exams scores. It shows a p-value of 0.004465 which means the null hypothesis of no significance difference among the mean scores would be rejected for the project performances.

	df	sum_sq	mean_sq	F	PR(>F)
Exams	2.0	83.444573	41.722287	8.786149	0.004465
Residual	12.0	56.983720	4.748643	NaN	NaN

Figure 25: ANOVA Output for Mid-Term

The Tukey Multiple Comparison test suggests that the significant difference only exist between the first and third exams. Hence the reason why the p-value is so close to 0.05 level of significance.

```
Multiple Comparison of Means - Tukey HSD, FWER=0.05
====================================================
group1 group2 meandiff p-adj   lower   upper  reject
----------------------------------------------------
 Exam1  Exam2   -3.452 0.0664 -7.1271  0.2231  False
 Exam1  Exam3   -5.738 0.0035 -9.4131 -2.0629   True
 Exam2  Exam3   -2.286 0.2602 -5.9611  1.3891  False
```

Figure 26: Pairwise Comparison of Mid-Term Performance

The family of box plots shown in figure 28 shows a close overlap between Exam1 and Exam2 and between Exam2 and Exam3 but a wide margin between Exam1 and Exam3. This is important as it provides a visual analysis of the pairwise comparison of the means.

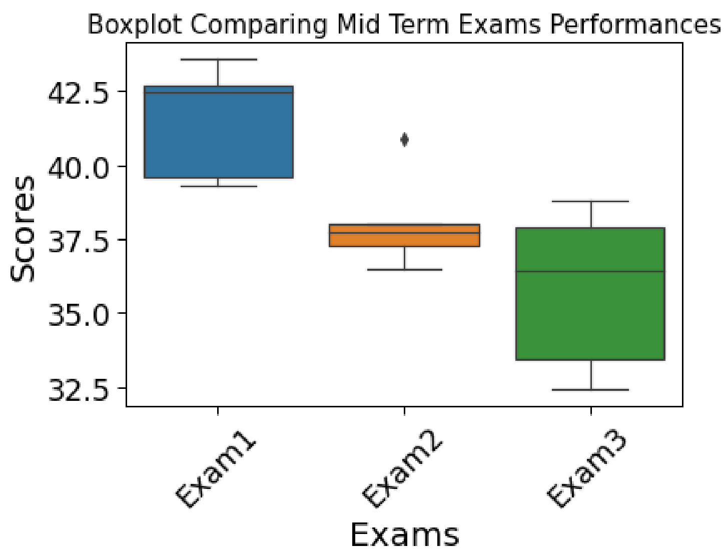

Figure 27: Boxplot for Multiple Comparison for Mid-term scores

CHAPTER FIVE - DISCUSSION

This section provides detailed discussion of the results from data visualizations to analysis performed on the datasets on student enrollment and performance. It reveals the weakness in the baseline model and the power of neural network models. It also provides in-depth analysis of the student performances over a 5-year period and gives an insight into possible areas of strength and weakness of the students, the topics that would need more attention and probable factors that accounts for the differences in the average performances on the different sections of the course.

5.1 Exponential Smoothing Baseline

The stationarity test of the data on student enrollment failed, making it impractical to use The Autoregressive Integrated Moving Average (ARIMA) model as a baseline unless the data is transformed for stationarity. However, it would not be logical to enforce stationarity as it is better to work with existing reality. Visualizing the data and statistical test using the Augmented Dickey – Fuller test reveals a high p-value of 0.0262 which is greater than 1% significant level in favor of the alternative hypothesis of the data set not being stationary. More so, the dataset is follows neither Trend or Seasonal pattern which makes the Single Exponential Smoothing method appropriate for use as a baseline.

An alternative to the classical ARIMA model is the Exponential Smoothing which was used as a baseline for the research. It depicts an irregular pattern of future student enrollment just like the use of neural network but however has a higher Root Mean Square Error (RMSE) which we use as our performance metric. The result of the exponential smoothing forecast at a smoothing constant value of 0.40 shows that in the future, the enrollment of the student would be irregular as it shows different up-and-down values as shown in figure 13. The Root Mean Square Error obtained is 0.27.

5.2 Shallow Versus Deep Neural Networks

Harnessing the power of neural networks which has a robust way of automatically extracting features from data without having to pass through the stationarity test proves to be more powerful. The LSTM models are of two folds – the Vanilla model (or simply shallow neural network) having one hidden layer and the stacked LSTM (or simply deep neural network) with more than one hidden layers. It was discovered that the Vanilla LSTM network proves better than the architecture with more hidden layers. In fact, the shallow neural network reported an RMSE value half of that of the deep neural network with the shallow network having an RMSE of 0.11 as against 0.24 for the deep neural network counterpart. This is in congruence with the report from a couple of literatures reviewed during the study [6] and [7].

5.3 Analysis of Student Performances

The 5-year performance record of the student is statistically analyzed using Analysis of Variance (ANOVA). The data is shown to follow a Gaussian distribution through P–P Plots (Probability plots) of each feature as well as through Shapiro Wilk test for normality. Having satisfied this assumption, the ANOVA analysis reveals a significant difference in the yearly average performance across the assessment sections – weekly labs, projects, and mid-terms.

Tukey multiple comparison of means is used to further detect the pair of assessment types that are significantly different as it helps figure out possible reasons for inconsistent performances. The analysis reveals that the scores of students on the first three labs and the last (11^{th}) lab are significantly different. This is because in some cases, the last lab would be optional and that would drop the average performance overall. Also, there is a significant difference between the average scores on Lab2 and Lab3 with the Lab9 respectively. During the ninth lab, the semester would have witnessed some dropouts in the courses, causing the average to be reduced.

The average performance on the projects is mostly consistent. The exception begins when projects 4,5,6,7 and 8 differs from project1. However, there exists no significant difference among projects 3 up to project 8. Following this, the students would have been more acquainted and gotten full grasp of the concepts after the second project thereby making the scores consistent.

The mid-terms report a significant difference only between the first mid-term and the third one. Again, some students who took the first mid-term would have dropped out, since the scores of the second and the third are not significantly different.

CHAPTER SIX - CONCLUSION

This research was conducted primarily to investigate, using a 25-year historical records of student enrollment in Programming Fundamentals course at Kansas State University – Manhattan Campus from 1995 to 2020, the future enrollment pattern for the course as well as statistically analyzing the students' performance using a 5-year student performance record.

The enrollment data depicts an irregular pattern in the student enrollment. Although traditional ARIMA models are quite effective in building time series forecasting models, it would fail to perform optimally in this case due to both the irregular nature of the data and failure of the data to attain stationarity which is an essential assumption the model must adhere to if it is to make great forecasts.

Both traditional time series and neural network models reveal irregular patterns in future enrollment as well. This is agreeable as several factors might be responsible for students' increase in enrollment for a period and a sharp decrease at some other points. Possible factors could be relocation from a particular region, low birth rate in a place as studied by [14].

Also, for people involved heavily in investments, a huge boom in the market would pave way for more financial commitment into enrolling more students while a poor performance or recession would in no doubt cause temporary delay in enrollment especially for large family size. It can be observed from the data visualization that enrollment in 2020 went down most probably due to the impact of the COVID-19.

More so, the DOT COM boom in the early 2000s would have both created a high need in computer scientist/knowledge of programming as well as aroused the interests of people in computing and these most probably accounts for the high student enrollment in that era.

The neural network-based time series forecasting proves to be better at forecasting student enrollment as it reports lower performance errors. It is therefore recommended that educational institutions embark on the use of this robust memory-based networks in order to achieve great results. Furthermore, it is preferable to experiment with a shallow neural network to achieve the minimum possible error in the forecast.

An in-depth statistical analysis of the students' performances shows that certain factors could be responsible for the significant difference in the yearly average performance of the students. The most probable factor that accounts for the inconsistency in the performance would be student dropouts, students not submitting a project or showing up for the lab, or students waiting till the last moment to get project done which would impact negatively on their scores in particular and the overall average in general. Overall, the Analysis of Variance is a robust model in testing the consistency in the average performance of students. If the differences in performance had left a wide margin between pairs of assessment groups, then it would be a way of revealing a weakness in a part of the course.

It is clear that students' performance drops as the programming projects gets more advanced. One potential solution to this problem could be that the professors in charge of the course involve a more pragmatic approach by probably gamifying the course where the students get more engaged and earn badges and virtual rewards as a way of motivating them [5]. Another possible solution would be to split the classes into groups as teamwork plays a great role in motivating students [7]. This, however, should come with caution of not duplicating each other's codes. Also, students tend to delay till the last minute to get started on the project assignments. It is suggested that the professors partition the course such that different sections of the projects are given specific timelines. That way, it does not only get the students more engaged, but they will be also more familiar with meeting deadlines.

REFERENCES

[1] Al-Shehri, H., Al-Qarni, A., Al-Saati, L., Batoaq, A., Badukhen, H., Alrashed, S., and Olatunji, S. O. (2017, April). Student performance prediction using support vector machine and k-nearest neighbor. *IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE*) (pp. 1-4). IEEE.

[2] Cirrincione, G., Kumar, R. R., Mohammadi, A., Kia, S. H., Barbiero, P., & Ferretti, J. (2020). Shallow versus deep neural networks in gear fault diagnosis. *IEEE Transactions on Energy Conversion,* 35(3), 1338-1347.

[3] Egbo, N. M., & Bartholomew, D. C. (2018). Forecasting students' enrollment using neural networks and ordinary least squares regression models. *Journal of Advanced Statistics*, 3(4).

[4] Gosasang, V., Chandraprakaikul, W., & Kiattisin, S. (2011). A comparison of traditional and neural networks forecasting techniques for container throughput at Bangkok port. *The Asian Journal of Shipping and Logistics,* 27(3), 463-482.

[5] Khaleel, F. L., Ashaari, N. S., Meriam, T. S., Wook, T., & Ismail, A. (2015). The of gamification application architecture for programming language course. In *Proceedings of the 9th international conference on ubiquitous information management and communication* (pp. 1-5).

[6] Lavilles, R. Q., & Arcilla, M. J. B. (2012). Enrollment forecasting for school management system. *International Journal of Modeling and Optimization*, 2(5), 563.

[7] Martins, S. W., Mendes, A. J., & Figueiredo, A. D. (2010, June). Diversifying activities to improve student performance in programming courses. In *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies* (pp. 540-545).

[8] Schindler, A., Lidy, T., & Rauber, A. (2016). Comparing Shallow versus Deep Neural Network Architectures for Automatic Music Genre Classification. In FMT (pp. 17-21).

[9] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2018, December). A comparison of ARIMA and LSTM in forecasting time series. In 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 1394-1401). IEEE.

[10] Slim, A., Hush, D., Ojah, T., & Babbitt, T. (2018). Predicting Student Enrollment Based on Student and College Characteristics. *International Educational Data Mining Society*.

[11] Tadayon, M., & Iwashita, Y. (2020). Comprehensive analysis of time series forecasting using neural networks. arXiv preprint arXiv:2001.09547.

[12] Thai-Nghe, N., Horváth, T., & Schmidt-Thieme, L. (2010, June). Factorization models for forecasting student performance. In Educational Data Mining 2011.

[13] URL: https://www.k-state.edu/media/newsreleases/2021-01/computer-science-initiative12821.html

[14] Yang, S., Chen, H. C., Chen, W. C., & Yang, C. H. (2020). Student Enrollment and Teacher Statistics Forecasting Based on Time-Series Analysis. Computational intelligence and neuroscience, 2020.

[15] Zhou, Y. (2016). High-Tech Industry. International Encyclopedia of Geography: People, the Earth, Environment and Technology: People, the Earth, Environment and Technology, 1-4.