SALES PREDICTION USING ARTIFICIAL NEURAL NETWORK

ILHAM ARIFF BIN MOHAMMAD FADZLI

Bachelor of Computer Science (Computer Systems & Networking)

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : ILHAM ARIFF BIN MOHAMMAD FADZLI

Date of Birth

Title : SALES PREDICTION USING ARTIFICIAL NEURAL NETWORK

Academic Session : 2022/2023

I declare that this thesis is classified as:

☐ CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*

☐ RESTRICTED (Contains restricted information as specified by the organization where research was done)*

☑ OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

(Supervisor's Signature)

New IC/Passport Number
Date:28/07/2023

Ts. Dr. Zuriani Mustaffa
Name of Supervisor
Date: 28/07/2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of *Doctor of Philosophy/ Master of Engineering/ Master of Science in …………………………..

(Supervisor's Signature)

Full Name : Ts. Dr. Zuriani Mustaffa

Position : Senior Lecturer

Date : 28/07/2023

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : ILHAM ARIFF BIN MOHAMMAD FADZLI

ID Number : CA19030

Date : 28 JULY 2023

SALES PREDICTION USING ARTIFICIAL NEURAL NETWORK

ILHAM ARIFF BIN MOHAMMAD FADZLI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Computer Systems & Networking)

Faculty of Computing
UNIVERSITI MALAYSIA PAHANG

JULY 2023

ACKNOWLEDGEMENTS

I am deeply grateful to my supervisor, Ts. Dr. Zuriani Binti Mustaffa, for giving me the opportunity to work with her as a supervisee and accepting my proposed topic of Sales Prediction Using Artificial Neural Network. I am very grateful for the guidance and support that Dr. Zuriani provided throughout my research. I am truly fortunate to have her as my mentor. Dr. Zuriani willingness to share her knowledge have been instrumental in the successful completion of this research.

I would also like to extend my gratitude to my family members for their unwavering support and companionship during this project, especially during the times when I felt stressed or faced difficulties while working on this research. Their constant encouragement, understanding, and belief in my abilities have been a source of motivation during both the highs and lows of this journey. My family willingness to lend a helping hand during times of stress and difficulty have been invaluable, and I am forever grateful for their presence in my life.

Finally, I want to extend my sincere thanks to my friends for their unwavering support and advice throughout this research. Their willingness to lend an ear, provide insightful feedback, and share their knowledge have been pivotal in overcoming challenges and completing this research. Their presence has not only lightened the burden but has also made this research journey an enriching and enjoyable experience.

ABSTRAK

Dalam penyelidikan ini, model Artificial Neural Network (ANN) telah dibuat untuk ramalan jualan. Penyelidikan ini mencadangkan penggunaan ANN untuk ramalan jualan sebagai alternatif kepada teknik ramalan konvensional. Objektif utama penyelidikan ini adalah untuk mencipta ANN yang boleh cuba meramalkan jualan masa depan dengan tepat menggunakan data jualan lalu. Model ANN yang dicadangkan telah dilatih menggunakan algoritma backpropagation. ANN terdiri daripada beberapa lapisan neuron. Dataset yang digunakan dalam kajian ini dikumpulkan daripada laman web Kaggle. Kesimpulannya, model ANN yang dicadangkan boleh membantu pemilik perniagaan dalam membuat keputusan yang lebih baik dengan meramalkan jualan masa hadapan, mengenal pasti corak dan arah aliran dalam set data.

ABSTRACT

In this research, an Artificial Neural Network (ANN) model was developed for the sales prediction. This research proposed the use of ANN for forecasting sales as an alternative to conventional forecasting techniques. The main objective of this research was to create an ANN model that can try to accurately forecast future sales using historical sales data. The proposed ANN model was trained using a backpropagation algorithm. The ANN consisted of multiple layers of artificial neurons. The sales dataset used in this study was collected from the Kaggle website. In conclusion, the proposed ANN model can assist business owners in making better decisions by forecasting future sales, identifying patterns and trends in the dataset.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Nowadays, there are an increasing number of Malaysians that are starting and running their own business. The advancement of technology like the internet and e-commerce platform have greatly reduced the barriers of entry for starting a business. This coincides with the rise in popularity of e-commerce platforms like Shoppe and Lazada among Malaysian customers. Many customers prefer to buy products at these e-commerce platforms because of convenience and time saving (Pillemer,2020). With this type of opportunity, many Malaysians jump start their business career since e-commerce platforms allow them to sell their product online without the need of a physical storefront. Additionally, social media has made it easier for business owners to promote their product and engage with their customers.

As a business owner, there are many important business decisions that they need to carefully consider to grow their own business and bring more revenue. Sales forecast is one of them. It is of the utmost importance for the business owner to have an accurate sales forecast. Sales forecasts will allow business owners to identify and plan their resources (inventory, promotion and cash flow) required for meeting the actual demand (Rohn,2022). Business owners can make informed business decisions based on their sales prediction (Mahalingam,2022). They can also spot potential issues early like sales not reaching the intended target and make the necessary adjustment to reach the sales target.

Historical forecasting method is one of several traditional forecasting methods that is used by Malaysian business owners to predict future sales. It involves searching past sales data that is similar to the current sales data and predicting that the sales result will overlap (Stojanovic,2022). Given the similarity of the sales data, it is possible to get reasonably precise sales forecasts. The disadvantage of this type of forecasting method is even if the sales data is similar, it is not exactly the same. This can sometimes cause inaccurate sales predictions. We can solve the traditional forecasting method problem by introducing an alternative such as implementing an Artificial Neural Network to help predict future sales.

1.2 Problem Statement

Currently, many Malaysian business owners use the traditional forecasting method to try and predict future sales. One of the traditional forecasting methods used by them is intuitive forecasting. As the name suggests, intuitive forecasting refers to when the business owner uses their intuition and experience to do sales forecasts (Stojanovic,2022). This forecasting method requires good intuition and years of personal experience managing the business. This forecasting method has a significant drawback as it relies heavily on the abilities of the individual involved and not everyone has good intuition or years of personal experience.

The other traditional forecasting method used by Malaysian business owners is the time series forecasting method such as moving average forecast. Moving average forecast calculates the average of a fixed number of past sales observations to estimate future sales. This method assumes that future sales will follow a similar pattern as observed in the past. While the moving average approach is simple to implement, it may struggle to capture more complex patterns and changes in sales trends, making it less accurate in dynamic business environments.

Traditional forecasting methods also struggle with handling big data due to their limited capacity to handle large volumes and variety of data. These methods often rely on simplistic models and assumptions that may not adequately capture the complexity and dynamics of big data sets. Furthermore, traditional methods typically require manual data preprocessing and feature selection, which can be time-consuming and error-prone when dealing with massive amounts of data. Overall, the conventional forecasting methods are ill-equipped to effectively harness the potential insights contained within big data, necessitating the development of more advanced techniques and tools.

Due to the disadvantage of traditional forecasting methods, this study proposes using Artificial Neural Networks (ANN) to do sales prediction. ANN has a wide range of real world applications and forecasting is one of them. This is because ANN are good at capture complex, nonlinear relationships between input variables and sales outcomes. By recognizing patterns, handling multiple variables, and efficiently processing large datasets, ANN can provide an accurate sales forecasts.

1.3 Objective

The objective of this research is:

- To study the Artificial Neural Network (ANN) for sales prediction.
- To develop a sales prediction model using ANN.
- To evaluate the result of this forecasting model.

1.4 Scope

The scope for this research is:

- The ANN model will be develop using the Python language at the Google Colab platform.
- Several important library will be used to help us in our research such as pandas, numpy, matplotlib, scikit-learn, tensorflow and keras.
- The dataset that we used in this study is the bakery sales dataset which is downloaded from Kaggle website.
- The dataset contains sales data for 5111 days, from 2006 to 2019 but this research only use 3 years of sales data (2006-2008).
- The data frequency is daily.
- The sales dataset will be divided into training set (70%), and testing set (30%).
- This study will run the algorithm multiple times.
- The predicted sales result will be evaluated and compared to the real sales data.

1.5 The Significant of this Research

- Business owners

Business owner can rely on accurate sales forecasting for effective decision-making. By utilizing ANN models for sales prediction, they can enhance their own inventory management, production planning, and resource allocation. This enables them to optimize their operations, reduce costs, and improve overall profitability.

- Marketing and Sales Managers

Sales and marketing professionals can benefit from this research by leveraging ANN-based sales prediction models to identify patterns, trends, and factors influencing consumer behavior. This information can assist them in developing targeted marketing campaigns, optimizing pricing strategies, and identifying potential customer segments for better sales performance.

- Financial Analysts and Investors

Sales predictions play a crucial role in financial analysis and investment decisions. Analysts and investors can utilize accurate sales forecasts derived from ANN models to assess a company's growth potential, evaluate its financial performance, and make informed investment choices. This research can enhance their ability to predict future revenue streams and assess the financial viability of businesses.

1.6 Thesis Organization

In this thesis organization, it will consist of five chapters. The first chapter contains the introduction of the research title. It also includes the introduction, problem statement, objective, scope, significant of this research and thesis organization. The second chapter is a review of the literature. It will focus on past research that has been done regarding sales forecasts. The third chapter is methodology. This is the section where the methodology used in modelling the research project is illustrated. Chapter four is review on the research result and discussion. This chapter disscuss the actual implementation of the research project and its result. Finally, chapter five is the conclusion of this research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Chapter 2 will focus on past research that has been done regarding sales forecasts. Each research will have different forecasting methods so that we can easily identify each forecasting method's strength and weakness.

2.2 Previous Research Work

2.2.1 Annual Automobile Sales Prediction Using ARIMA Model

The authors of this research paper, Sana Prasanth Shakti, Mohan Kamal Hassan, Yang Zhenning, Ronnie D. Caytiles and Iyengar N.Ch.S.N. used the ARIMA model to do sales forecasts (Shakti et al.,2017). ARIMA technique is applied to time series data. Data that has been collected over a constant period of time is known as a time series. The Mahindra Tractors sales dataset was used by the authors. The dataset includes information on tractor sales for the ten-year period (2003-2014), which is used to forecast sales for the following five years.

To construct a time-series model using ARIMA, the authors need to identify variables of p, d, and q. The meaning of those variables is p represents Autoregressive (AR), d represents Integrated (Stationary) and q represents Moving Average (MA). The formula that authors used for autoregressive, moving average and stationary series is shown below.

Autoregressive (AR):

$$y_t = c + \sum_{i=1}^{p} \varphi_i y_{t-i} + \varepsilon_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \ldots + \varphi_p y_{t-p} + \varepsilon_t \tag{2.1}$$

Moving average (MA):

$$y_t = \mu + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t = \mu + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q} + \varepsilon_t \tag{2.2}$$

Stationary series (Integrated):

$$\begin{aligned} &(1st\ order)\ \ \nabla x_t = (1 - B)\, x_t = x_t - x_{t-1} \\ &(2nd\ order)\ \ \nabla^2 x_t = (1 - B)^2 x_t = x_t - 2x_{t-1} + x_{t-2} \end{aligned} \tag{2.3}$$

Figure 2.1 below shows the flow diagram of the ARIMA model that the authors implement in this study.

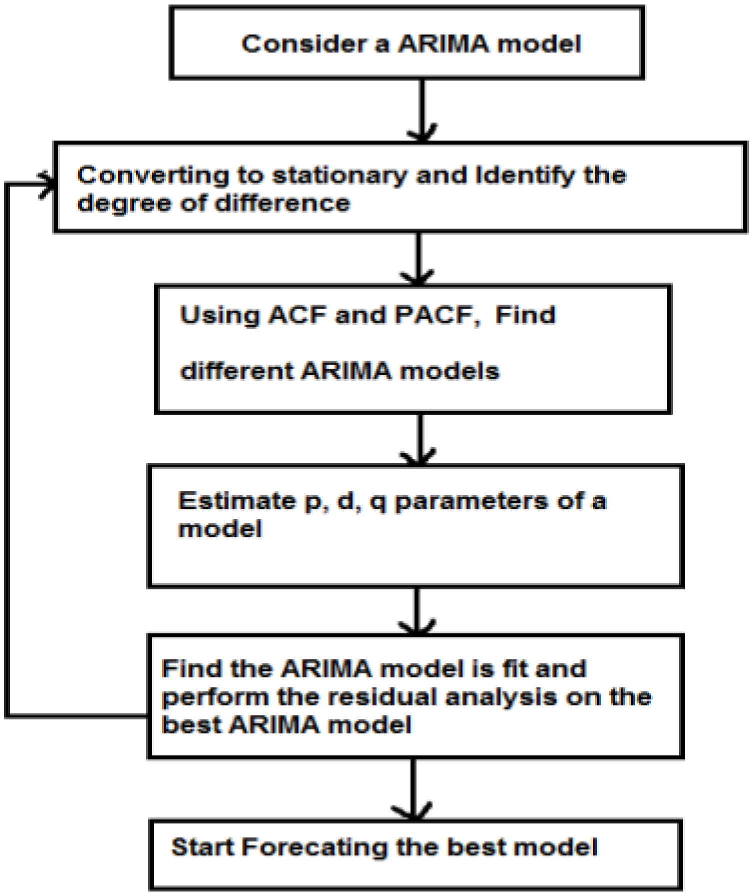

Figure 2.1 Flow diagram of ARIMA model

Before constructing a model, it is beneficial to preprocess the raw data to improve its quality and make predictions more accurate. This preprocessing can include data cleaning, which involves identifying and correcting any missing values in the dataset, and data transformation which optimizes the performance of the algorithms used. In this case, the authors used Z-score normalization to convert the raw data into a usable format. The formula used is shown below.

Z-score normalization:

$$V' = \frac{(V - \overline{A})}{\sigma_A} \tag{2.4}$$

This study uses the programming language R to implement the ARIMA model. The first step is to convert the data into a stationary form. This is achieved by finding the difference in the mean of the sales numbers. Then, the transformed data is log transformed to adjust the variance, resulting in final data that has both a log transformed mean and variance. Next, ACF and PACF graphs are plotted to identify the potential AR and MA models. Figure 2.2 & 2.3 below show the ACF and PACF.

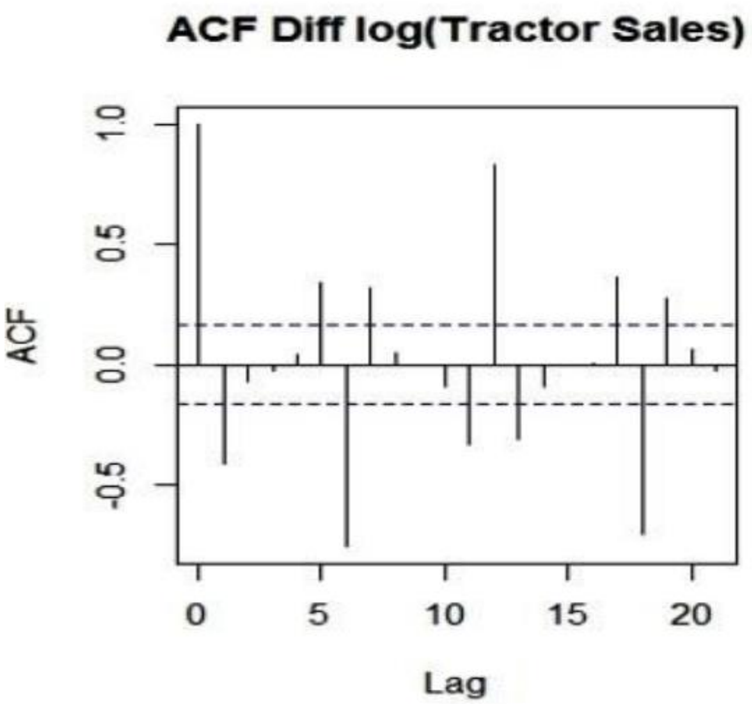

Figure 2.2 ACF graph for AR model

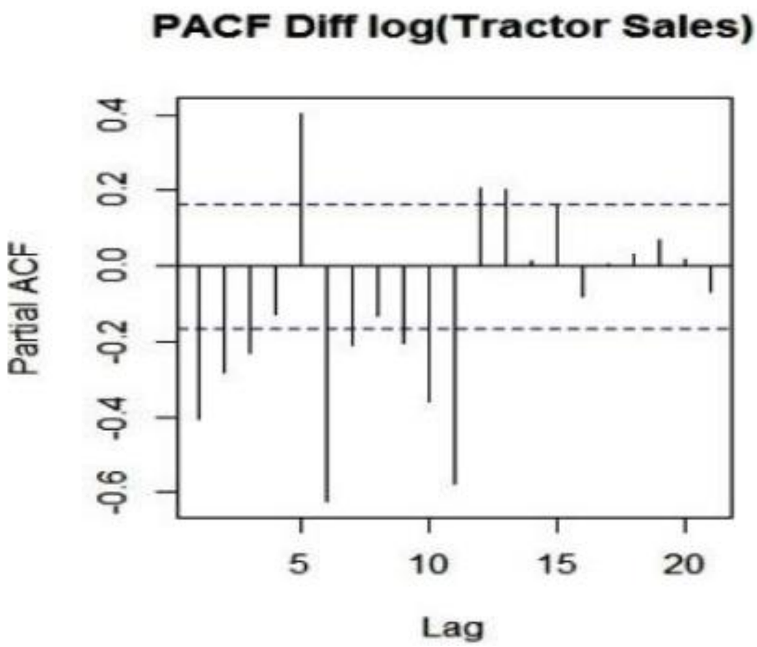

Figure 2.3 PACF graph for the MA model

To identify the best fitting ARIMA model for sales predictions, the authors analyzed the two ACF and PACF graphs. The selection of the optimal model is determined by the AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) values, where the model with the lowest AIC is considered the best fit. The final graph is plotted for the best fit ARIMA model of number of sales in the following years. Figure 2.4 below shown the output with the forecasted sales of tractors in blue colour. The sales prediction covers the period from 2015 to 2018 which is shown below.

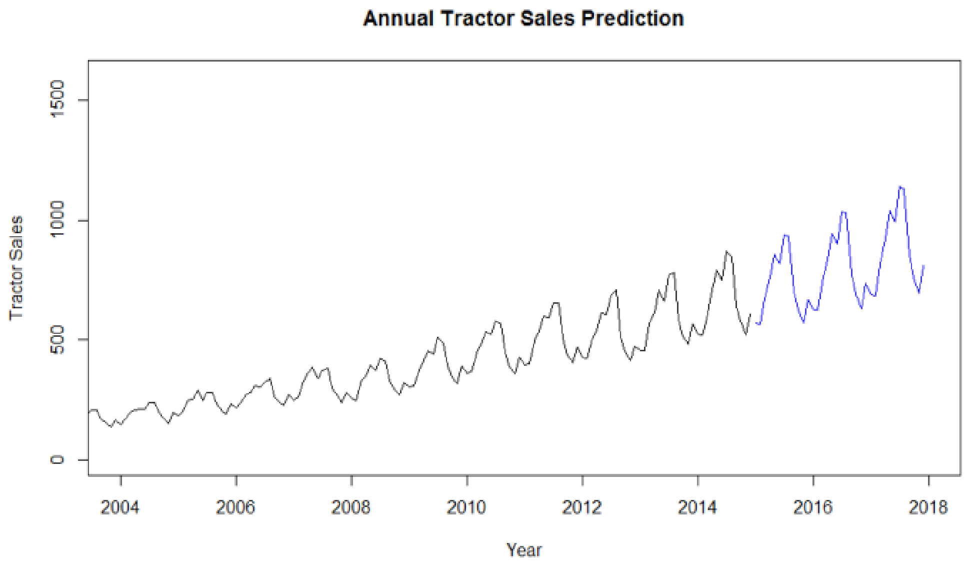

Figure 2.4 ARIMA model prediction up to 2018

Forecasting sales several years into the future is a challenging task. A more realistic approach is to use a short-term forecasting model, such as for a few business quarters or a year, which has a higher chance of being accurate. For long-term forecasting, it is advisable to regularly re-evaluate the model at intervals of around six months to incorporate any new information as it becomes available. Figure 2.5 below shows the prediction by monthly.

```
$pred
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
2015 2.754168 2.753182 2.826608 2.880192 2.932447 2.912372 2.972538 2.970585
2016 2.796051 2.795065 2.868491 2.922075 2.974330 2.954255 3.014421 3.012468
2017 2.837934 2.836948 2.910374 2.963958 3.016213 2.996138 3.056304 3.054351
          Sep      Oct      Nov      Dec
2015 2.847264 2.797259 2.757395 2.825125
2016 2.889147 2.839142 2.799278 2.867008
2017 2.931030 2.881025 2.841161 2.908891

$se
            Jan        Feb        Mar        Apr        May        Jun        Jul
2015 0.01603508 0.01866159 0.02096153 0.02303295 0.02493287 0.02669792 0.02835330
2016 0.03923008 0.04159145 0.04382576 0.04595157 0.04798329 0.04993241 0.05180825
2017 0.06386474 0.06637555 0.06879478 0.07113179 0.07339441 0.07558934 0.07772231
            Aug        Sep        Oct        Nov        Dec
2015 0.02991723 0.03140337 0.03282229 0.03418236 0.03549035
2016 0.05361850 0.05536960 0.05706700 0.05871534 0.06031866
2017 0.07979828 0.08182160 0.08379608 0.08572510 0.08761165
```

Figure 2.5 Sales prediction by monthly

2.2.2 Daily Sales Forecasting for Grapes by Support Vector Machine

Another research on sales forecast was proposed by Qian Wen, Weisong Mu, Li Sun, Su Hua, and Zhijian Zhou. The sales forecast was predicted by using a Support Vector Machine (SVM). The authors are interested in predicting sales for three types of grapes(Qian et al., 2014). The name of the three grapes is XiaoMiFeng, JuFeng and MeiGuiXiang. The dataset used in this research was taken from a fruit supermarket called Fu Man Jia. The dataset includes information from July 2011 to the end of September 2012, as grapes were harvested extensively during this period. Library for Support Vector Machines (LIBSVM) is used to implement the SVM. Figure 2.6 below shows the framework of forecasting that is used in this study. The framework starts from new data and history data until the output is obtained at the end which is shown below.

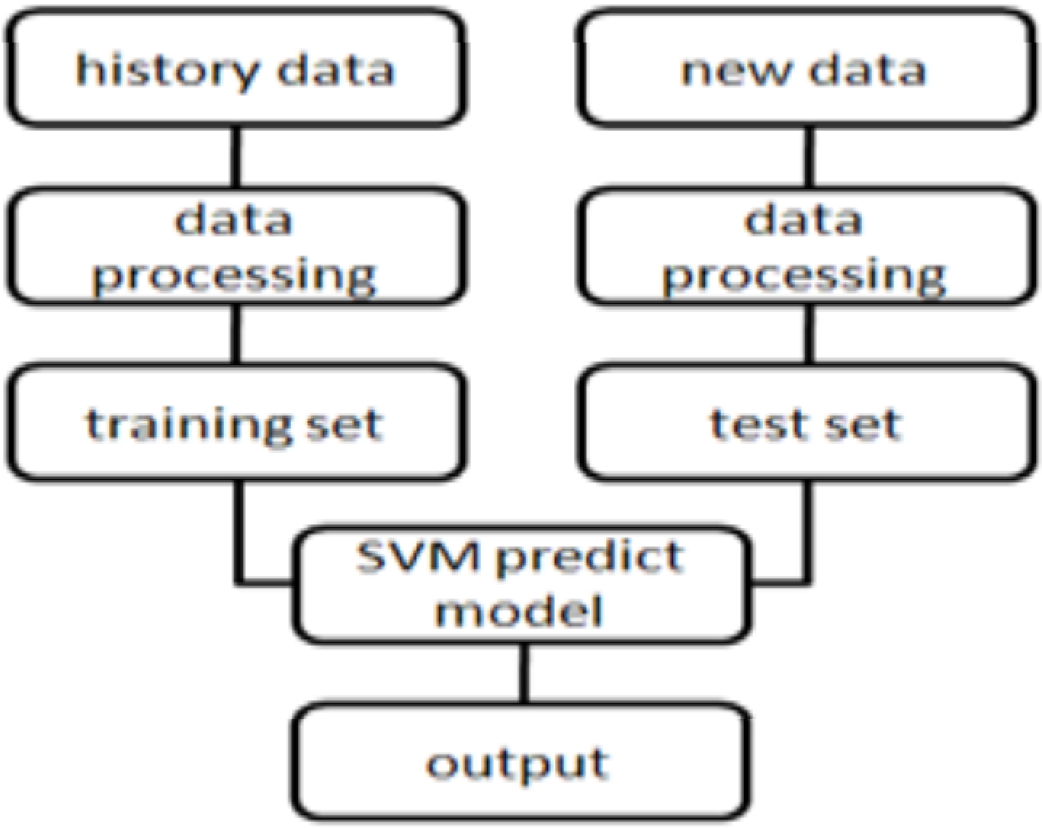

Figure 2.6 Framework

This study utilised the radial basis function (RBF) as the kernel function, as illustrated below.

Kernel function:

$$K\left(x, x_j\right) = exp\left\{-\parallel x_j - x \parallel^2 \div \delta^2\right\} \tag{2.5}$$

The standard Support Vector Regression (SVR) model is shown below. Where ϕ (x) is in the high-dimensional feature space, which is non-linearly mapped from the input space x.

Support Vector Regression:

$$y = f(x) = (w \times \phi(x)) + b \tag{2.6}$$

For the ε-SVR model, authors select the ε-insensitive loss function as its error measurement. Based on the ε-insensitive loss function, the decision function of the ε-SVR model is shown below.

Decision function of the ε-SVR model:

$$f(x) = \sum\left(\overline{\alpha_i} - \overline{\alpha_i^*}\right) K(x, x_i) + \overline{b}$$

$$\overline{b} = \begin{cases} y_j - \sum_{i=1}^{n}\left(\overline{\alpha_i} - \overline{\alpha_i^*}\right) K\left(x_i, x_j\right) + \varepsilon & \overline{\alpha_i} \in (0, C) \\ y_j - \sum_{i=1}^{n}\left(\overline{\alpha_i} - \overline{\alpha_i^*}\right) K\left(x_i, x_j\right) - \varepsilon & \overline{\alpha_i^*} \in (0, C) \end{cases} \tag{2.7}$$

For the LS-SVR model, the authors select a quadratic loss function as its loss function. The equation for the quadratic loss function is presented below.

Quadratic loss function:

$$L\left(d_i, y_i\right) = \sum_{i=1}^{n}\left(d_i - y_i\right)^2 \tag{2.8}$$

By combining the above loss function and RBF kernel function we get the LS-SVR decision function which is shown below.

Decision function of the LS-SVR model:

$$y = f(x) = \sum_{i=1}^{n} \overline{\alpha}_i K\left(x_i, x\right) + \bar{b} \tag{2.9}$$

The output of the support vector machine model will be the predicted sales. The input value that will be inserted to the support vector machine model will be shown below.

Input variables of grape sale forecasting:

$$X = \left(S_{d-1}, S_{d-7}, W_d, W_{d-1}, P_d, P_{d-1}, T_d, T_{d-1}\right) \tag{2.10}$$

Where,

S_{d-1} – Sales quantity at the day before the forecasting day

S_{d-7} – Sales quantity at the day 7 days before the forecasting day

W_d – Type of date at the forecasting day (workday or weekend)

W_{d-1} – Type of date at the day before forecasting day (workday or weekend)

P_d – Sale price of grape at the forecasting day

P_{d-1} – Sale price of grape at the day before forecasting day

T_d – The weather condition of the forecasting day

T_{d-1} – The weather condition of the day before forecasting day

The data used in this study includes information about weather conditions and holidays, which are not numerical values. Therefore, the authors have quantified these data points as shown below. Table 2.1 will show the quantified value of weather conditions.

Quantified value of date:

$$W_d = \begin{cases} 0 & monday,\ tuesday,\ wednesday,\ thursday,\ friday \\ 1 & saturday,\ sunday \end{cases} \tag{2.11}$$

Weather	value
sunny	1
cloudy	0.9
overcast	0.8
Light rain	0.7
moderate rain	0.5
Showery rain	0.4
downpour	0.2

Table 2.1 Quantified value of weather condition

For data preprocessing, data normalization is used in order to avoid data overflow. It is also to eliminate singular values and noise in the raw data. The data is then divided into two sets, the training set and the testing set. The training set is inputted into the SVM model and the parameters are adjusted to their optimal values. Finally, forecasting is conducted by inputting the testing set into the trained SVM model, resulting in predicted sales values.

In order to verify the validity of the prediction performance of the SVM method, the authors use absolute error as the statistical metrics which is shown below.

Day absolute error:

$$DAE = \left| \frac{\left(d(i) - f(x_i) \right)}{d(i)} \right| \times 100\% \tag{2.12}$$

The result of the predicted sales of three grapes will be shown below at Figure 2.7, 2.8, & 2.9 and Table 2.2, 2.3, & 2.4. It will show the comparison with the real sales data and also other forecasting methods predicted results.

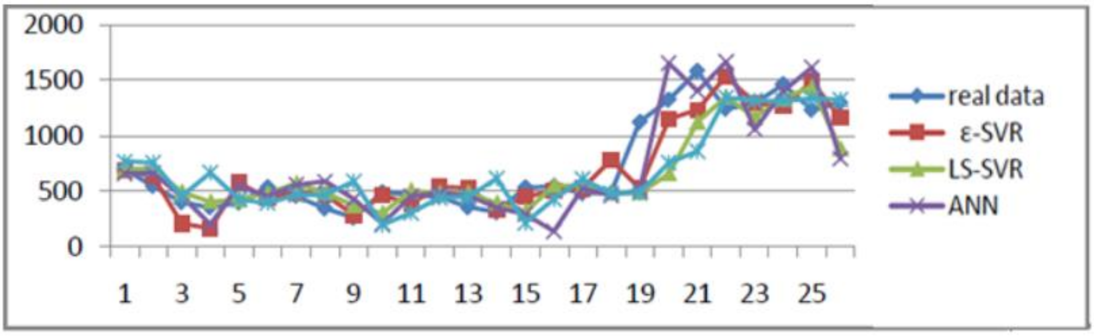

Figure 2.7 The result of sale forecasting for XiaoMiFeng

method	Average relative error	Maximum relative error	Time spent(s)
ε-SVR	0.2124	0.5395	50.32
LS-SVR	0.2229	0.5632	185.81
ANN	0.2890	0.7440	61.12
DR	0.2508	1.2559	5.02

Table 2.2 Comparison of real data and forecasting result for XiaoMiFeng

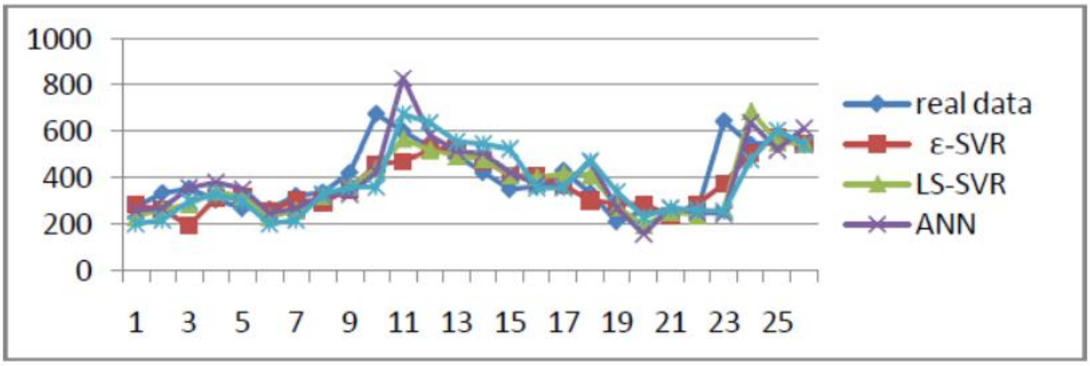

Figure 2.8 The result of sale forecasting for JuFeng

Method	Average relative error	Maximum relative error	Time spent(s)
ε-SVR	0.1407	0.4495	64.38
LS-SVR	0.1528	0.5938	238.27
ANN	0.1896	0.6223	70.30
DR	0.1467	0.6064	6.57

Table 2.3 Comparison of real data and forecasting result for JuFeng

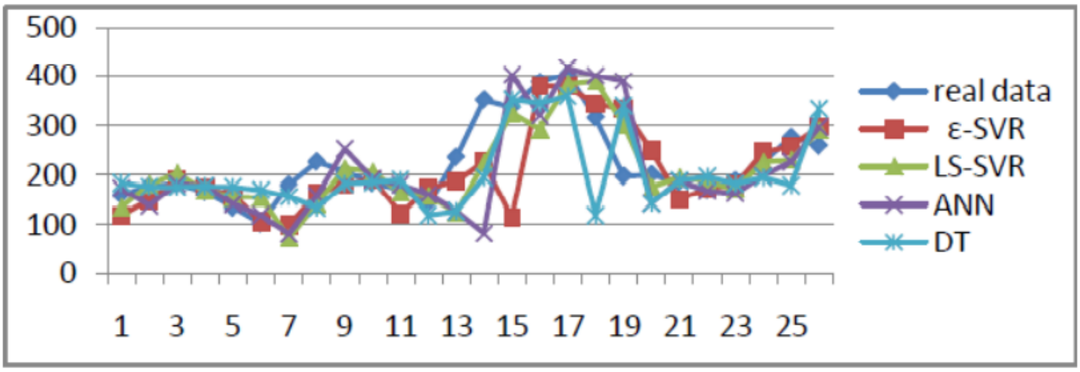

Figure 2.9 The result of sale forecasting for Meiguixiang

Method	Average relative error	Maximum relative error	Time spent (s)
ε-SVR	0.1926	0.6777	48.16
LS-SVR	0.1845	0.5957	196.05
ANN	0.2163	0.9715	50.34
DR	0.1725	0.7113	6.13

Table 2.4 Comparison of real data and forecasting result for Meiguixiang

Based on the result above, the SVM method achieves better results compared to the other forecasting method. The predicted sales result for the three grapes is closer to the real sales data and has minimal forecasting error.

2.2.3 A Fuzzy Logic Based Approach towards Sales Forecasting: Case Study of Knit Garments Industry

The last research that we will review regarding sales forecasts is by Md Mamunur Rashid, Md Rubel Khan, Sourav Kumar Ghosh. The sales forecast of ready-made garments was predicted by using Fuzzy Logic (Rashid et al., 2020). An algorithm based on a Fuzzy Inference System (FIS) was developed to predict future sales of ready-made garments using various input variables. In this study, MATLAB Fuzzy Logic Toolbox will be used to predict using Fuzzy Logic. The sales dataset is taken from the Bangladesh garments industry, DBL Group. The dataset contains information such as number of ready-made garments sold and its year (from 2003-2012). Table 2.5 below shows the dataset that will be used in this study.

Year	No. of RMG sold (Million)	Year	No. of RMG sold (Million)
2003	0.1	2008	05
2004	0.3	2009	10
2005	0.6	2010	14
2006	01	2011	18
2007	03	2012	20

Table 2.5 Sales data of previous years (2003-2012)

A Fuzzy Logic system is composed of four main elements: the fuzzifier, rules, inference engine, and defuzzifier. The process starts with the fuzzification, where input data is transformed into a fuzzy set using fuzzy linguistic variables, fuzzy linguistic terms, and membership functions. Then, inferences are made based on a set of rules. Finally, the fuzzy output is converted into a precise output by using membership functions during the defuzzification stage. Figure 2.10 below shows the fuzzy sales controller.

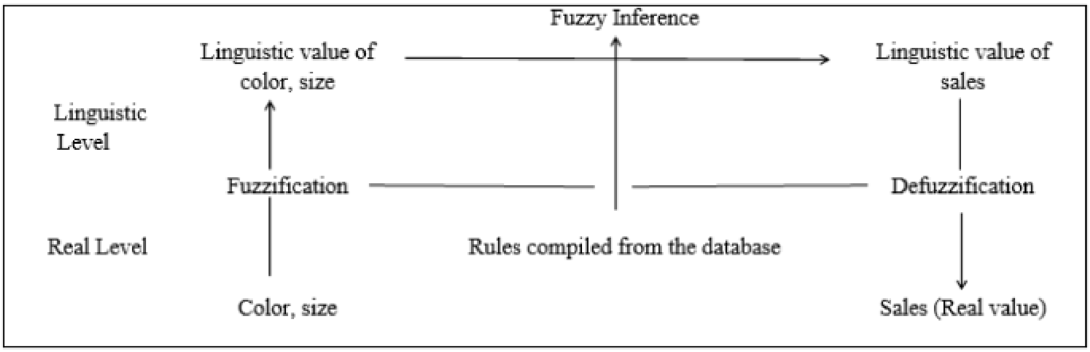

Figure 2.10 Fuzzy sales controller

Fuzzy Set represents a graded membership over the interval [0, 1], where the membership function is the extent to which the variable is considered to be part of the fuzzy set. The sales fuzzy controller is composed of three parts: fuzzification, fuzzy inference, and defuzzification. Figure 2.11 below shows the algorithm of fuzzy logic that will be used in this study.

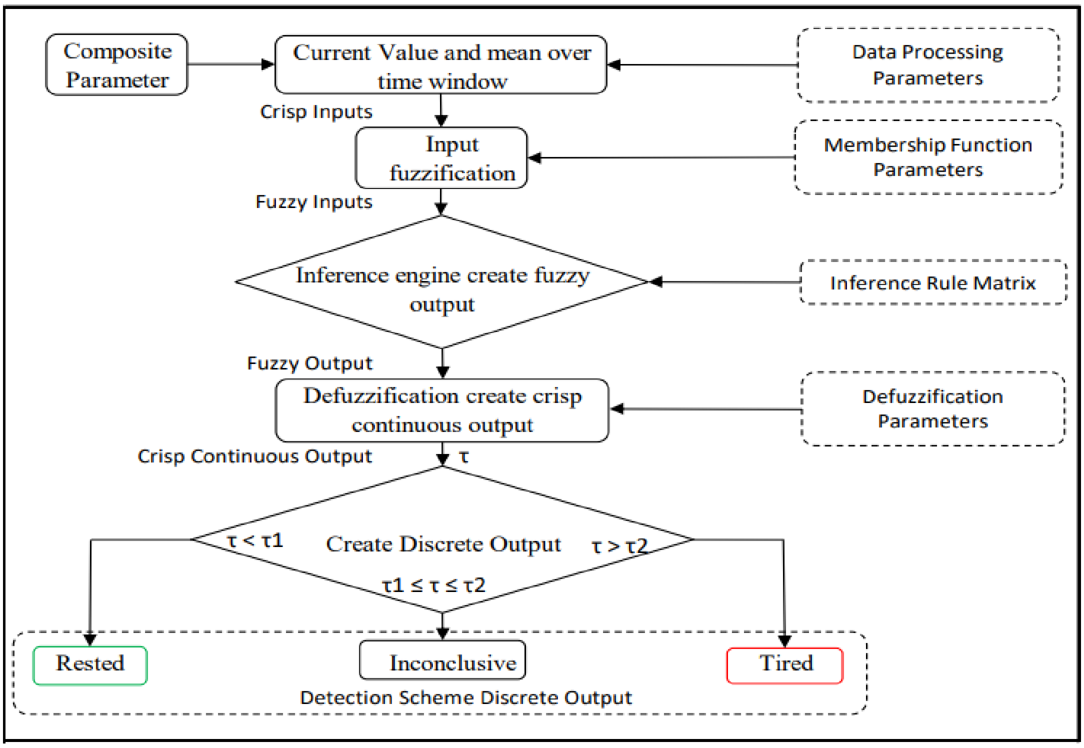

Figure 2.11 Algorithm of Fuzzy Logic

The accuracy of a fuzzy inference system is dependent on how closely the input data represents actual events. The membership function of the variables should be comprehensive enough to include all the characteristics of the real-world scenario, thus capturing all the uncertainties present in the system during formulation. In this case, 16 factors were considered as inputs, and the number of predicted ready-made garments sold was used as the output. Table 2.6 below shows the sales forecast of ready-made garments obtained from the Fuzzy Logic model.

Year	Season	Seasonal sales forecast of RMG by Fuzzy Model (Million)
2013	January	21.3718
	July	21.9919
2014	January	22.0436
	July	22.4474
2015	January	24.1320
	July	25.2533
2016	January	26.4923
	July	26.8662
2017	January	27.3248
	July	28.8625
2018	January	29.4988
	July	29.7288

Table 2.6 Sales forecast of RMG obtained from fuzzy model

To calculate the seasonal factor, the authors use a linear regression analysis that is shown below. Table 2.7 is the calculation of seasonal factor.

Linear regression analysis:

$$\widehat{\beta}_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \; And \;\; \widehat{\beta}_0 = \bar{y} - \widehat{\beta}_1 \bar{x} \qquad \beta_1 = 1.269, \beta_0 = 10.799 \tag{2.13}$$

Year	Season	Actual No of RMG sold(Million)	From Trend Equation Y=10.799+1.269X	Ratio of Actual/Trend
2010	January	12	12.068	0.944
	July	14	13.337	1.0497
2011	January	16	14.606	1.095
	July	18	15.857	1.135
2012	January	19	17.144	1.108
	July	20	18.413	1.086

Season	2010	2011	2012	Seasonal Factor
January	0.944	1.095	1.108	1.066
July	1.0497	1.135	1.086	1.090

Table 2.7 Calculation of seasonal factor

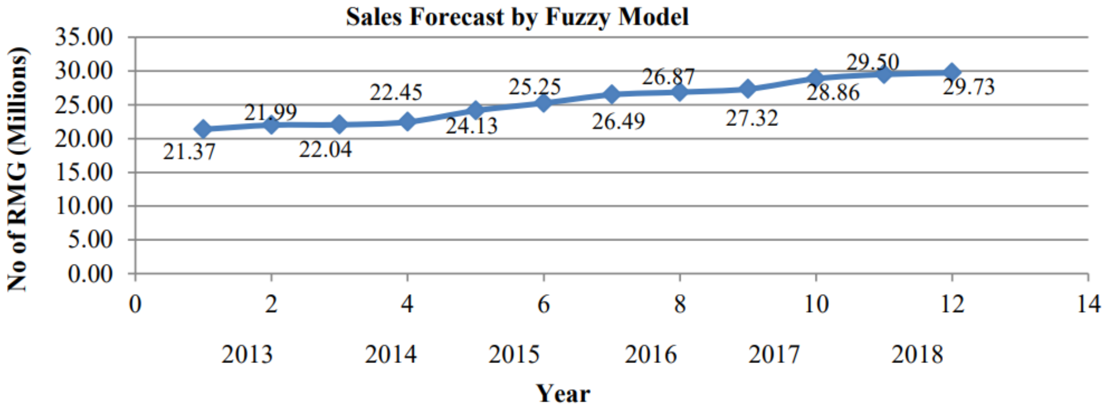

Figure 2.12 Sales forecast by fuzzy model

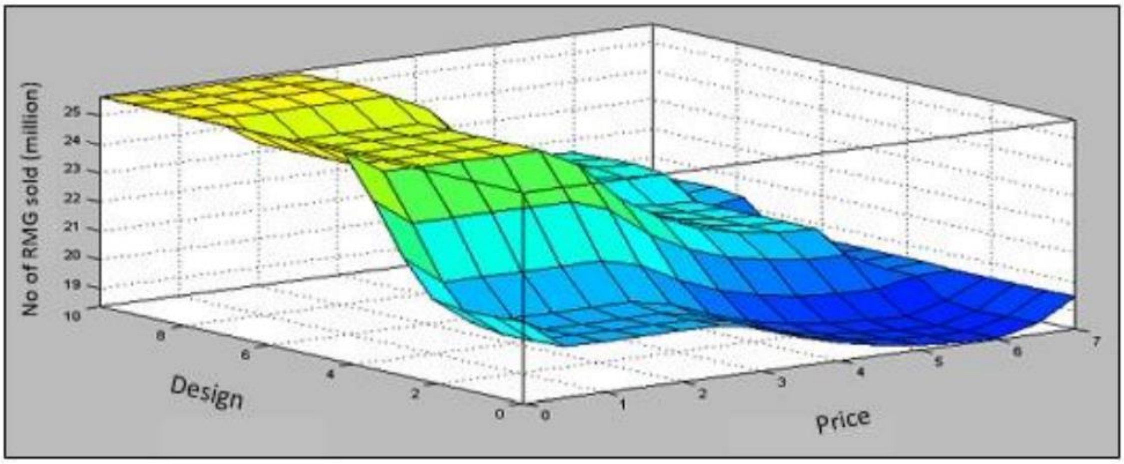

Figure 2.13 Sales forecast of RMG relating with design and price

Figure 2.12 & 2.13 above show the predicted sales by fuzzy model and also its relationship with design and price. The authors also do another traditional forecast to compare its result with the fuzzy model result which is shown in Figure 2.14 and Table 2.8 below.

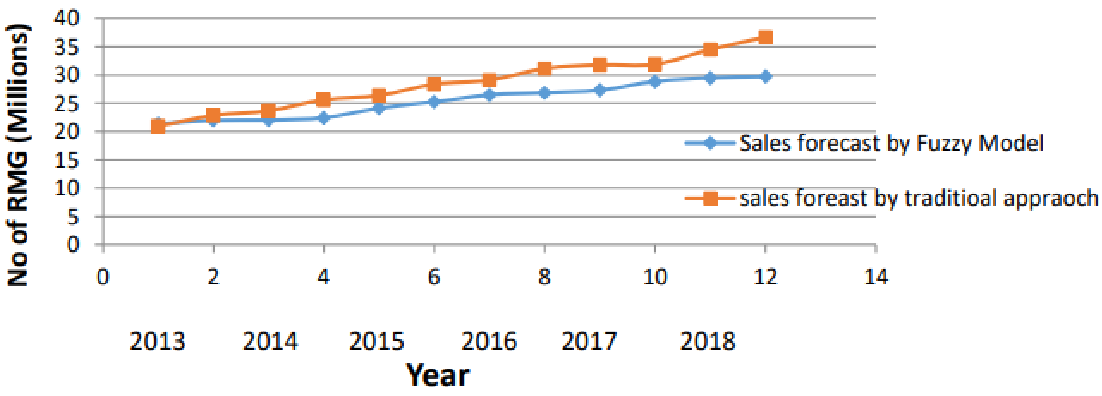

Figure 2.14 Comparison of sales forecast with fuzzy model and traditional approach

Year	Season	Seasonal sales forecast of RMG by Fuzzy Model (Million)	Traditional Sales Forecast of RMG (Million)	Traditional Sales Forecast of RMG with Seasonal Factor (Million)
2013	January	21.3718	20.98	22.365
	July	21.9919	22.837	24.892
2014	January	22.0436	23.687	25.250
	July	22.4474	25.603	27.907
2015	January	24.1320	26.392	28.134
	July	25.2533	28.369	30.922
2016	January	26.4923	29.098	31.018
	July	26.8662	31.136	33.938
2017	January	27.3248	31.803	33.902
	July	28.8625	31.902	34.773
2018	January	29.4988	34.509	36.787
	July	29.7288	36.669	39.969

Table 2.8 Sales forecast compared between fuzzy model and traditional forecast

The authors of the study found that the forecasting error of the Fuzzy model developed was very low, at only 3.605%. The results demonstrate that the fuzzy model provides more accurate demand predictions by accounting for inherent uncertainties and with minimal error.

2.3 Comparison of Previous Research Works

Contents	**Research 1 (ARIMA)**	**Research 2 (SVM)**	**Research 3 (Fuzzy Logic)**	**Proposed Research (ANN)**
Research	Annual Automobile Sales Prediction Using ARIMA Model	Daily Sales Forecasting for Grapes by Support Vector Machine	A Fuzzy Logic Based Approach towards Sales Forecasting: Case Study of Knit Garments Industry	Sales Prediction Using Artificial Neural Network
Author	Sana Prasanth Shakti, Mohan Kamal Hassan, Yang Zhenning, Ronnie D. Caytiles and Iyengar N.Ch.S.N.	Qian Wen, Weisong Mu, Li Sun, Su Hua, and Zhijian Zhou	Md Mamunur Rashid, Md Rubel Khan, Sourav Kumar Ghosh	Ilham Ariff Bin Mohammad Fadzli
Objective	Sales Forecast of tractor using Autoregressive Integrated Moving Average (ARIMA)	Sales Forecast of grape using Support Vector Machine (SVM)	Sales Forecast of ready-made garment using Fuzzy Logic	Sales forecast of bakery product using Artificial Neural Network (ANN)
Technique	Autoregressive Integrated Moving Average (ARIMA)	Support Vector Machine (SVM)	Fuzzy Logic	Artificial Neural Network (ANN)
Data	Sales dataset from Mahindra Tractors	Sales dataset from Fu Man Jia	Sales dataset from DBL Group	Sales dataset from Kaggle website
Limitation	It assumes that the underlying time-series data is stationary, which may not always be the case	SVM algorithms are not suitable for large data sets because of the high computational	The rule is based on the predefined rules and if the rules are flawed, the	ANN requires a lot of data to train the model and can be sensitive to overfitting

		cost of training and prediction	result will be inaccurate	

Table 2.9 Comparison of previous research works

Table 2.9 above show the comparison between previous works. The proposed research using ANN is also added to highlight the comparison of this research to the past research.

From the past research, we can highlight the advantages and disadvantages with each forcasting technique for sales prediction. The advantage of using ARIMA is that it is a widely used and well-established method for time-series forecasting, but it can be difficult to determine the optimal values of the p, d, and q parameters, which are critical to the model's performance. Next, SVM is effective when there is a clear division between classes, but can struggle with noisy or imbalanced data. Fuzzy Logic is good at accepting the imprecise input information, but the accuracy of these prediction can be compromised since they rely on imprecise data and inputs. Finally, ANNs are able to learn from examples, making them well suited for unsupervised learning and semi-supervised learning. However, ANN is data-dependency as it require huge amount of data for the training phase, if not then the predicted result will be not accurate.

2.4 Summary

Each research used a different forecasting method to predict future sales. The result of the different forecasting method will vary depending on the dataset. Some forecasting methods are more suitable on certain datasets while other forecasting methods will perform better in different datasets. There is no perfect forecasting method that will give predicted results exactly as the true sales.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter 3 will discuss the methodology that is used in the sales prediction by using Artificial Neural Networks (ANN). This chapter will cover the Research Framework, Research Requirement, Proposed Design, Data Design, Proof of Initial Concept, Testing Plan and Potential Used of Proposed Solution.

3.2 Research Framework

Figure 3.1 below shows the research framework that will be implemented in this study.

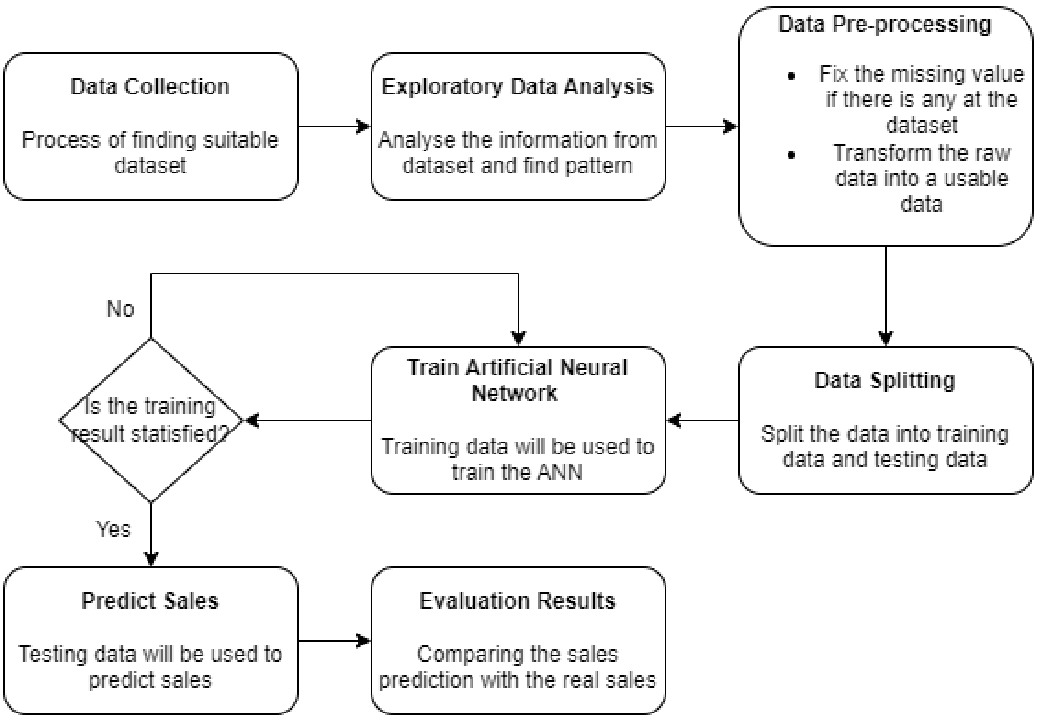

Figure 3.1 Research Framework

3.2.1 Data Collection

The dataset that will be used is the bakery sales dataset which is downloaded from Kaggle website. This dataset is open to the public and available to download. The sales dataset contains daily sales of bakery product from 2006-2019. The dataset consists of 8 variables that are date, weekday, cakes, pies, cookies, smoothies, coffee and promotion. Our research will use 3 years of sales data from the dataset starting from 2006-2008. Table 3.1 below shows the sample of the dataset that will be used in this study.

	B	C	D	E	F	G	H	I
	Date	weekday	cakes	pies	cookies	smoothies	coffee	promotion
0	1/1/2006	Sunday	45	41	50	19	73	promotion
1	2/1/2006	Monday	48	18	18	44	5	none
2	3/1/2006	Tuesday	1	40	99	41	8	none
3	4/1/2006	Wednesday	4	10	15	58	95	none
4	5/1/2006	Thursday	4	44	67	71	20	promotion
5	6/1/2006	Friday	40	27	104	39	76	promotion
6	7/1/2006	Saturday	10	21	10	26	5	none
7	8/1/2006	Sunday	20	58	59	74	68	promotion
8	9/1/2006	Monday	22	5	28	61	54	none
9	10/1/2006	Tuesday	37	43	76	72	67	promotion
10	11/1/2006	Wednesday	24	3	60	20	10	none

Table 3.1 Sample of the bakery sales dataset

3.2.2 Exploratory Data Analysis

We will apply Exploratory Data Analysis (EDA) to gain a deeper understanding of the dataset. This will include identifying and analysing the variables and attributes in the dataset. Additionally, EDA will help us discover valuable information such as patterns and trends in the data (Patil,2018). This process will also aid non-technical users in understanding the dataset through the use of visual aids.

3.2.3 Data Pre-processing

The sales dataset that we obtained from the Kaggle website was just the raw data and not yet suitable to be used in ANN. It was critical for us to preprocess the raw data so that the sales prediction using ANN will be accurate. Therefore, the raw data will go through the preprocessing stage to ensure the data in the dataset are usable. We start with the data cleaning, by fixing the missing value if there is any in the dataset. Then, applied the normalization to convert the raw data to usable data, between the range of 0-1.

3.2.4 Data Splitting

The sales dataset that has been converted into usable data by preprocessing will be now split to two by data splitting. The aim for data splitting is to separate the dataset into training data and testing data. This was done to train the ANN and get accurate predictions. The ratio used for this study is 70:30 ratio.

3.2.5 Train Artificial Neural Network

Training data will be used with the help of backpropagation algorithm to train the ANN model. Repeat this process until a satisfied training result is produced by the ANN model. The purpose of training the ANN is to adjust the parameters of the neural network in order to minimise the error between the predicted outputs and the actual outputs.

3.2.6 Prediction Result

The testing data will be utilised to do sales prediction. The testing data will be input into a trained neural network. The result of sales prediction will be compared to the real sales data. The sales prediction result by ANN model will be evaluated.

3.3 Research Requirement

3.3.1 Input

We will retrieve the bakery sales dataset which will be in the csv file. The sales dataset contains bakery product sales for 1094 days (2006-2008). The data variables such as cakes, pies, cookies, smoothies, coffee and promotion will be used in the ANN model.

3.3.2 Output

Our ANN model will give 5 output which is the predicted sales of cakes, pies, cookies, smoothies, coffee. The predicted sales value will be compared to the real sales data.

3.3.3 Process Description

The initial step is to download the bakery sales dataset from Kaggle in the form of a .csv file. Subsequently, we will use Google Colab as a coding platform to implement Artificial Neural Networks (ANN) using Python. The dataset will be cleaned to identify and fill any missing values. Afterwards, the dataset will be subject to Exploratory Data Analysis to find valuable information from the sales dataset. Next, the raw data in the dataset will then be transformed into usable data using normalization to prepare it for the ANN model. Once normalization is complete, the data will be divided into a 70:30 ratio, with 70% being designated as training data and 30% as testing data. We will train the ANN model using the training data until we achieve satisfactory results. The testing data will be utilised to make predictions about future sales, and its predicted results will be compared to actual sales data.

3.3.4 Constraints and Limitation

This study only focuses on Artificial Neural Networks (ANN) to do sales prediction. There are many more forecasting methods that are suitable to do sales forecasts. Even in machine learning, ANN is only one among several that can do sales forecasts. This study cannot conclude that the ANN is the best method for sales forecasting because of the lack of evidence. Time constraints are the limiting factor that prevents us from testing all the available forecasting methods for sales forecasts.

3.3.5 Software Requirement

Software	**Description**
Google Colab	Google Colab will be used as a platform to code Python
Google Chrome	Google Chrome will be used to do research on sales prediction using Artificial Neural Networks
Google Docs	Used to write the research report
Microsoft Excel	Used for the lemonade sales dataset
Draw i.o	Used to construct the research framework and any flowchart

Table 3.2 Software requirement

Table 3.2 above shows the software requirement that will be used in this research.

3.3.6 Hardware Requirement

Hardware & Model	**Features**	**Description**
Laptop (Dell Inspiron 3443)	Processor: Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz 2.20 GHz RAM: 8GB System: 64-bit Operating System Edition: Window 10	Used for research development, documentation and presentation.

Table 3.3 Hardware requirement

Table 3.3 above shows the hardware requirement that will be used in this research.

3.3.7 Case Study

Sales prediction plays a crucial role for Malaysian business owners that are looking to grow their own business. Since using the traditional forecasting method has its own disadvantages, this study proposes using Artificial Neural Networks (ANN) to do sales forecasts. This is because ANN is good at prediction which will be able to help us do sales forecasts. The study aims to explore the capabilities of ANN as a predictive modeling technique and its potential to enhance sales forecasting accuracy.

3.4 Proposed Design

3.4.1 Artificial Neural Network

This research implements an Artificial Neural Network (ANN) to predict bakery product's sales. To be more specific, we use a type of ANN called Feed Forward Backpropagation Neural Network (FFBPNN) which is the commonly used ANN. Backpropagation algorithm is used to train the ANN by adjusting the weights and bias of the network in order to minimise the error between the predicted output and the desired output (Priyanka,2022). This is done in order to improve the accuracy of the sales prediction using ANN.

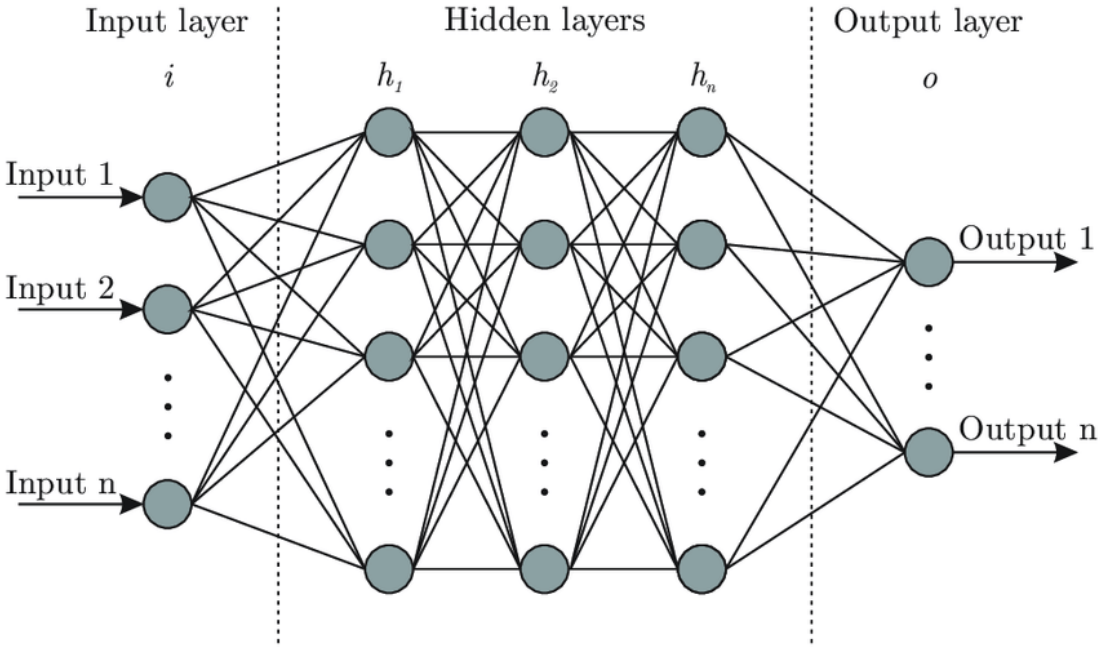

Figure 3.2 Architecture of Artificial Neural Network

Figure 3.2 above shows the typical architecture of ANN. ANN has three layers: an input layer, hidden layer and an output layer. The number of nodes in the input layer depends on the number of input data variables from the dataset. The number of nodes at the output layer depends on the desired output. The hidden layer can have one or more layers and the number of nodes in that layer can vary as well.

Since we want to input 6 data variables to the ANN, our input layer will consist of 6 input nodes. We will also have two layer of hidden layer consisting of 40 hidden nodes at the first hidden layer and another 20 hidden node at the second layer. After several rounds of parameter tuning the neural network, the combination of using two hidden layers was found to yield the best results. Specifically, the first hidden layer was designed with 40 hidden nodes, while the second hidden layer consisted of 20 hidden nodes. This configuration was determined to be optimal, as it achieved the desired balance between model complexity and generalization capability resulting in accurate sales prediction. Our output layer will have 5 output node, as the goal is to predict the number of bakery product sales such as cakes, pies, cookies, smoothies, coffee.

ReLU will be chosen as the activation function in ANN model. ReLU offers several advantages as an activation function in our ANN models. Firstly, ReLU introduces non-linearity to the network, enabling it to learn complex relationships between inputs and outputs. Additionally, ReLU is computationally efficient, making it faster to compute and optimize during the backpropagation process. It addresses the vanishing gradient problem by providing a constant gradient of 1 for positive inputs, resulting in faster and more stable training. ReLU's sparsity property allows it to discard negative values, focusing the network on relevant features and reducing sensitivity to irrelevant or noisy inputs.

Backpropagation algorithm is used to train artificial neural network model. The mean square error is utilize as the loss function. MSE calculates the average difference between the predicted output of the network and the target output, providing a measure of how well the network is performing. By minimizing the MSE during training, the network adjusts its weights and biases through gradient descent, iteratively updating them to reduce the error and improve the accuracy of the predictions. The use of MSE in backpropagation allows the network to effectively learn and adjust its parameters, optimizing its performance and enhancing its prediction capabilities.

3.4.2 Flowchart

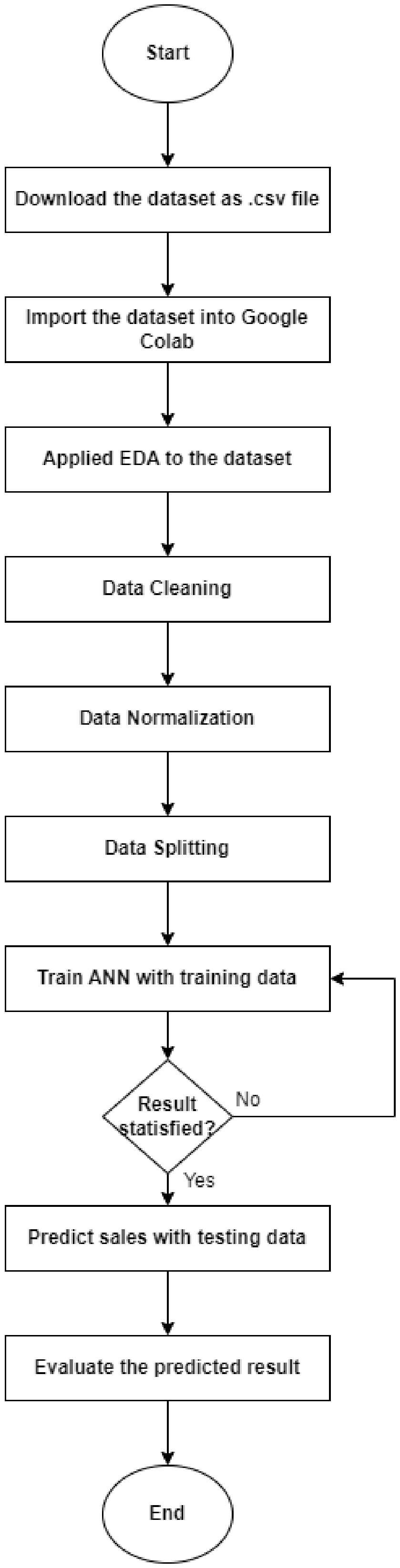

Figure 3.3 Flowchart

Figure 3.3 above shows the entire process of sales prediction using ANN in the form of flowchart. We start by downloading the dataset which is the bakery sales dataset from Kaggle in a .csv file. Then, we import the dataset to the Google Collab which is a platform for us to write the code in Python language. After that, the dataset is applied with EDA to gain valuable insight from the dataset. The dataset will then go through data preprocessing starting with data cleaning. Check the dataset for any missing value. Then, the data from the dataset will transform from raw data into usable data with data normalization. It will normalize the data to ensure all data are in the range of 0 and 1. Next, split the normalized data into the ratio of 70:30. 70% will be the training data and 30% will be the testing data.

The 70% training data will be used to train the ANN until a satisfied result is obtained from the ANN. Then, the other 30% of testing data will be used to predict bakery product sales. The predicted sales result will be compared to the real sales and be evaluated.

3.5 Data Design

Table 3.4 below shows the variables and its description from the bakery sales dataset.

Variables	**Description**
Date	Date (2006-2008)
Weekday	Day from monday to sunday
Cakes	Number of cakes sales
Pies	Number of pies sales
Cookies	Number of cookies sales
Smoothies	Number of smoothies sales
Coffee	Number of coffee sales
Promotion	Promotion or none (no promotion)

Table 3.4 Dataset description

3.6 Proof of Initial Concept

3.6.1 Evidence of Early Work

First, open the Kaggle website to find a suitable dataset to be used for our research. Then, download the dataset from the Kaggle website in a .csv file. The Kaggle website and the bakery sales dataset is shown on Figure 3.4 & 3.5 below.

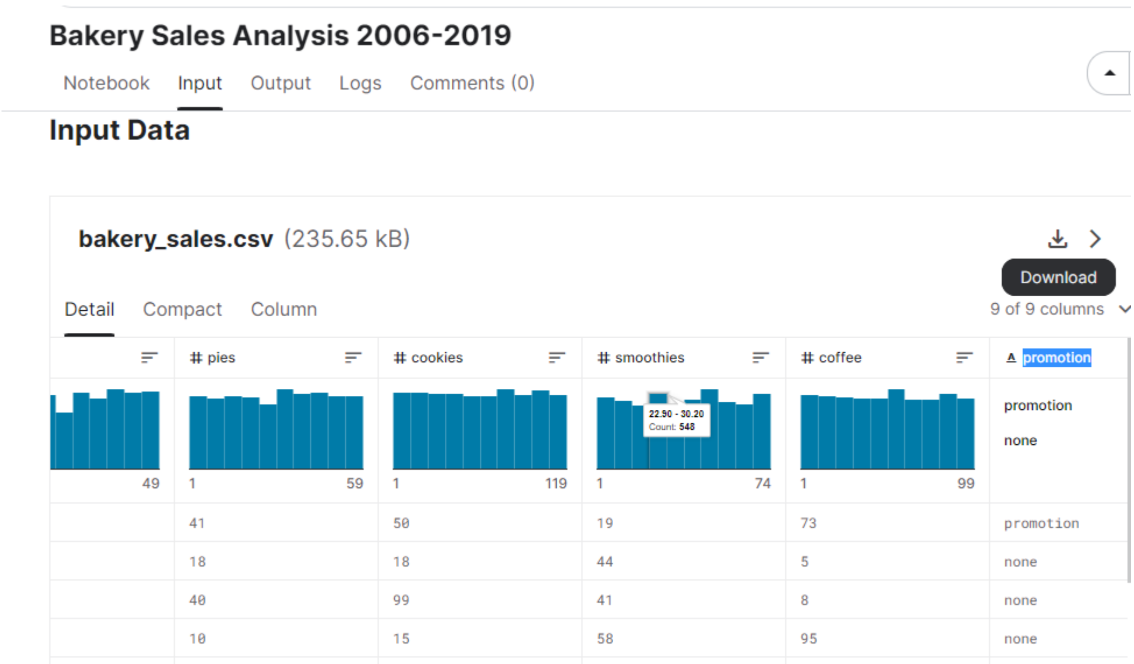

Figure 3.4 Bakery sales dataset from Kaggle

	B	C	D	E	F	G	H	I
	Date	weekday	cakes	pies	cookies	smoothies	coffee	promotion
0	1/1/2006	Sunday	45	41	50	19	73	promotion
1	2/1/2006	Monday	48	18	18	44	5	none
2	3/1/2006	Tuesday	1	40	99	41	8	none
3	4/1/2006	Wednesday	4	10	15	58	95	none
4	5/1/2006	Thursday	4	44	67	71	20	promotion
5	6/1/2006	Friday	40	27	104	39	76	promotion
6	7/1/2006	Saturday	10	21	10	26	5	none
7	8/1/2006	Sunday	20	58	59	74	68	promotion
8	9/1/2006	Monday	22	5	28	61	54	none
9	10/1/2006	Tuesday	37	43	76	72	67	promotion
10	11/1/2006	Wednesday	24	3	60	20	10	none

Figure 3.5 Bakery sales dataset in Microsoft Excel

We perform Exploratory Data Analysis (EDA) in order to acquire a more thorough comprehension of the dataset. EDA will aid us in uncovering useful insights formthe dataset.

```
#View the data
df.head()
```

	Unnamed: 0	Date	weekday	cakes	pies	cookies	smoothies	coffee	promotion
0	0	1/1/2006	Sunday	45	41	50	19	73	promotion
1	1	2/1/2006	Monday	48	18	18	44	5	none
2	2	3/1/2006	Tuesday	1	40	99	41	8	none
3	3	4/1/2006	Wednesday	4	10	15	58	95	none
4	4	5/1/2006	Thursday	4	44	67	71	20	promotion

Figure 3.6 View the dataset

From the Figure 3.6 above, we can know our own dataset such as what variable is in the dataset. Next, Figure 3.7 below show the information at our dataset such as data type.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1096 entries, 0 to 1095
Data columns (total 9 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Unnamed: 0   1096 non-null   int64
 1   Date         1096 non-null   object
 2   weekday      1096 non-null   object
 3   cakes        1096 non-null   int64
 4   pies         1096 non-null   int64
 5   cookies      1096 non-null   int64
 6   smoothies    1096 non-null   int64
 7   coffee       1096 non-null   int64
 8   promotion    1096 non-null   object
dtypes: int64(6), object(3)
memory usage: 77.2+ KB
```

Figure 3.7 Dataset information

Figure 3.8 below shows there is no missing value and the dataset is in good condition.

```
Unnamed: 0    0
Date          0
weekday       0
cakes         0
pies          0
cookies       0
smoothies     0
coffee        0
promotion     0
dtype: int64
```

Figure 3.8 Data Cleaning

3.7 Testing Plan

First, the bakery sales dataset is downloaded from Kaggle in a .csv file and imported to Google Colab. The dataset will go through a data cleaning to find and fix any missing value. Next, the data will be normalized to convert it from raw data into a usable form. The next step is to divide the data into two sets, with 70% of the sales data being used as training data and the remaining 30% as testing data. The training data will be used to train the ANN until it produces a satisfied result. Then, testing data is utilised to predict the bakery product sales. Finally, the predicted result will be compared to the real sales and be evaluated.

3.8 Potential Used of Proposed Solution

There are several potential uses of the proposed solution for sales prediction using ANN. The trained ANN model can be used to predict future sales based on historical data, which can help business owners to make better decisions on their inventory management, production planning, and cash flow. Next, ANN can be used to identify patterns and trends in the dataset, which can help business owners to understand the factors that affect sales and make adjustments accordingly to reach the desired sales target. Finally, ANN can be used to optimise pricing strategy by predicting the demand for a product at different price points.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter will provide a thorough explanation of how the implementation was carried out during the course of this research. In order to predict sales, an ANN model was created with the help of Google Collab as a platform to write the Python code. Additionally, there will be a visualization stage to display the predicted output. The implementation process will be elaborated on in this section.

4.2 Input

The data used for this research was sourced from Kaggle, and it was published by Santosh H Kamble. The bakery sales dataset obtained from Kaggle contains 5114 rows of data and comprises 8 data variables which are date, weekday, cakes, pies, cookies, smoothies, coffee and promotion. However, for the purpose of our research, we will only utilize 3 years of sales data (2006-2008) and 6 data variables - cakes, pies, cookies, smoothies, coffee and promotion - as input for our ANN model. We choose these 6 data variables because it will affect the final sales prediction. The promotion variable is converted into binary values (1 and 0) using one-hot encoding before being inputted into the ANN model. This variable represents whether a promotion was active or not for that day. The remaining five variables, namely cakes, pies, cookies, smoothies, and coffee, represent the number of sales for each product category. The dataset being used for our research is shown below in Figure 4.1

	B	C	D	E	F	G	H	I
	Date	weekday	cakes	pies	cookies	smoothie:	coffee	promotion
0	1/1/2006	Sunday	45	41	50	19	73	promotion
1	2/1/2006	Monday	48	18	18	44	5	none
2	3/1/2006	Tuesday	1	40	99	41	8	none
3	4/1/2006	Wednesday	4	10	15	58	95	none
4	5/1/2006	Thursday	4	44	67	71	20	promotion
5	6/1/2006	Friday	40	27	104	39	76	promotion
6	7/1/2006	Saturday	10	21	10	26	5	none
7	8/1/2006	Sunday	20	58	59	74	68	promotion
8	9/1/2006	Monday	22	5	28	61	54	none
9	10/1/2006	Tuesday	37	43	76	72	67	promotion
10	11/1/2006	Wednesday	24	3	60	20	10	none
11	12/1/2006	Thursday	7	53	108	6	82	promotion
12	13/1/2006	Friday	25	40	48	7	29	none
13	14/1/2006	Saturday	25	20	118	28	97	promotion
14	15/1/2006	Sunday	13	53	48	26	95	promotion
15	16/1/2006	Monday	2	8	87	40	36	none
16	17/1/2006	Tuesday	39	32	33	22	69	none
17	18/1/2006	Wednesday	40	33	105	25	21	promotion
18	19/1/2006	Thursday	24	5	106	63	48	promotion
19	20/1/2006	Friday	47	53	54	50	95	promotion
20	21/1/2006	Saturday	25	28	5	52	30	none
21	22/1/2006	Sunday	18	25	30	14	37	none

bakery_sales

Figure 4.1 Bakery sales dataset

4.3 Training and Testing

The bakery sales dataset will be split into two in a ratio of 70:30. 70% of the dataset will be the training data set and the other 30% will be the testing data set. The 70% training data will be used to train the ANN until a satisfied result is obtained from the ANN. Then, the other 30% of testing data will be used to predict 5 bakery product sales such as cakes, pies, cookies, smoothies, coffee. The training dataset will contain 766 rows of data variables which are from cakes, pies, cookies, smoothies, and coffee and promotion. Testing dataset will contain the remaining 328 rows of data variables.

4.4 Output

In this research, we have 5 output variable for our ANN model which is the prediction bakery product sales such as cakes, pies, cookies, smoothies, coffee. The 6 input data variables from the database will be used to get the prediction of bakery product sales. The 30% of testing data will be used to get the predicted sales output. Data visualization will be utilized to compare the predicted sales to the actual sales. Example of output data visualization is shown below in Figure 4.2.

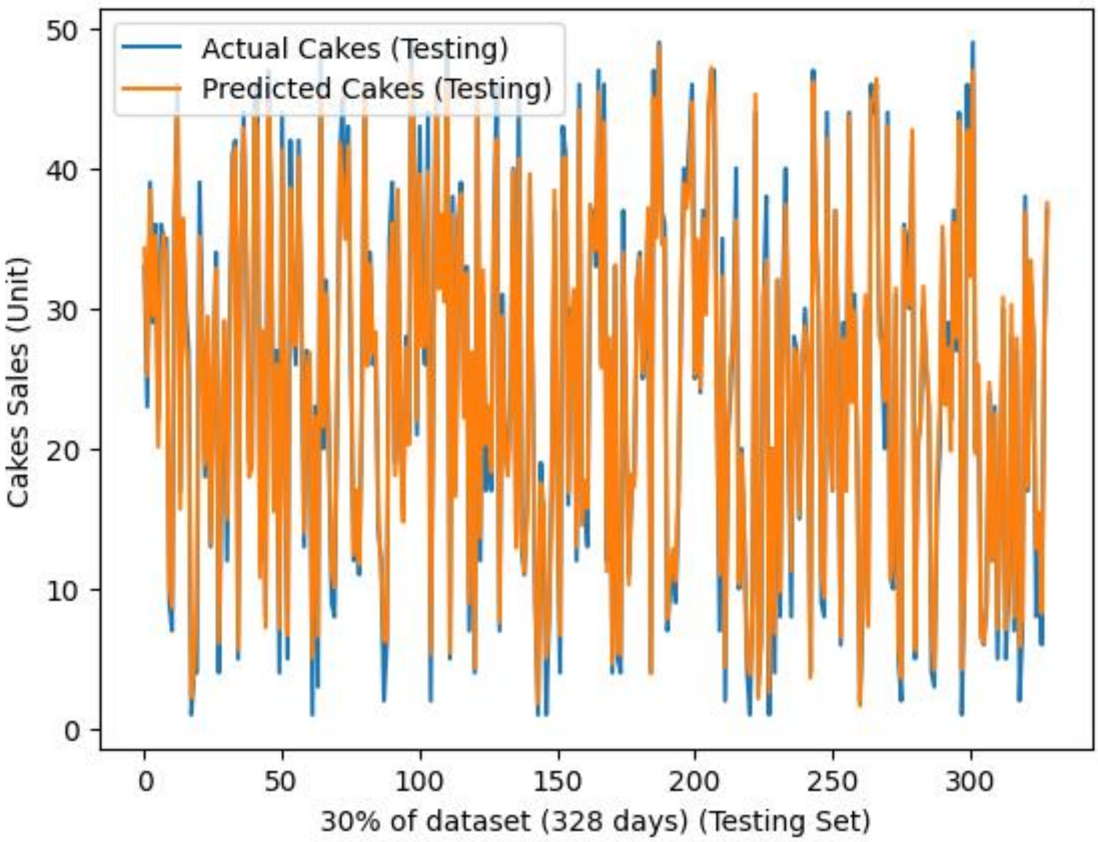

Figure 4.2 Cakes sales prediction

4.5 Code Description

This is the code for building an Artificial Neural Network (ANN) model for predicting bakery product sales such as cakes, pies, cookies, smoothies, coffee based on 6 variable input.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler, LabelEncoder
from sklearn.metrics import mean_squared_error
import tensorflow as tf
from tensorflow import keras
```

Figure 4.3 Library

Starts by importing the required libraries such as Pandas for data manipulation, NumPy for numerical operations, Matplotlib for data visualization, scikit-learn for preprocessing and evaluation, and TensorFlow with Keras for building and training the ANN model. Which is shown above on Figure 4.3.

```
# Load the dataset
data = pd.read_csv('bakery_sales.csv')
```

Figure 4.4 Read dataset

Load the dataset which is in .csv file that is shown above. The extracts the features (inputs) and target variables (outputs) from the dataset. The features include columns cakes, pies, cookies, smoothies, coffee, and promotion. The target variables include all columns except promotion. Which is shown in Figure 4.5 below.

```
X = data[['cakes', 'pies', 'cookies', 'smoothies', 'coffee', 'promotion']]
y = data[['cakes', 'pies', 'cookies', 'smoothies', 'coffee']]
```

Figure 4.5 Dataset variables

Performs label encoding on the promotion column using the LabelEncoder class from scikit-learn. It is a common preprocessing technique which is applied to categorical variables to convert them into numerical values. The numerical values of promotion is 1 and none is 0. Which is shown on Figure 4.6 below.

```
# Data preprocessing
# Perform label encoding for the promotion column
promotion_encoder = LabelEncoder()
X['promotion'] = promotion_encoder.fit_transform(X['promotion'].copy())
```

Figure 4.6 Encoded

Uses the MinMaxScaler class from scikit-learn to normalize the input features. It scales the input data to a specified range (0 to 1) using the minimum and maximum values. The normalized input features are stored in the X_scaled array. Then, splits our dataset into training and testing sets using the train_test_split() function from scikit-learn. It allocates 70% of the data for training and 30% for testing. The random state is set to 42 for reproducibility. This is shown on Figure 4.7 below.

```
# Normalize the input features
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X_encoded)

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)
```

Figure 4.7 Normalize & split

Build the ANN model. The first hidden layer has 40 hidden node with the ReLU activation function. The second hidden layer has 20 hidden node with ReLU activation. The output layer has 5 output node (one for each target variable). Then, compiles the model by specifying the optimizer (Adam) and the loss function (mean squared error) to minimize during training. Next, The trains the model on the training set for 100 epochs. The batch size is set to 32, and verbose is 0 to ensures that no progress information is displayed during training. The training history is stored in the history variable. This is shown below in figure 4.8.

```python
# Build the ANN model
model = keras.Sequential([
    keras.layers.Dense(40, activation='relu', input_shape=(X_train.shape[1],)),
    keras.layers.Dense(20, activation='relu'),
    keras.layers.Dense(5)  # Output layer with 5 units (one for each target variable)
])

# Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model on the training set
history = model.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0)
```

Figure 4.8 Build and train ANN

```python
# Perform sales prediction on the testing set
y_test_pred = model.predict(X_test)

# Visualize the predicted sales and actual sales on the testing set for all target variables
labels = ['Cakes', 'Pies', 'Cookies', 'Smoothies', 'Coffee']

for i in range(y_test.shape[1]):
    plt.plot(np.arange(len(y_test)), y_test.iloc[:, i], label=f'Actual {labels[i]} (Testing)')
    plt.plot(np.arange(len(y_test)), y_test_pred[:, i], label=f'Predicted {labels[i]} (Testing)')

    plt.xlabel('30% of dataset (328 days) (Testing Set)')
    plt.ylabel(f'{labels[i]} Sales (Unit)')
    plt.legend()
    plt.show()
```

Figure 4.9 Sales prediction & visualization

Perform sales prediction on the testing set and visualise the predicted sales vs actual sales on the testing set for all 5 variable output which is shown above in Figure 4.9. We can even gain more information regarding the sales value difference between the predicted sales and the actual sale which is shown below in Figure 4.9.

```python
df_test_comparison = pd.DataFrame({
    'Actual Cakes': y_test['cakes'].values.flatten(),
    'Predicted Cakes': y_test_pred[:, 0],
    'Actual Pies': y_test['pies'].values.flatten(),
    'Predicted Pies': y_test_pred[:, 1],
    'Actual Cookies': y_test['cookies'].values.flatten(),
    'Predicted Cookies': y_test_pred[:, 2],
    'Actual Smoothies': y_test['smoothies'].values.flatten(),
    'Predicted Smoothies': y_test_pred[:, 3],
    'Actual Coffee': y_test['coffee'].values.flatten(),
    'Predicted Coffee': y_test_pred[:, 4]
})

print(df_test_comparison)
```

Figure 4.9 Comparison between predicted sales and actual sales

Finally, calculate and show the root mean squared error (RMSE) and the mean squared error (MSE) of each output variable which is shown below in Figure 4.10.

```python
# Calculate RMSE and MSE for each target variable
rmse_scores = np.sqrt(mean_squared_error(y_test, y_test_pred, multioutput='raw_values'))
mse_scores = mean_squared_error(y_test, y_test_pred, multioutput='raw_values')

# Create a DataFrame to display the scores
scores_df = pd.DataFrame({'Target Variable': labels, 'RMSE': rmse_scores, 'MSE': mse_scores})

# Display the scores
print(scores_df)
```

Figure 4.10 RMSE and MSE code

4.6 Process Description

The initial step is to download the bakery sales dataset from Kaggle in the form of a .csv file. Subsequently, we will use Google Colab as a coding platform to implement Artificial Neural Networks (ANN) using Python. The dataset will be cleaned to identify and fill any missing values. Afterwards, the dataset will be subject to Exploratory Data Analysis to find valuable information from the dataset. The dataset will be divided into a 70:30 ratio, with 70% being designated as training data and 30% as testing data. We will train the ANN model using the training data until we achieve satisfactory results. The testing data will be utilised to make predictions about future sales, and its predicted results will be compared to actual sales data.

4.7 Result

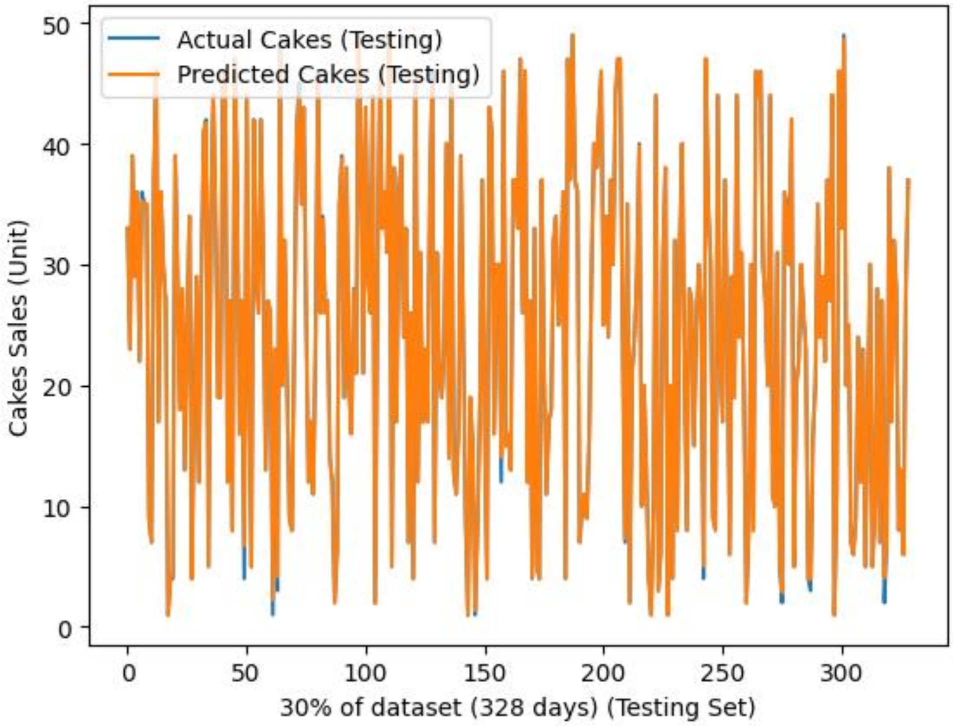

Figure 4.11 Cakes sales comparison

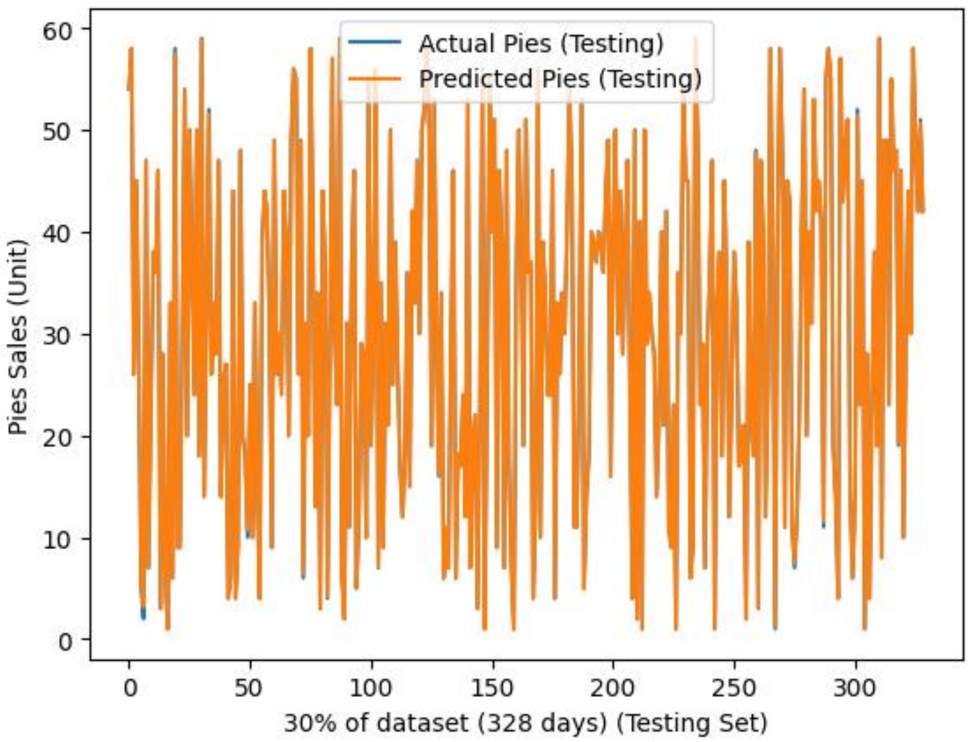

Figure 4.12 Pies sales comparison

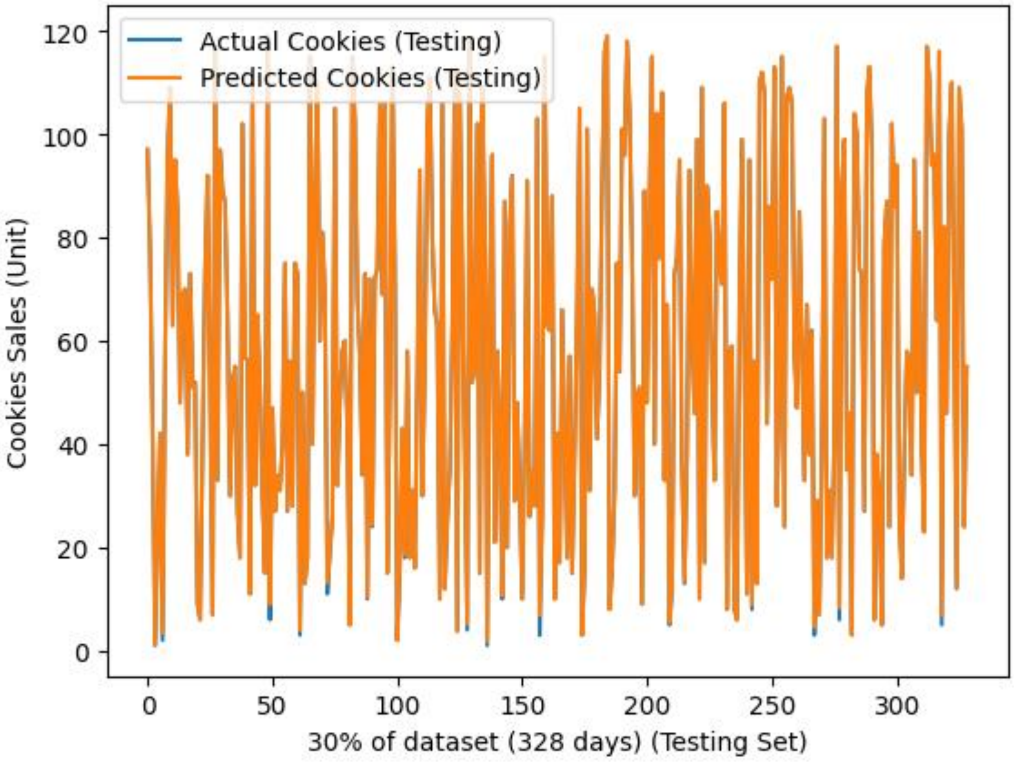

Figure 4.13 Cookies sales comparison

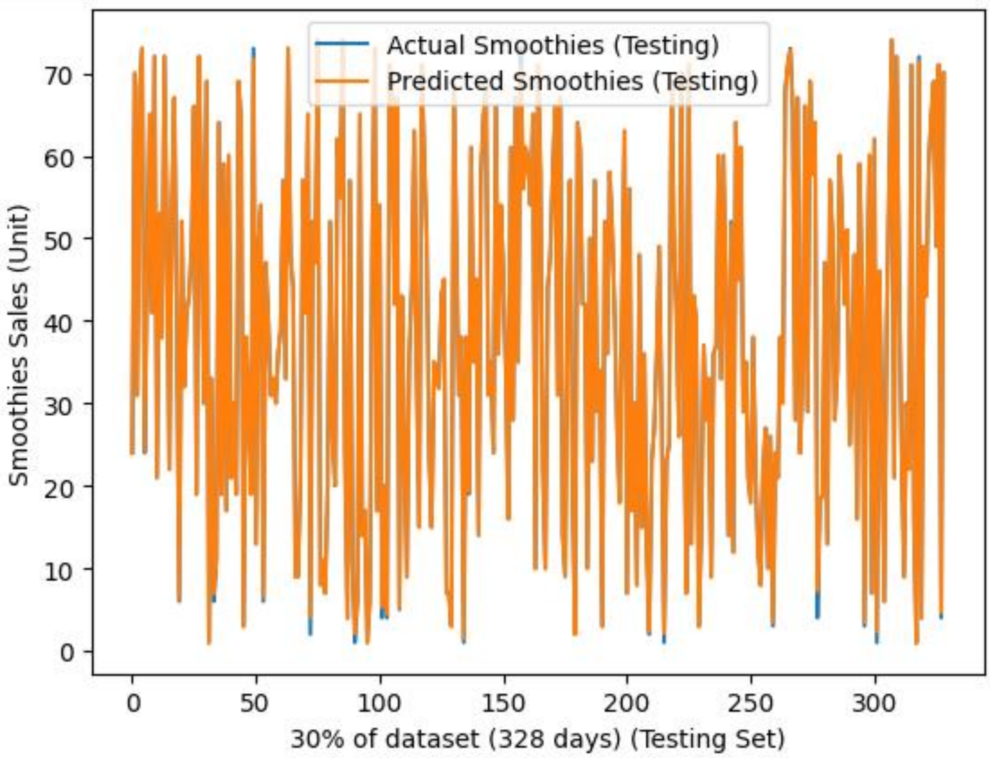

Figure 4.14 Smoothies sales comparison

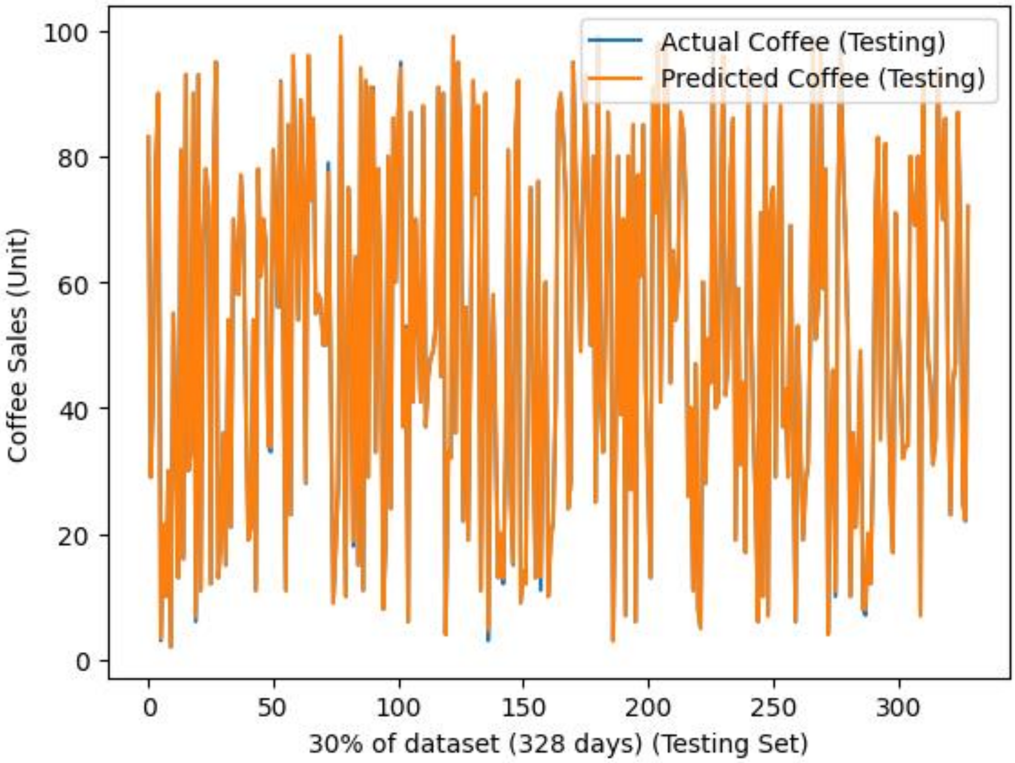

Figure 4.15 Coffee sales comparison

Based on the result of Figure 4.11, 4.12, 4.13, 4.14 & 4.15 above, the ANN model is working correctly and be able to produce an accurate sales prediction. This accurate sales prediction is achieved after we already fine tune the ANN model parameter and utilising the combination of 40 hidden node & 20 hidden node for our two hidden layer. Figure 4.16 below show the result of RMSE & MSE of each of the predicted variable at the testing set.

	Target Variable	RMSE	MSE
0	Cakes	1.707024	2.913930
1	Pies	1.463274	2.141170
2	Cookies	3.030024	9.181046
3	Smoothies	1.905107	3.629433
4	Coffee	1.525352	2.326700

Figure 4.16 All 5 variable RMSE & MSE result

	Actual Cakes	Predicted Cakes	Actual Pies	Predicted Pies
0	33	34.303078	54	53.247219
1	23	25.180897	58	56.828773
2	39	38.438984	26	26.551512
3	29	29.490536	45	44.714737
4	36	35.205585	28	28.742054
..	...	...	...	...
324	8	12.895584	58	53.909950
325	13	15.445188	49	47.460049
326	6	8.240918	42	40.657452
327	28	28.295826	51	50.636196
328	37	37.535931	42	41.908878

Figure 4.17 Cakes & Pies sales comparison (value)

	Actual Cookies	Predicted Cookies	Actual Smoothies	Predicted Smoothies
0	97	93.615234	24	21.854729
1	77	75.681557	70	69.387268
2	43	40.697868	31	30.107227
3	1	3.707116	68	69.113365
4	22	21.830498	73	73.433838
..	...	...	...	...
324	12	17.966379	69	67.638718
325	109	105.842682	49	46.833797
326	101	100.902359	71	70.764931
327	24	26.182121	4	6.182090
328	55	52.495766	70	68.966164

Figure 4.18 Cookies & Smoothies sales comparison (value)

	Actual Coffee	Predicted Coffee
0	83	81.838699
1	29	27.964920
2	44	42.900383
3	79	79.848747
4	90	90.084274
..	...	...
324	87	87.308708
325	67	65.663567
326	25	24.369818
327	22	23.539362
328	72	70.895271

Figure 4.19 Coffee sales comparison (value)

Figure 4.17, 4.18 & 4.19 above show the value sales comparison between predicted sales and the actual sales. The predicted sales closely align with the actual sales data in the testing set, showcasing the impressive accuracy of the artificial neural network (ANN) in forecasting sales.

4.8 Discussion

Before we obtain an accurate sales prediction result, we need to do trial and error to fine tune parameter of the ANN model so that we can get sales prediction as accurate as possible. Previously we try using 20 hidden node & 10 hidden node combination but the ANN model give inaccurate sales prediction result. Which is shown below in Figure 4.20.

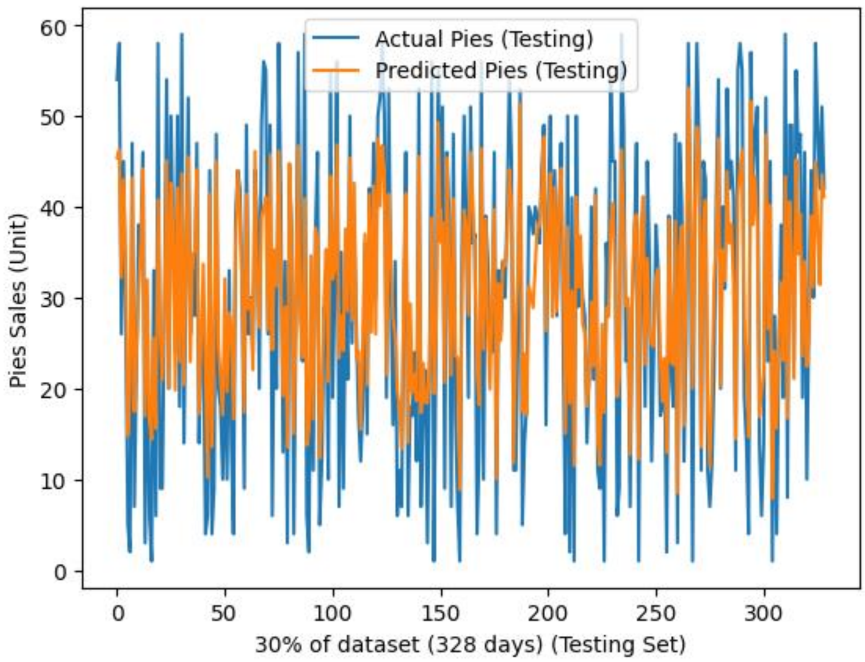

Figure 4.20 Inaccurate Pies sales prediction

The more we increase the number of hidden node at our hidden layer the better our sales prediction will be. This is because by increasing the number of hidden nodes in a neural network, we will increase the model capacity to learn complex relationships and capture more nuanced patterns in the data. This can be particularly advantageous when dealing with complex datasets or problems that require a higher level of abstraction.

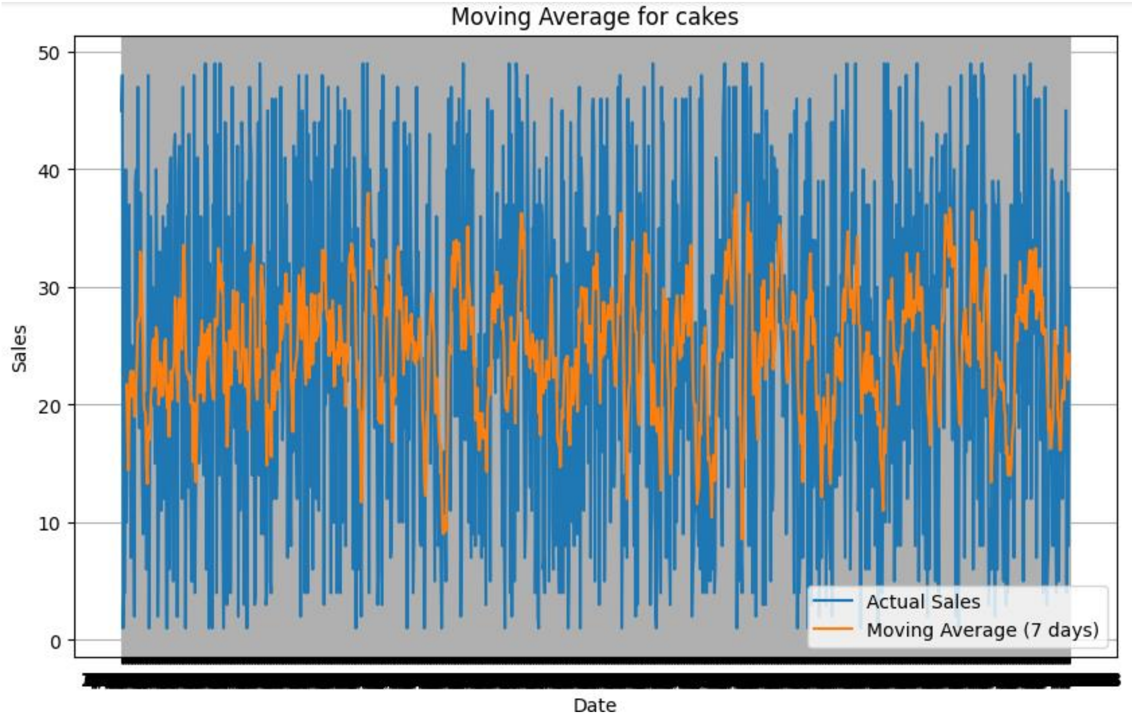

Figure 4.21 Sales prediction using moving average

Figure 4.21 above show cakes sales prediction using moving average forecast that give vastly inaccurate sales prediction. Moving average forecast is not suitable for this kind of large and complex dataset. ANN is the best choise to be used for this type of dataset.

4.9 Tools

In order to carry out this research, several tools will be employed. Google Collab will serve as the platform for us to develop the Python code and construct the ANN model. For the purpose of writing reports and documents, Microsoft Word software will be utilized. Additionally, we will need to install several libraries into Google Colab, including pandas, numpy, scikit-learn, tensorflow, and matplotlib. These libraries will greatly help us in constructing the ANN model for sales prediction.

4.10 Constraints and Limitation

This research solely focuses on utilizing Artificial Neural Networks (ANN) for sales prediction. While there are numerous other forecasting methods available to perform sales forecasts, ANN is just one of many approaches that can be used for this purpose. The lack of evidence prevents this study from determining that ANN is the most effective method for sales forecasting. Time constraints are the primary limiting factor, preventing us from assessing all the potential forecasting methods available for sales forecasts.

4.11 Case Study

Due to the drawbacks associated with traditional forecasting methods, this research suggests using Artificial Neural Networks (ANN) for sales forecasts. ANN is a suitable method for sales forecasts because of its predictive capabilities. The study aims to explore the capabilities of ANN as a predictive modeling technique and its potential to enhance sales forecasting accuracy.

CHAPTER 5

CONCLUSION

5.1 Objective Revisit

- To study the Artificial Neural Network (ANN) for sales prediction.

This objective is already achieved by discussing my ANN proposed design. This reasearch used a type of ANN called Feed Forward Backpropagation Neural Network (FFBPNN).

- To develop a sales prediction model using ANN.

This research has succesfully build ANN model and conduct sales prediction using the bakery sales dataset. The details have been discussed in chapter 4.

- To evaluate the result of this forecasting model.

ANN forecasting model has succesfully give an accurate prediction using the bakery sales dataset. It is more suited on this type of dataset that is large and complex compare to moving average forecast that give inaccurate sales prediction.

5.2 Limitation

- Parameter tuning

ANN have many parameters such as learning rate, number of layers, and number of neurons that need to be tuned to achieve optimal performance. Finding the right combination of parameters can be a time-consuming and challenging task.

- Noisy or missing data handling

ANN can be sensitive to noisy or missing data, and their performance may degrade when faced with incomplete or corrupted input. Preprocessing steps or data imputation techniques are often necessary to handle such issues.

5.3 Future Works

For future work, we can try to add more additional features that can impact sales such as economic indicators, seasonal factors, marketing campaigns, competitor data, or customer sentiment. By incorporating a broader range of variables, the predictive power of the ANN can be enhanced. Next, we can continue to fine tuning the network architecture. Experiment with different network architectures, including variations in the number of layers, neurons, activation functions, and regularization techniques. Fine-tuning the architecture can help optimize the model's performance and improve its ability to capture complex relationships within the sales data.

We can also implement preprocessing techniques to handle outliers and noisy data points that might adversely affect the model's performance. Outlier detection algorithms or robust normalization methods can help mitigate the impact of outliers and enhance the accuracy of the predictions. Finally, develop methods for updating the sales prediction model in real-time as new sales data becomes available. This allows the model to adapt to changing market conditions and ensures that the predictions remain up to date.

REFERENCES

Rohn, S. (2022, November 7). *Sales forecasting 101: Benefits, challenges, tools (2023)*. The Whatfix Blog | Drive Digital Adoption. Retrieved December 5, 2022, from https://whatfix.com/blog/sales-forecasting/

Stojanovic , F. (2022, May 21). *Sales forecast report: How to create one and what should you include?: Databox blog*. Databox. Retrieved December 5, 2022, from https://databox.com/sales-forecast-report#:~:text=Historical%20sales%20forecasting%20involves%20analyzing,are%20several%20issues%20with%20it.

V, K. (2021, April 14). *Artificial Neural Network: How does artificial neural network work*. Analytics Vidhya. Retrieved December 6, 2022, from https://www.analyticsvidhya.com/blog/2021/04/artificial-neural-network-its-inspiration-and-the-working-mechanism/#:~:text=The%20working%20mechanism%20of%20Artificial%20Neural%20Network&text=Signals%20are%20sometimes%20modified%20at,world)%20and%20the%20process%20repeats.

Marr, B. (2022, October 12). *What are artificial neural networks - a simple explanation for absolutely anyone*. Forbes. Retrieved December 6, 2022, from https://www.forbes.com/sites/bernardmarr/2018/09/24/what-are-artificial-neural-networks-a-simple-explanation-for-absolutely-anyone/?sh=377d86e11245

Burns, E., & Burke, J. (2021, March 26). *What is a neural network? explanation and examples*. Enterprise AI. Retrieved December 6, 2022, from https://www.techtarget.com/searchenterpriseai/definition/neural-network

Pillemer, N. (2020, September 1). *10 advantages of e-commerce for consumers & businesses*. Become. Retrieved February 15, 2023, from https://www.become.co/blog/ecommerce-advantages-consumers-businesses/

Mahalingam, K. (2022, November 30). *Importance of sales forecasting & six factors to consider for accurate forecasting*. Chargebee Compass. Retrieved February 15, 2023, from https://www.chargebee.com/blog/importance-of-sales-forecasting/

Priyanka. (2022, August 16). *Back propagation algorithm - back propagation in Neural Networks*. Intellipaat Blog. Retrieved February 15, 2023, from https://intellipaat.com/blog/tutorial/artificial-intelligence-tutorial/back-propagation-algorithm/

Krishnan, S. (2021, August 6). *How do determine the number of layers and neurons in the hidden layer?* Medium. Retrieved February 15, 2023, from https://medium.com/geekculture/introduction-to-neural-network-2f8b8221fbd3

Patil, P. (2018, May 30). *What is exploratory data analysis?* Medium. Retrieved February 15, 2023, from https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15

APPENDIX A
CODE

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler, LabelEncoder
from sklearn.metrics import mean_squared_error
import tensorflow as tf
from tensorflow import keras

# Load the dataset
data = pd.read_csv('bakery_sales.csv')

# Extract the relevant variables
X = data[['cakes', 'pies', 'cookies', 'smoothies', 'coffee', 'promotion']]
y = data[['cakes', 'pies', 'cookies', 'smoothies', 'coffee']]

# Display the head of the dataset
print(data.head())

# Data preprocessing
# Perform label encoding for the promotion column
promotion_encoder = LabelEncoder()
X['promotion'] = promotion_encoder.fit_transform(X['promotion'].copy())

# Display the result
print("\nEncoded promotion column:")
print(X['promotion'])

# Normalize the input features
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)

# Create a DataFrame with normalized input features and original target variables
df_normalized = pd.DataFrame(X_scaled, columns=['cakes', 'pies', 'cookies', 'smoothies', 'coffee',
'promotion'])

# Display the DataFrame
print(df_normalized)

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)

# Display the number of samples in the training and testing sets
print("Number of samples in the training set (days):", X_train.shape[0])
print("Number of samples in the testing set (days):", X_test.shape[0])

# Build the ANN model
model = keras.Sequential([
    keras.layers.Dense(40, activation='relu', input_shape=(X_train.shape[1],)), # 40 hidden nodes
    keras.layers.Dense(20, activation='relu'), # 20 hidden nodes
    keras.layers.Dense(5)  # 5 output nodes
])
```

```
# Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')

# Train the model on the training set
history = model.fit(X_train, y_train, epochs=150, batch_size=32, verbose=1)

# Visualize the training loss
plt.figure(figsize=(8, 6))
plt.plot(history.history['loss'])
plt.title('Training Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.show()

# Perform sales prediction on the testing set
y_test_pred = model.predict(X_test)

# Visualize the predicted sales and actual sales on the testing set for all target variables
labels = ['Cakes', 'Pies', 'Cookies', 'Smoothies', 'Coffee']

for i in range(y_test.shape[1]):
    plt.plot(np.arange(len(y_test)), y_test.iloc[:, i], label=f'Actual {labels[i]} (Testing)')
    plt.plot(np.arange(len(y_test)), y_test_pred[:, i], label=f'Predicted {labels[i]} (Testing)')

    plt.xlabel('30% of dataset (328 days) (Testing Set)')
    plt.ylabel(f'{labels[i]} Sales (Unit)')
    plt.legend()
    plt.show()

# Create a dataframe to compare actual and predicted sales
df_test_comparison = pd.DataFrame({
    'Actual Cakes': y_test['cakes'].values.flatten(),
    'Predicted Cakes': y_test_pred[:, 0],
    'Actual Pies': y_test['pies'].values.flatten(),
    'Predicted Pies': y_test_pred[:, 1],
    'Actual Cookies': y_test['cookies'].values.flatten(),
    'Predicted Cookies': y_test_pred[:, 2],
    'Actual Smoothies': y_test['smoothies'].values.flatten(),
    'Predicted Smoothies': y_test_pred[:, 3],
    'Actual Coffee': y_test['coffee'].values.flatten(),
    'Predicted Coffee': y_test_pred[:, 4]
})

print(df_test_comparison)

# Calculate RMSE and MSE for each output variable
rmse_scores = np.sqrt(mean_squared_error(y_test, y_test_pred, multioutput='raw_values'))
mse_scores = mean_squared_error(y_test, y_test_pred, multioutput='raw_values')

# Create a DataFrame to display the scores
scores_df = pd.DataFrame({'Output Variable': labels, 'RMSE': rmse_scores, 'MSE': mse_scores})

# Display the scores
print(scores_df)
```

APPENDIX B
SAMPLE APPENDIX 2