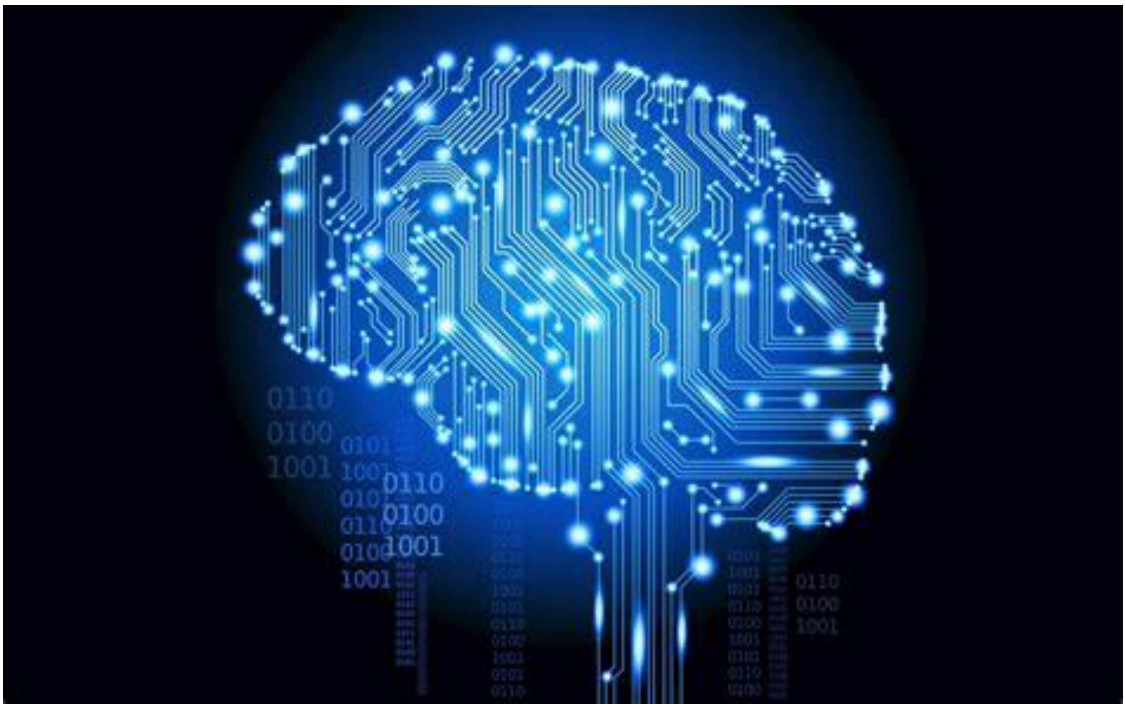
ASSESSMENT 2
STATISTCS & DATA ANALYTICS

ELI BELLEZZA



CONTENTS

Section 1: Introduction and Business Context

The business I am representing in this report is Finco. Finco is a UK based investment manager. They invest money into the stock market on behalf of their clients and take a fee for providing this service. The majority of their clients are classified as retail customers, meaning they are non-professional private citizens, businesses and families who are not regarded, from a regulatory point of view, as being sophisticated investors. Though in this business context Finco is a hypothetical firm, it is a space I have worked in, so I do have an understanding of this type of company and the challenges they face. Most companies in this space operate using a similar model, it is the returns that differentiate them, they are mostly similar in all other instances.

The data analytics Finco currently performs can be separated into two broad categories: traditional and big data analytics. Finco's traditional data can be broken down into local data based on the output of the company. These sources include CRM records, client trading logs and employee KPI. Analytics performed on this traditional data include analysing and then forecasting sales targets and predicting client returns for the year.

Big data analytics are performed on sources such as stock market data and economic news. Examples are assets pricing data and central bank forecasting. This data is integral to Finco because it drives trading decisions and ultimately revenue generation. Here models are created from the huge expanse of financial data as well as the trading patterns of other institutions being analysed.

The CRISP-DM methodology could help Finco better meet its strategic priorities with respect to data analytics and business intelligence by providing a clear pathway to assessing the relevance of data mining applications. It is described in a report written by the creators of the methodology as, "CRISP-DM succeeds because it is soundly based on the practical, real-world experience of how people conduct data mining projects." (Chapman et al, 200, p2)

When looking to generate returns in the stock market, an industry term is used: 'edge'. Edge can be described as something which gives your trading the ability to beat other traders in the market. That something could be a strategy, technique, approach or piece of knowledge. As trading is a zero-sum game, edge means you are 'winning' the majority of your trades whilst also meaning you are 'beating' the majority of participants in the market. A useful analogy is that of a casino and the phrase 'the house always wins. We all know this means that over the long term the house, i.e. the casino, will win the games that its patrons participate and bet on. Just like trading in the stock market, you cannot win every single trade or bet but with an edge you can ensure you win more often than losing, thus ensuring profitability over the long term. If the casino can its patrons gambling for an extended period, they know they are likely to win overall.

Regarding this report, I will be exploring whether it is possible to give traders at Finco an edge in their trading by using machine learning to predict the future move of a stock price. The machine learning method will be applied to historic stock price data. In this case we will using data from electric vehicle manufacturer, Tesla (ticker: TSLA).

If the implementation proves effective it could certainly open an worthwhile pathway for Finco to explore whether implementing machine learning techniques in their own financial analysis is feasible.

Section 2: Data Selection and Pre-Processing

The data I have chosen for this exercise is the historic stock price data from Tesla (TSLA). The data spans a 10-year period, and each instance is the pricing of TSLA for a given trading day. The dataset includes data for every trading day from 29/06/201 until 03/02/2020. To clarify, a trading day is a day on which the stock index Tesla is listed on, the NASDAQ, is open for trading. These hours are Monday through Friday from 9:30am until 4pm. As well as weekends, the exchange closes on major American holidays such as Christmas day and Independence Day.

The dataset was obtained from the free online resource Kaggle. Kaggle is an open-source website where any user can upload datasets amongst other files. There is a social media element which draws users to the site because people with an interest in computing can communicate and work through problems together. A URL link to the dataset will be published in the bibliography.

The below table, Figure 1, describes the 7 attributes of the dataset.

Fig1 - Table of Attributes description.

Attribute Name	Description	Data Type
Date	Trading day on which the stock price data was gathered	Date – formatted as dd/mm/yyyy
Open	The price of TSLA at the opening of the market on the related trading day	Integer or Floating Point representing the price in USD($)
High	The highest price TSLA achieved on the related trading day	Integer or Floating Point representing the price in USD($)
Low	The lowest price TSLA achieved on the related trading day	Integer or Floating Point representing the price in USD($)
Close	The price of TSLA at the closing of the market on the related trading day	Integer or Floating Point representing the price in USD($)
Adjusted Close	The adjusted close price factors in corporate actions TSLA may have taken. It is meant to be a better reflection of the true value of the stock. Actions could be dividend payments or additional share offerings.	Integer or Floating Point representing the price in USD($)
Volume	How many shares of TSLA were traded on the related trading day	Integer representing the total number of TSLA shares traded

The data for tables below, Figures 2 & 2.1 was taken from WEKA. In the Explorer interface of the programme, the key values listed in these table are automatically calculated.

Training Set Key Values Table – Fig 2

Attribute Name	Minimum Value	Maximum Value	Average Value	Standard Deviation Value
Date	not applicable	not applicable	not applicable	not applicable
Open	16.14	287.67	132.364	94.283
High	16.33	291.42	134.692	95.97
Low	14.98	280.4	129.919	92.828
Close	15.8	286.04	132.352	94.289
Adjusted Close	17.05	286.04	132.352	94.289
Volume	118500	37163900	4270591.366	427619.776

Test Set Key Values Table – Fig 2.1

Attribute Name	Minimum Value	Maximum Value	Average Value	Standard Deviation Value
Date	not applicable	not applicable	not applicable	not applicable
Open	17.94	575.69	185.591	120.066
High	18.45	589.8	188.726	122.068
Low	17.66	567.43	182.018	118.038
Close	18.32	580.99	185.351	120.219
Adjusted Close	18.32	580.99	185.351	120.219
Volume	201100	33649700	5707274.345	5136648.753

Figure 3 – Overall list of values

Number of Instances	Missing Values	Outliers/Erroneous Values
2416	0	0

As we can see from Figure 3 above, the data is already of a very high quality. With no missing values and outlying values, the level of pre-processing will not have to be extensive. This is typical of stock data. Large-cap or ‘blue-chip’ companies like Tesla will be traded on huge global exchange, in this case the NASDAQ. With large stock exchanges such as these data collection is automated and always of the highest quality. Inaccurate or missing pricing data could have grave consequences for the global economy, so it is imperative that the data is of the highest quality.

I did not use any instance weighting as the dataset did not have any significant or outlying values that needed to be considered. A scenario where weighting may have had to be introduced could be a stock market crash or major economic event that caused the stock price to move to the extreme. In a case like this, weighting would be used to ensure that the one-off extreme volatility was not taken into account by the model as ‘typical’ behaviour of the stock price. Weighting can be implemented through the pre-processing tab of the WEKA Explorer Interface.

The table below, Figure 4, shows how many instances in the training set and the test set, as well the percentage split of the total set for each subset.

Fig 4 – Subset breakdown

Sample Type	Dataset	Number of Instances
In Sample	Training Set (70% of total data)	1659
Out Sample	Test set (30% of total data)	725

To create the subset split, the RemovePercentage filter through the WEKA Explorer interface was used. This allows a user to customise the percentage split for their data as well crucially ensuring there are no repeat instances in both sets. In this case, the training set is the top 70% of the dataset and the test set is the inverse 30%.

The 'curse of dimensionality' issue is something which needs to be taken into consideration when doing machine learning. This issue is best described as, "Essentially meaning that there is a threshold that after being crossed by the model, the predicative accuracy decreases. This threshold is crossed when the model has too many attributes to process.

One of the steps I have taken toa address this issue is to only look to predict the close price of TSLA. The model would be at risk over training if asked to predict multiple values and accuracy would drop. I chose to remove 'Adjusted Close' attribute. The reason for this can be seen in Figures 2 & 2.1. The 'Adjusted Close' attribute has the same significant values as the 'Close' attribute. This indicates its usefulness in adding meaningful depth to the dataset is limited. Another way the issue was mitigated was in the decision to predict only one attribute out of 6 numerical attributes, the Close price of TSLA. The reason this value was chosen was because it is the most relevant pricing figure as it is often used in the production of stock price charts and is seen as the most accurate valuation of a company. It is also the price of the last transaction made before the market close, thus providing the best reference price for formulating the next day's trading strategy. The model had highest probability of predictive accuracy when only needing to predict one attribute. The ability to remove an attribute is part of the WEKA Explorer interface, which is where most pre-processing is conducted within the program.

Section 3: Machine Learning Method(s) and their Implementation

In order to predict the future stock price of Tesla, I will be using regression. Regression is commonly used for predicting numerical outcomes. It describes or predicts the relationship between to variables. If there are multiple variables, this is called multivariate regression as opposed to univariate regression. The values used to predict the variable are called independent variables and the predicted variable is known as the dependant variable. An error term is an important part of a regression equation. The error term represents the difference in the reality of the independent variables with the statistical model created. It considers the potential variation in the dependent variables. This is important because the independent variable may not take into account certain parameters when building the model. In visual terms it may be the points on a scatter plot on a graph through which the regression line goes.

This capability is best illustrated by the following equations published the Siew and Nordin research into predicting stock prices using WEKA.

In a regression model, a predicted value Y is related to a
function of x and b.

$$Y = f(x, b)$$

Y is the dependent variable; x is the independent variable
and b is the unknown parameter.
A linear regression model.
takes the form:

$$Y = b0+b1\ x1+ \ldots +bnxn+ e$$

where x1 to xn are independent variables, and e is called the
error term. A linear regression equation written in vector form.
[10] is:

$$Y= a+bx+e$$

(Siew & Nordin, 2012, p4)

There are weaknesses to regression that are necessary to highlight. Chief among them is

I will be using WEKA to implement the machine learning methods onto the data I have selected. It can best be summarised as an open-source machine learning and data mining implementation programme through which users can run models of their choosing. Analysis is provided through WEKA and there has a high degree of flexibility combined with an ease of usability.

I downloaded the Timeseries processing tool from WEKA package manager. The tool is described as follows, “Class that implements time series forecasting using a Weka regression scheme.” (WEKA) This tool is ideal for implementing regression for numerical forecasting. It is of particular use in this case for predicting stock data as it allows for the input of a date-based time frame on which the model will output its forecast.

In order to select the appropriate model for implementing regression I chose to implement cross-validation through WEKA. The output data could then be studied to see if the model should be chosen for implementation on the chosen task. I used the process of 10-fold-validation in the WEKA Explorer Interface, under the Classify tab. The benefit of fold validation, whether 10-fold or K-fold is that sample data is taken from across the entire dataset, thus ensuring any bias included when selecting the subsets is eliminated.

Cross validation is particularly useful in assessing the accuracy of predictive models due their ability to limit overfitting. Overfitting is where the prediction too closely follows the existing data thus resulting in the protection being of high bias and low variance. In order to practice cross validation, a test must be selected from the dataset. It is crucial there are no overlapping values in either the training or the test set so when it comes to implementing the model on alternative datasets, the accuracy of the model can be correctly ascertained.

The metrics output from the test data, such as correlation co-efficient and mean absolute error allow for assessment to the model’s accuracy on different data sets.

Below is a description of each model implemented and the processes they use to make the prediction.

Linear Regression: Here a line created through the estimation of coefficients is created in order to predict the future of a numerical value in relation to the underlying data. The line is the relationship between the independent and dependant variables. This is a pre-computing form of prediction that still to this day is used in predicting numerical outcomes. It is particularly useful for financial data because there is a linear correlation between data points.

MultiLayerPerceptron: This is a neural network that is best described in WEKA. A classifier that uses backpropagation to learn a multi-layer perceptron to classify instances. The network can be built by hand or set up using a simple heuristic. The network parameters can also be monitored and modified during training time. The nodes in this network are all sigmoid (except for when the class is numeric, in which case the output nodes become unthresholded linear units). (WEKA)

SMOReg: This method implements vector regression through a SVM (Support Vector Machine). This allows the user to customise the level of error in the model. One strength over linear regression is ability to provide a more accurate prediction over data with a large variation of values. The output of this model is best described in a paper conducting using the same techniques to predict share prices on an Indian stock exchange (NSE), “This implementation globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes by default. Thus, the coefficients in the output are based on the normalized/standardized data, not the original data.” (Padhye and Karuna, 2016,p80)

With this model I changed the hyperparameters under the filterType to standardise training data.

M5P: This method creates a tree. The leaves of this tree can be seen as various linear regression models, attached together (the branches). Strengths of this model include the ability to deal with missing and erroneous values. Pruning can be implemented, which has been in this case, to minimises the size of the tree to avoid over fitting or over training. Yang and Witten, Witten being the one the creators of WEKA, describe pruning as well a process called smoothing which aids prediction accuracy, “When pruning an inner node is turned into a leaf with a regression plane… to avoid sharp discontinuities between the subtrees a smoothing procedure is applied that combines the leaf model prediction with each node along the path back to the root, smoothing it at each of these nodes by combining it with the value predicted by the linear model for that node.” (Yang and Witten 1997)

RegressionByDiscretization: A regression scheme that employs any classifier on a copy of the data that has the class attribute (equal width) discretized. The predicted value is the expected value of the mean class value for each discretized interval (based on the predicted probabilities for each interval). (WEKA)

With this model the hyperparameters were changed, the most effective classifier was RandomForest, described as, “Class for constructing a forest of random trees.” (WEKA) This yielded the most accurate prediction.

GaussianProccesses: This method is best summarised in the research conducted by Pandhye and Karuna. All attributes are standardised, and normalised and nominal values are converted to binary ones. The strength of this model comes from ease of implementation. The hyper-parameters do not need to be tuned as with the other models. (Padhye and Karuna, 2016,p81) This is derived from the

Gauss-Markov Theorem, which, “implies that the least squares estimator has the smallest mean squared error of all linear estimators with no bias.” (Hastie, Friedman and Tisbshirani, 2017, p54)

With this model I changed the hyperparameters under the filterType to standardise training data.

Section 4: Evaluation of Results

Below is the table, Figure, listing the predictions the model made of the TSLA close price on.

Attribute	**Linear Regression**	**Multilayer Perceptron**	**SMOReg**	**M5P**	**Regression ByDiscretization**	**Gaussian Proccesses**
Predicted Close	260.8427	261.9861	261.6983	261.6682	264.4979	261.1552
Actual Close	261.5	261.5	261.5	261.5	261.5	261.5
% Error	0.25%	0.18%	0.075%	0.0064%	1.14%	0.13%

The evaluation of the can be further broken down into several key metrics used in regression models for machine learning. WEKA can output this data through output window in timeSeries Forecast interface. Metrics can be selected in the Advanced Configuration tab of the interface. A large number of metrics were chosen so that they could also be evaluated to see which was most effective in recognising the most accurate model.

Below is a descriptive table of each evaluation metric and their relevance to this particular task.

Metric	Description	Relevance
Mean Absolute Error (MAE)	The lower the value the better the prediction. This metric measures the difference between wo variables. These variables are continuous. The model studies it’s accuracy on the independent variables before outputting the dependent variables	Particularly useful when looking assessing prediction of linear data, such as stock data.
Mean Squared Error (MSE)	Measures the averages of the squares of the errors. It is expressed as a positive value	Uses are best suited to data that contains outlying data
Root Mean Squared Error (RMSE)	The lower the value the better the prediction. Measure the square root of the average of the squared difference between the independent and dependent variables	Useful when studying data with large errors.
Relative Absolute Error (RAE)	A value of less than 100 is considered accurate	Another metric to measure the accuracy of the absolute error in prediction
Root Relative Squared Error (RRSE)	Close or less and 100 is considered accurate. Calculated by dividing the RMSE by the RMSE obtained by predicting the mean of independent variables	

The below table lists the output values of each the metrics.

Evaluation Metrics

Metric	**Linear Regression**	**Multilayer Perceptron**	**SMOReg**	**M5P**	**Regression ByDiscretization**	**Gaussian Proccesses**
Mean Absolute Error (MAE)	2.7085	3.4862	2.6468	2.9641	5.7025	
Mean Squared Error (MSE)	19.1575	22.807	19.2353	20.995	54.1982	2.7658
Root Mean Squared Error (RMSE)	4.3769	4.7757	4.3858	4.5055	7.3619	4.4346
Relative Absolute Error (RAE)	101.7019	130.878	99.4016	111.3245	214.0186	103.8439
Root Relative Squared Error (RRSE)	99.3448	108.384	99.5489	102.2661	167.0431	100.652

From these metrics we can see that the SMOReg model was the most effective at predicting the future close price of TSLA. From the similar study conducted by Žmuk & Jošiæ, the findings were similar. This model was most effective at predicting the future stock price on a short-term horizon, I chose one day, they had chosen from 5 – 30 days. "[when the] base period length is reduced from 10 to 5 [days] and conclusions became not so straightforward. For NASDAQ and Nikkei 225 the most precise forecasting approach turned out to be SMOreg whereas for Dow Jones that is multilayer perceptron and for S&P 500 Gaussian processes. Those conclusions remained the same for all four observed forecast horizons." (Žmuk & Jošiæ, 2020, page 481). This finding correlates with this study's findings and is particularly interesting because it concludes that model accuracy can vary from index to index. In this case TSLA is listed on the NASDAQ and the prediction horizon was short: 1 day. This concurs with the findings from the Žmuk & Jošiæ of SMOReg being most effective on the NASDAQ over a short-term horizon.

SMOReg being the most accurate predictor of stock price was also concluded by Siew and Nordin in their research paper exploring the accuracy of stock price prediction using WEKA, "In this research which utilized the WEKA regression techniques, SMO Regression technique has outperformed the other regression techniques in the experiment." (Siew and Nordin, 2012, p 5)

In terms of the most accurate metric, Relative Absolute Error and Mean Absolute Error were the most accurate in demonstrating SMOReg as the most accurate model. Both metrics share and Absolute Error component. This is the magnitude of the difference between the actual value and the numerical prediction.

Section 5: Discussion

In summary it can be said that the models I implemented did learn the problem and produce satisfactory results. I felt 5 out of the 6 models I implemented gave an accurate predicted Close price of TSLA for 17/3/2017. The RegressionByDiscretization did not give an a particularly accurate Close price.

Though the efficacy of machine learning in prediction is well documented, it was still surprising to see the accuracy of the models first-hand. When I compare my results to the findings of other papers, where regression models through WEKA have been implemented to predict a future stock price, I can see my results are similar in their conclusions.

Before summarising the application of the results of this study regarding Finco, there is an important caveat which needs to be explained. Black Swan events are unforeseen happenings in the world which have an impact on the global economy and/or stock market. They are very difficult to predict and can often the negate modelling and forecasting that is conducted on the stock market. A relevant hypothetical example to this study could be an earthquake causing massive damage to a TSLA production facility. It could the US government deciding to impose huge taxes on electric vehicle manufacturers or even something more distantly related, like a credit crisis in Europe or a mineral embargo in Africa. If unpredictable yet inevitable events cause the market price to massively deviate, they may impact the accuracy of a machine learning model by creating huge atypical outlying values.

Another key point to consider is market manipulation or over/undervaluation. A stock may be trading at an incorrect price, whether through manipulation or by improper valuing due to a large number of buyers or sellers, driven by sentiment, causing the market to move. If the pricing input into the model is inaccurate, when the market corrects, as has always happened to date, the prediction output may be wildly wrong, thus causing a trader to make trades based on inaccurate data, therefore increasing the downside risk and chance of loss.

With regards to Finco there is a clear benefit for them to develop their edge in the market through implementing machine models to predict future stock price movements. From a risk management standpoint there is also value from adopting machine learning. Being able to predict the price of a stock not only provides an opportunity to make money in the market but will also enable Finco to plan risk mitigation strategies if their clients are holding stocks that are predicted to fall in price.

Machine learning falls under a form financial analysis called technical analysis. This is where charts and data are studied through scientific methods. The alternative form of financial analysis is called fundamental analysis. Here balance sheets, company earnings and statements, as well as financial news is studied. Both forms of analysis are integral to success in the trading the stock market. Though technical analysis, encompassing machine learning, has been responsible for most of the large-scale success in the market, chiefly through the field of quantitative finance, fundamental analysis is still very much crucial. This is mainly due to the uncontrollable yet inevitable factors I mentioned earlier: Black Swan events, investor sentiment and manipulation. A machine learning model is less likely to inform you of an impending tsunami than watching the news. Having said that, with human trading on the long-term decline in professional finance, companies like Finco cannot afford to neglect the opportunity to adopt machine learning methods into their business. With the ability to analyse and process a huge amount of financial data, better informed decision may be able to be made when formulating trading strategies for their clients. An increase in returns would certainly make their offering more attractive for prospective customers, which could in turn increase the number of incoming clients, thus increasing revenue by generating more fees from increased assets under management.

Bibliography

Dataset - Kaggle (2020) *Kaggle terms of service*. Available at: https://www.kaggle.com/timoboz/tesla-stock-data-from-2010-to-2020

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R., 2000. *CRISP-DM 1.0: Step-by-step data mining guide*. SPSS inc.

Hastie, T., Friedman, J. and Tisbshirani, R., 2017. *The Elements of statistical learning*. New York: Springer.

Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2016. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann: Saint Louis

Siew, L. & Nordin, Md Jan. (2012). *Regression techniques for the prediction of stock price trend*. 10.1109/ICSSBE.2012.6396535.

Wang ,Y., Witten, I. H.: Induction of model trees for predicting continuous classes. In : Poster papers of the 9th European Conference on Machine Learning, 1997.

WEKA, *WEKA terms of service,* Available at: https://weka.sourceforge.io/doc.stable/weka/classifiers/meta/RegressionByDiscretization.html

Berislav Žmuk & Hrvoje Jošiæ, 2020. "**Forecasting stock market indices using machine learning algorithms**," Interdisciplinary Description of Complex Systems - scientific journal, Croatian Interdisciplinary Society Provider Homepage: http://indecs.eu, vol. 18(4), pages 471-489.

Padhye.,S and Karuna.,G, 2016. *Regression Analysis for Stock Market Prediction using Weka Tool without Sentiment Analysis,* Sixth International Conference on Computational Intelligence and Information Technology