# CHAPTER II :LITERATURE REVIEW

## 2.1 Introduction

Chapter II introduces and explains on the different types of Machine Learning (ML). On top of that, a strand of literature that becomes the foundation for this study is reviewed based on previous studies. This chapter has seven sections which includes Section 2.1 on the introduction of the chapter, Section 2.2 introduces and explains on the concept of ML. Followed by Section 2.3 on the explanation of traditional ML which includes Logistic Regression (LR), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGB). Next, Section 2.4 on the description of Deep Learning which includes Deep Belief Network (DBN), Multilayer Perception (MLP) and Stacked Auto-encoder (SAE). Furthermore, Section 2.5 discusses the difference between traditional ML and deep learning, Section 2.6 reviews on which ML model works best, and lastly the Section 2.7 the summary of all the sections to conclude the chapter.

## 2.2 Introduction to Machine Learning (ML)

Machine Learning (ML) is an idea of where the machine is granted to learn and train from experience, and without being too explicitly designed. Therefore, programmer, where in this case is the investor would have to write the code, so the machine could make sense of the data and build logic or prediction based on the given data. ML can also work as an inference. The main difference between ML with statistics is that ML do predictions on data points while statistics define relationships between data points. It is worth to mention because some investors or traders got mixed ideas on the concept of ML with the concept of statistics.

The first step on how ML works is to have a training data as demonstrated in the Figure 2.2-1. It is a preprogramed coding that is accessible and is usually saved in library of the language used. If it is not in the library, the packages are available to be downloaded online. Next, the researcher or programmer will train the ML algorithms to create a model input data. Now the model is generated, a new input data will undergo ML algorithms to create predictions. The new input data can be anything: trading stocks, students' heights, or water density. It depends on the objective of the research. If the algorithm works, it a successful model. If not, the process programming will repeat again and again until a desired output is obtained based on the aim of the research.

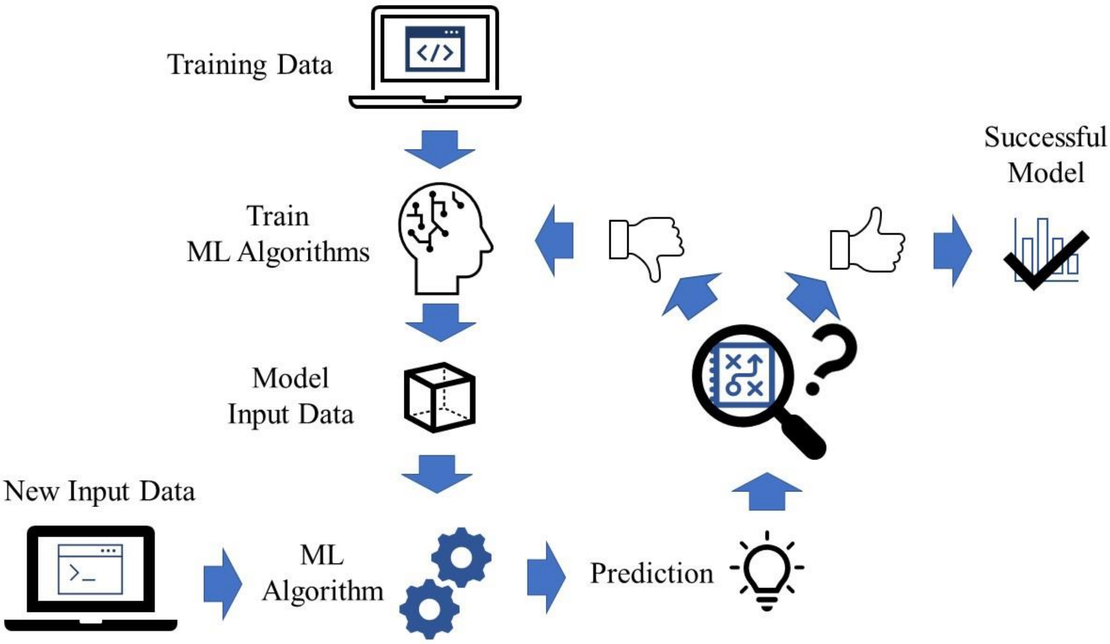


Figure 2.2-1: Process of how ML works.

There are three different sub-categorized of ML which are called supervised learning, unsupervised learning, and reinforces learning. This study will emphasize on supervised learning (also known as traditional ML) and unsupervised learning (also known as deep learning). Supervised learning is when the machine will learn and be guided by the coding written by the programmer. The dataset will act as an instructor and train the machine or the model. Hence, the trained model can start to make decision or prediction when new data is being inputted to it (Atul, 2019). Unsupervised learning on the other hand, is when the model learns via observation and obtains structures in the data. It will automatically determine the relationships in the dataset and the patterns by forming clusters once the model is provided a dataset.

In training the model, most algorithm practices backward propagation of errors, for short backpropagation. Backpropagation is a method that computes error function gradient with respect to the weight of neural network. It is used to improve the prediction's accuracy in data mining. The training network structure often comes from artificial neural networks (ANN) that is inspired to imitate the human neural networks' function present in the brain. There exists input and output layer in training. Each layer consists of cells/nodes like neurons in a brain as activations and the connection from one cell to another cell is called synapses where the weight and the biases are calculated as shown in Figure 2.2-2. Equation 2.1 shows the relation between cells, weights, and biases to generate the output. It is consisted of a non-linear activation function of the weighted sum of inputs for a layer to another cell in the next layer (Nabipour, et al., 2020).

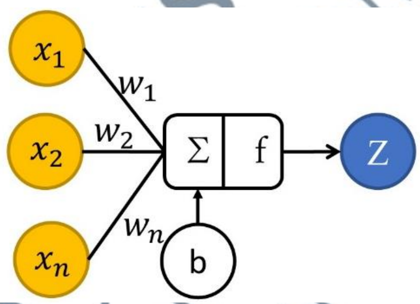


Figure 2.2-2: The relationship between input and output cells.

$$Z = f(x, w + b) = f\left(\sum_{i=1}^{n} x_i^T \omega_i + b\right) \quad (2.1)$$

where $Z$ is the output, $x$ is the inputs, $w$ is th weight, $b$ is the bias, and $n$ is the number of the input for the final node.

Other than that, there are four main techniques of the ML that can be used in ML learning, namely Regression, Decision Tree, Random Forest, and Classification. Regression is a process of forming a mathematical pattern that can be used to foresee one variable by another variable or variables. Decision Tree is a figure that applies a branching method to demonstrate every possible outcome of a decision while Random Forest will randomly pick out data and certain variables to construct several decision trees and then the outcome is averaged. Lastly, classification is identifying the data to which of a set of categories belongs to with the basis that the category of training dataset known. Usually, the models have a combination of techniques used in an algorithm. The study will further explain two subset of ML which is traditional ML and Deep Learning.

## 2.3 Traditional Machine Learning

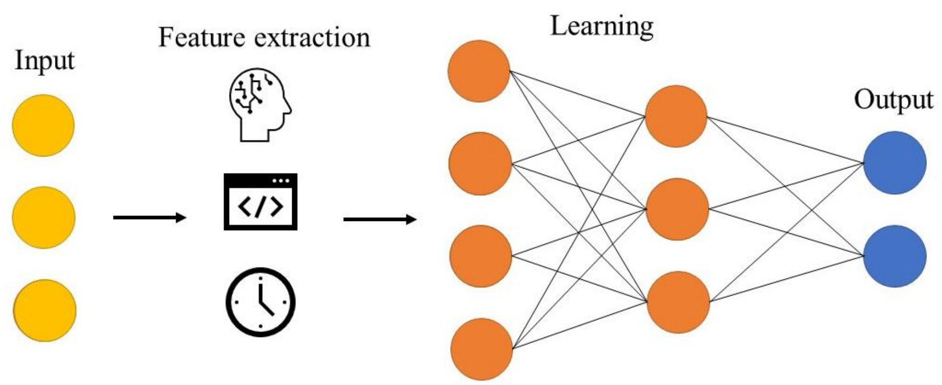


Figure 2.3-1: Traditional Machine Learning

Traditional Machine Learning (ML) is the innovation of AI which can alter the algorithm without human involvement to generate preferred output by training itself through structured input. The main idea of how traditional ML works is that the input will undergo a computational algorithm to create results as shown in Figure 2.3-1. Within the computational algorithm, the researchers have to extract features manually so that the input will train to undergo the technique explained in Section 2.2. Some examples of traditional ML models are Random Forest (RF), Super Vector Machine (SVM), naïve Bayes (NB), and extreme gradient boosting algorithm (XGB). This study focuses on three of traditional ML algorithms that is explained in detail below.

### 2.3.1 Logistic Regression

Logistic Regression (LR) is used for binary classification. When there is only binary output exists, the probability of certain class or event will accumulate at the value of 0 and 1. There is no probability values in prediction can be obtained in between the range thus it is too rigid. Therefore, to solve the issue, the probability function is as shown from Equation 2.1 below to give more accurate values. The Z value will be a log odds of the event $ln(\frac{P}{1-P})$ where, $P$ is the probability of event (Prabhakaran, 2017). From Figure 2.3-2, it demonstrates the difference between linear regression and logistic regression in fitting the model.

$$Z_i = \ln\left(\frac{P_i}{1-P_i}\right) = \beta_0 + \beta_1 x_1 + .. + \beta_n x_n \quad (2.1)$$

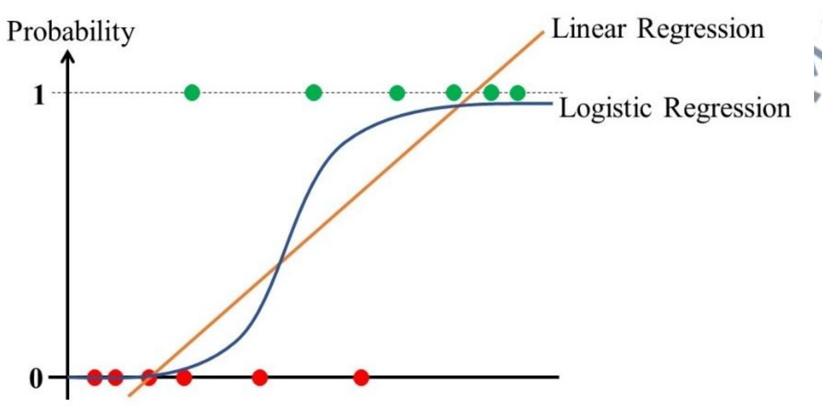


Figure 2.3-2: Logistic Regression.

The benefits of using logistic regression in prediction is that it is user friendly in terms of the execution and interpretation (Rout, 2020). Since it is only for binary classification, there are no assumptions made in features extraction, and thus the coefficients of the model become the feature indicator. Other than that, logistic regression has less tendency for overfitting but are expose when using a very big dataset.

Nevertheless, the drawback of using logistic regression is that it generates linear boundaries. Thus, it can only be used to predict discrete functions. Hence, it has a low performance on non-linear data. Additionally, the model has to meet the assumption that multicollinearity between the independent variables does not exists and also no outliers in the data.

### 2.3.2 Support Vector Machine

Support Vector Machine (SVM) is supervised learning model that analyse data used for classification and regression analysis. It is similar to probabilistic approaches and generates non-overlapping partitions and employ attributes (Jatain & Ranjan, 2017). SVM is also under neural network. It also works like a black box because the work behind it cannot be seen. For SVM, the main objective is to find an optimal margin that distinctly classifies the data points. It supports the vector to maximize the margin of the model fit or the hyperplane. There are different types of kernel that can be used for SVM. To name a few, linear, polynomial, and radial basis kernel. Figure 2.3-3 below shows the different kernel that SVM used in finding the optimal margin.

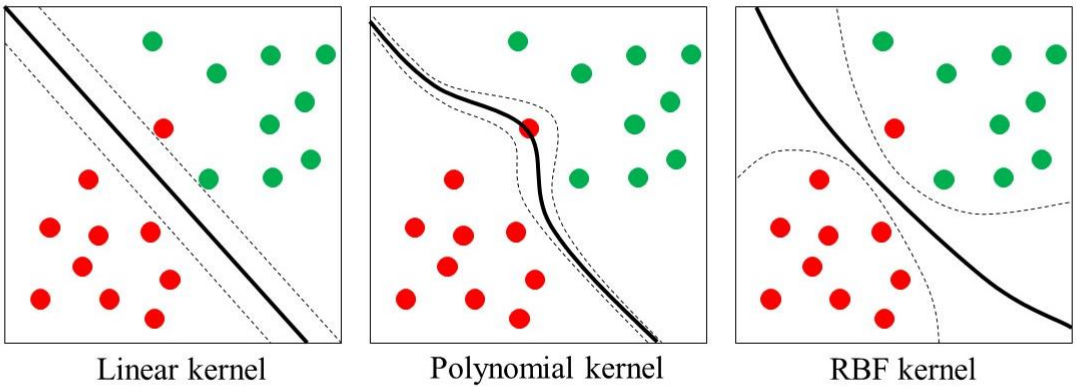


Figure 2.3-3: Support Vector Machine.

The advantage of using support vector machine is that it works well for all types of data either structured or unstructured data. The best part of using SVM is the number of kernel available. There are many types to choose from to manipulate the data into preferred results. Therefore, it is useful in solving complex problem. It also works well with high dimensional data. SVM also practices generalization, thus it reduces overfitting.

The disadvantages of using SVM is that usually the training time is long for large datasets. Since there are a number of kernel available, it will be a bit challenging to find the best kernel to use for the model to work (Kumar, 2019). Other than that, it is difficult to interpret the final variable weighs and model unless the model is using linear kernel. Hence, it is impossible to use the parameters in SVM model to evaluate the feature importance. Lastly, it does not do well for overlapped classes.

### 2.3.3 Extreme Gradient Boosting

Extreme Gradient Boosting (XGB) is similar to gradient boosting framework but more efficient. It is used to analyse data for classification and regression in vector form which generates a prediction model typically in decision trees form and generalizes them by allowing optimization of an arbitrary differentiable loss function (Nishida, 2017). The model is trained in succession by boosting that is using an iterative approach. Each time the model is training, it will make an improvement by correcting the previous errors and then add the model to ensemble. Therefore, the final prediction is consisted of all the corrections made from previous training. Figure 2.3-4 below shows how the model works.

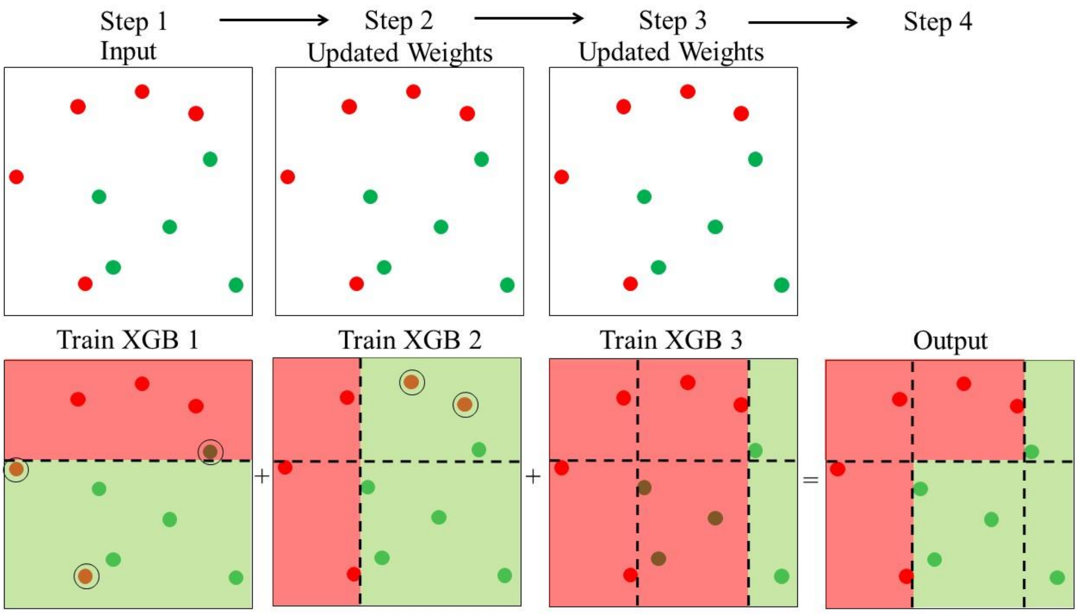


Figure 2.3-4: Extreme Gradient Boosting.

Among the pluses of using XGB include the training time is fast, work well with large datasets and it has a high-performance gradient boosting tree model. XGB also has in-built regularization which helps in preventing the model from overfitting (Heuer, 2019). It also has the capability of handling null values and the outliers have minimal influence on the model. Unlike SVM that needs to have a normalized data to work, the data used for XGB can be either normalized or not normalized for the model to work.

The minuses of using XGB is that the model must have enough data to function itself. Next, the data is exposed to overfit if the parameter for the trees are too deep or there are too many hyperparameters. Other than that, XGB has a semi-black box properties, whereby it is tricky for interpretation and hard for visualization.

## 2.4 Deep Learning

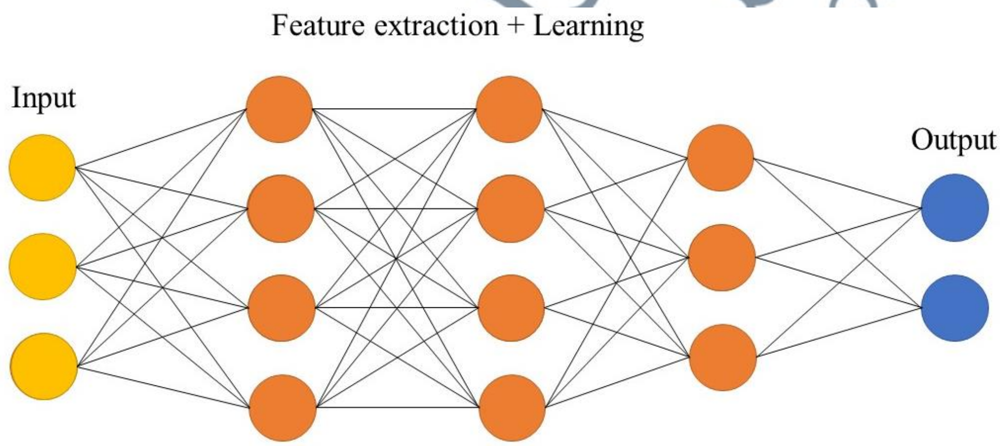


Figure 2.4-1: Deep Learning

Deep learning or Deep ML is when the algorithm train by self-learned to create the predictions, or in the other words it extracts the features automatically rather than manually extracted by the programmer. Deep Learning is an application of algorithms stimulated by the structure and function of the brain's neural and also known as unsupervised learning. Deep Neural Network (DNN) is a subset of deep ML where the numerous layers of the artificial neural networks (ANN) are designed so that each layer delivers a different interpretation of the input dataset (Sze et al., 2017). Difference between DNN with ANN is the number of hidden layers. ANN usually have one hidden layer while DNN have more layers thus the name called deep.

As shown in Figure 2.4-1, in deep learning, there are input layer (yellow), output layer (blue) and hidden layers (orange) in between the input and output that can make up from two to hundreds of layers. Examples of deep ML models are Deep Belief Network (DBN), gated recurrent unit (GRU), Long-Short Term Memory (LSTM), and Recurrent Neural Network (RNN). The study focuses on three of deep learning algorithms that is explained in detail below.

### 2.4.1 Deep Belief Network

Deep Belief Network (DBN) is an algorithm that utilise probabilities and unsupervised learning to generate outputs. DBN applied Greedy learning algorithm which uses layer by layer method for learning the top-down, generative weights (synapses). Therefore, each layer receives the data in different form, where the previous output layer becomes the input for the current layer and the output of the current layer becomes the input of the next layer. Also, to note, every pair of layers in the hidden including the input layers consist of Restricted Boltzmann Machines and Pretraining (RBM) that is considered as a binary version of factor analysis. RBM is a generative stochastic ANN that has the ability to learn probability distribution over its set of inputs. Figure 2.4-2 below shows the framework of DBN.

The advantage of using DBN is that because it uses greedy algorithm, it has the ability to find a good set of model parameters in a short amount of time (Agarwalla et al., 2016). Other than that, it is efficient when handling a big set of unlabeled data hence it can be pre-trained to learn in unsupervised fashion. Lastly, it facilitates learning better since the parameters in all layers can be optimized jointly.

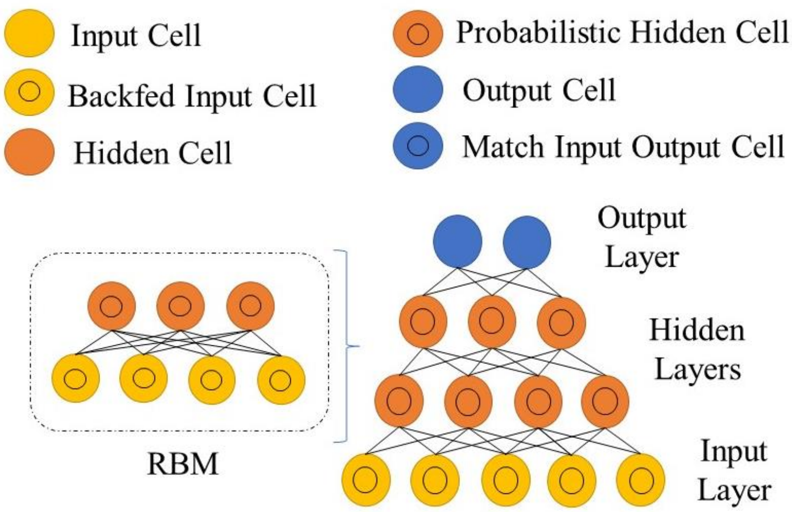

Figure 2.4-2: Deep Belief Network Framework.

The disadvantage of using DBN is that if there are outliers in the first layer, it will affect all the other layers since it practices greedy algorithm. This is because the algorithm is suboptimal, where it learns one layer at a time and thus, it cannot adjusted the lower-level parameters.

### 2.4.2 Multilayer Perception

Multilayer Perception (MLP) is a class of feedforward artificial neural network (FFNN) that utilizes backpropagation for training. In terms of network structure, MLP is similar to DBN but different in training the model. It has multiple layers and non-linear activation (cell) as compared to one single linear perceptron. The layers are all fully-connected where all the output cells consisted of weighted sum of all input cells. In other words, fundamentally it has multiple layers of perceptron and thus, each cell in each layer is connected to each other. Figure 2.4-3 below shows the framework of MLP.

The benefit of using MLP is that it is flexible and can be used for both regression and classification problems. It is also good in handling nonlinear data with large number of inputs thus it is reliable when the task involved have many features (Ciaburro & Venkateswaran, 2017). Other than that, it is fast in prediction once trained.

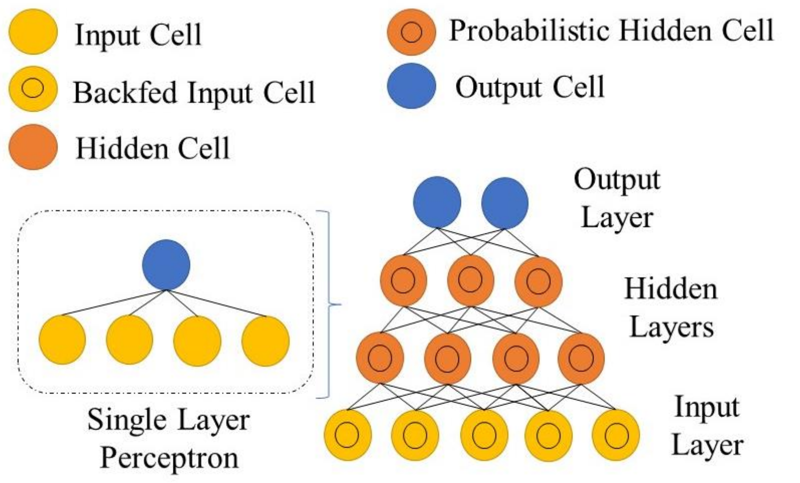


Figure 2.4-3: Multilayer Perceptron Framework

The drawback of using MLP is that it depends on the training data a lot. It requires a considerable amount of storage and computation (Sze et al., 2017). Hence, the risk of over-fitting and generalization could occur. The influence of the dependent variables from the independent variable are pretty much unknown since it is neural network which have a property of a black box. This can be another shortcoming of MLP model under deep learning.

### 2.4.3 Stacked Auto-Encoder

Stacked Auto-Encoder (SAE) is stacked of encode and decode of autoencoders to sort a deep network by feeding the latent representation found on the layer below as input to the current layer. There exist numbers of encoding and decoding layers in between the input and output. First stage, each layer is trained to minimize the error by reconstructing its input. Next, in fine-tuning stage, logistic regression layer is applied to output layer. The derived layer is trained like multilayer perceptron. Figure 2.4-4 below illustrates the framework of SAE.

Since SAE can be trained without generating samples as compared to RBMs, it helps speed up the training time (Vincent et al., 2010). Therefore, it become a popular alternative to DBNs. SAE also is an unsupervised algorithm, and thus it does not require a labelled data.

Having said that, SAE is less effective when errors are present in the first layer (Voulodimos et al., 2018). Consequently, the model may reconstruct the average training data.

Therefore, the data should undergo denoising to prevent the model to reconstruct the average training data. Similar to MLP, it also depends on huge the training data.

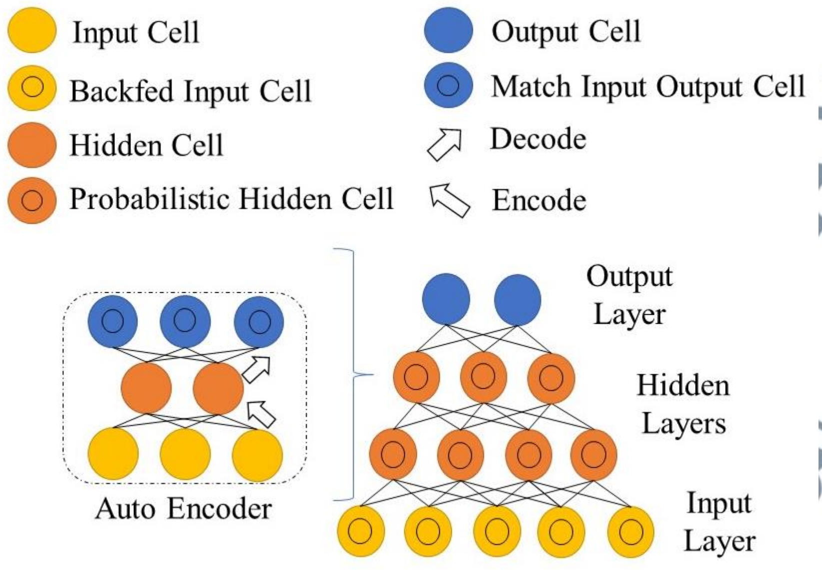


Figure 2.4-4: Stacked Auto Encoder Framework.

## 2.5 Differences between traditional ML and deep learning

The main difference between traditional ML and deep learning is the presentation of data to the system. The training dataset for traditional ML is small compared to deep learning. Therefore, traditional ML always require structured data or need more classifiers or features, while deep learning depends on layers of ANN that do not need to have many features.

Sometimes traditional ML learn to understand data through human involvement especially when the actual output is not the anticipated one. Deep learning however does not require human intervention, as they learn through their own errors (Kapoor, 2019). However, to improve the quality of the model, deep learning requires more data points as compared to traditional ML. Other than that, the training time for traditional ML is shorter than deep learning. It is due to the traditional ML's ability to learn through pre-programmed specified parameter while deep learning only able to distinguish the differences or concepts within layers of neural networks when subjected to millions of data.

## 2.6 ML model that works best

Based on recent research shown in Table 2.5-1 below, the results show mixed findings of between the ML models used, traditional ML and deep learning, that works better in forecasting. Table 2.5-1 shows the research background in recent studies.

Table 2.6-1: Recent studies on the application of ML models in stock market predictions.

| Authors, Year | Assets | Market Geography | Algorithms Used | Result Metric |
|---|---|---|---|---|
| Nabipour et al., 2020 | 4 group of stocks – Tehran Stock Exchange | Iran | Traditional ML{DT, RF, GB, XGB Bagging, Adaboost}, Deep Learning{ANN, RNN, LSTM} | MAPE, MAE, RMSE, MSE |
| Butt et. al, 2019 | 3 Commodities Prices, 2 Microstructure market variables | Malaysia | Traditional ML{RF, SVM}, Deep Learning{ANN} | RMSE, MAE, RAE, MSE |
| Dongdong et. al, 2019 | 424 – S&P500 (stock), 185 – CSI 300 (stock) | United States, China | Traditional ML{LR, SVM, XGB, RF, NB, CART}, Deep Learning{GRU, RNN, LSTM, MLP, DBN, SAE} | AR, RR, PR, F1, ARR, WR, ASR, MDD |
| Soon et.al, 2018 | 1 KLCI (stock) | Malaysia | Deep Learning{FFNN, RNN } | MSE, AR |
| Chong et al., 2017 | 38 – KOSPI (stock) | Korea | Traditional ML{AR*}, Deep Learning{ANN, DBN,MLP, PCA, SAE} | NMSE, RMSE, MAE, MI |
| Sagir & Sathasivam, 2017 | KLCI (index) | Malaysia | Traditional ML{SVM}, Deep Learning{ANN} | MSE, Rsqrt |
| Yong et al., 2017 | STI (index) | Singapore | Deep Learning{MLP} | RMSE, MAPE, PP, SR |
| Agarwalla et al., 2016 | 3 stocks – National Stock Exchange | India | Traditional ML{SVM, RF}, Deep Learning{LSTM, MLP} | AR, MSE |
| Dey et al., 2016 | 2 individual stocks | United States | Traditional ML{LR, SVM, RF, XGB}, Deep Learning{ANN} | AR, RR, PR |
| Krauss et al., 2016 | 200 - S&P 500 (stock) | United States | Traditional ML{RF, GB}, Deep Learning{DNN} | R, STD , SR, AR |
| Dash & Dash, 2016 | BSE SENSEX, S&P500 (index) | India, United States | Traditional ML{SVM, NB, kNN, DT}, Deep Learning{ANN[FL]} | PP |

| | | | | |
|---|---|---|---|---|
| Phooi M'ng et al., 2016 | KLCI (index) | Malaysia | Deep Learning{ANN} | R, ARR |
| Yiing & Thim, 2015 | KLCI (index) | Malaysia | Traditional ML{AR*}, Deep Learning{ANN} | MSE, RMSE, MAE, MAPE |
| Monfared & Enke, 2014 | NASDAQ (index) | United States | Traditional ML{GARCH}, Deep Learning{RBF, GARCH-ANN} | MSE, MSE Reduction |
| Yao et al., 1999 | KLCI (index) | Malaysia | Traditional ML{AR*}, Deep Learning{ANN} | NMSE, AR, R, ARR |

**ANN**: Artificial Neural Network, **AR**: Accuracy Rate, **AR***: Autoregressive, **ARR**: Annualized Return Rate, **ASR**: Annualized Sharpe Ratio, **CART**: Classification & Regression Tree, **DBN**: Deep Belief Network, **DT**: Decision Tree, **F1**: F1 Score, **FFNN**: Feed-Forward Neural Network, **FL**: Functional Link, **GARCH**: Generalized Autoregressive Conditional Heteroskedasticity, **GB**: Gradient Boosting, **GDA**: Gaussian Discriminant Analysis , **GRU**: Gated Recurrent unit, **kNN**: k-Nearest Neighbor, **LR**: Logistic Regression, **LSTM**: Long Short-Term Memory, **MAE**: Mean Absolute Error, **MAPE**: Mean Absolute Percentage Error, **MDD**: Maximum Markdown, **MLP**: Multilayer Perception, **MSE**: Mean Squared Error, **NB**: Naïve Bayes, **NMSE**: Normalized Mean Squared Error, **PCA**: Principle Component Analysis, **PP**: Profit Percentage, **PR**: Precision Rate, **R**: Return, **RAE**: Relative Absolute Error, **RBF**: Radial Basis Function, **RF**: Random Forest, **RMSE**: Root Mean Squared Error, **RNN**: Recurrent Neural Network, **RR**: Recall Rate, **Rsqrt**: R Squared, **SAE**: Stacked Auto-Encoder, **STD**: Standard Deviation, **SVM**: Support Vector Machine, **XGB**: Extreme Gradient Boosting, **WR**: Winning Rate

Based on previous studies that varies in terms of assets, market geography, algorithms used and research methodology, there are two groups of evaluation indicators that can be grouped from the result metric which are directional and performance. The evaluation indicators are used to evaluate the efficiency of ML models. The first group is by using directional evaluation indicators which focus more on the efficiency of the model itself in classification and/or regression when predicting the stock price. Second is by using performance evaluation indicators which focus more on the efficiency of the models as trading strategy in term of profitability and risk assessment.

When evaluating the directional evaluation indicator in predicting whether the stock price will go up or down, most study commonly used Accuracy Rate (AR) or Mean Squared Error (MSE) as result metric to evaluate the performance (Agarwalla et al., 2016; Dongdong et. al, 2019; Krauss et al., 2016; Nabipour, et al., 2020). Some even used F1 Score since it is said to be a better indicators compared to AR (Huigol, 2019). AR is the measure of the degree of closeness of a measured or calculated value which is used in classification. A high AR indicates that it has a better number of predictions. Usually, researchers chose AR because it is easier to compute compared to MSE. MSE on the other hand, is the average squared difference between the estimated values and what is estimated which is used in regression. A low MSE indicates that the prediction values reproduce the reality. However, MSE is prone to outliers. So, if there is one outlier data, it could jeopardise the accuracy of the overall results. To see the efficiency of the model by using directional evaluation indicators, it depends on how the data are interpreted to shows how significant the results are.

Other than that, the efficiency of the model can also be evaluated by using performance evaluation indicator of whether the models have the ability to produce profit and assess the risks in trading strategy. There is different type of trading performance evaluation indicators that are available, and each have different interpretations and approaches. As an example, some used Sharpe Ratio (SR), Winning Rate (WR), Annualized Return Rate (ARR) or Profit Percentage (PP). Thus, it depends on the result metric of the research.

The first finding from previous studies stated that traditional ML has better efficiency compared to deep learning. From the most recent study, it determined that that the performance of traditional ML is better in most of the directional evaluation indicators compared to deep learning (Dey et al., 2016; Dongdong et al., 2019; Nabipour, et al., 2020) while the performance evaluation indicators show not much different in efficiency (Dongdong et al., 2019). Similarly, a study in Malaysian commodities derivatives and microstructure market variables showed that RF have a superior performance than SVM and ANN (Butt et. al, 2019) which means traditional ML are better than deep learning. It is also emphasized that the larger dataset for training the model can positively enhance the prediction of exchange rate. In the same way, a study by Krauss et al. (2016) found that RF in directional and performance evaluation indicators outperform gradient-boosted trees and DNN in their application, making it the method of choice in light of noisy feature space.

Next, the second finding from previous studies stated that deep learning has better efficiency compared to traditional ML (Monfared & Enke, 2014; Yiing & Thim, 2015; Sagir & Sathasivam, 2017). They noted that due to multiple layers in DNN, the outcome is better in forecasting the stocks based on results for directional evaluation indicators (Chong et al., 2017). Some researchers also suggested that it is useful to explore and develop more rebost algorithm like ANN with technical indicators in forcasting returns (Phooi M'ng et al., 2016; Yao et al., 1999). A case study on Malaysian stock price prediction which focuses on deep learning algorithm under ANN showed that both FFNN and RNN performed well by achieving more than 90% prediction accuracy (Soon et. al, 2018). A study by Yong et al. (2017) on the other hand, found that DNN produces high profit however yeilds less accurate results as the model predict further into the future.

Lastly, the finding from previous studies stated that a combination of traditional ML and deep learning could give a better performance as compared to individual algorithms in forecasting a stock (Chong et al., 2017; Dash & Dash, 2016; Krauss et al., 2016). This is supported by a study from Kristjanpoller et al. (2014) that showed that the performance of GARCH models improved with ANN models and hence producing robust and consistent results for different specifications and volatility measures. It is also noteworthy that the ensemble models from traditional ML with deep learning are more useful in extreme event forecasting since the structure of volatility process becomes more complex (Monfared & Enke, 2014).

The results from past findings of three traditional ML models and three deep learning models are plotted as shown in Figure 2.6-1. It can be seen that for directional evaluation indicators namely Accuracy Rate (AR) and F1 Score, traditional ML (three models on the left) worked better than deep learning (three models on the right) while for performance evaluation indicators namely Winning Rate (WR) and Annualized Return Rate (ARR), deep learning algorithms worked better than traditional ML. However, there is not much different in values for the performance evaluation indicators. Note that each finding has different features, number of data set, different techniques in data analysis and different evaluation indicator used. The MSE values are not included in Figure 2.6-1 because the values vary by ten to the power of 3. There are a lot of ML algorithms that is available and being used in forecasting. The finding differs from one study to another. Therefore, this study focuses on stock prediction specifically on Malaysian stocks to assess the efficiency of the selected ML algorithms.

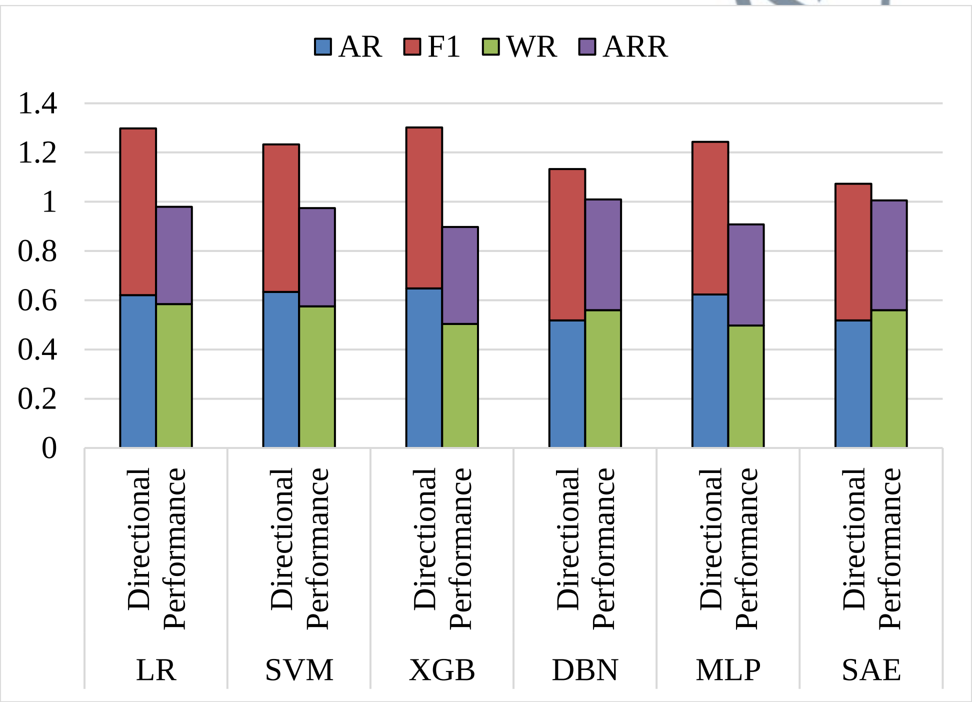


Figure 2.6-1: Comparison of Evaluation Indicators on ML models used in Literature Review.

## 2.7 Conclusion

Deep learning is still on the rise together with traditional ML. The main difference between traditional ML and deep learning is the presentation of data to the system. Traditional ML must have a human involvement in coding or features input to complete the work whereas deep learning can self-learned as it requires less human involvement in the programing. In conclusion, there is mixed finding on which ML models work better in forecasting. Based on previous studies, the findings are divided into three insightful findings, where (1) traditional

ML works better than deep learning, (2) deep learning works better than traditional ML, and (3) the combination of both traditional and deep learning model works better than individual algorithms. It is worth mentioning that mixed or hybrid between traditional ML and deep learning create a better model in forecasting.

Based on literature review, there are few studies that analyse on both traditional ML and deep learning models. Moreover, some studies concentrate on the stock index's prediction or select a few stocks with restricted features based on their own preferences. Hence, it is beneficial to explore ML models with technical indicators in forecasting stock movement so the findings can contribute to the extension of ideas and data analysis in stock trading strategy. Therefore, this study aims to find out the efficiency of both traditional ML and deep learning as trading strategy in stock movement prediction by assessing the evaluation indicators.