**Artificial Intelligence (AI) Solution Implementation Using Machine Learning (ML) Algorithms**

Author: Lakshmi Narayana Bharadwaj Ganapavarapu

Email: lg22110@essex.ac.uk

Ph: +918019761936

**Disclaimer**

This paper has screenshots of the Machine Learning (ML) models, that contain valuable information on how the model has performed. Readers are encouraged to zoom in on the document for better visibility before commencing the evaluation of the document.

**Abstract**

It is evident by now that the term Artificial Intelligence or shortly referred to as AI is an age old yet effective and a powerful concept, that has capabilities of accomplishing the most complex of tasks; so are the problems of the current world. AI has not only found its place in almost all major sectors of the world but also has provided them with solutions whose performance is incomprehensible to the human brain. Simply put, life these days has become simpler with the advent of AI. Finance is one such sector which extensively utilizes a plethora of concepts associated with AI to get things done.

This paper is obligated to focus on the possible applications of AI and Machine Learning (ML) algorithms on an experimental level that includes various classification, regression, and clustering techniques, so that financial organizations could apprehend, benefit, and integrate these crackers of concepts into their work pipelines to achieve their targets, sub-targets, generate valuable insights from their data, and beyond and make a profound impact in their domain on their competition. Before we begin experimenting on the data, with the ML concepts, we will briefly look at how businesses need to operate. We will then, look at the conceptual side of the ML algorithms, and the same will later be applied onto a couple of experimental datasets to observe how they perform. We shall constrain the paper to specific ML algorithms that prove to be useful in most sectors of the world and can also be effectively used in the finance domain.

**Brief Understanding on How Businesses Should Operate**

Whenever a business or an organization wants to transition their decision making by depending on data or build from ground up, the key component that is required is the data. The Cross Industry Standard Process for Data Management (CRISP-DM) method takes care of almost all the aspects of a business. It comprises of six phases namely:

* Business Understanding: Establishing necessary business requirements
* Data Understanding: Getting to know the available data and it’s availability.
* Data Preparation: How to prepare data for modelling?
* Modelling: Feasible techniques that can we apply to the data
* Evaluation: How are the models performing and what is the best possible model as per the business understanding?
* Deployment: How can end users benefit from the product or the service offered?

One needs to ensure that all the aforementioned steps are executed by keeping various parameters like company policies, ethical issues, security, and so on, in mind. Below is a harmonious representation of the CRISP-DM method (Danso, 2022):



Figure 1: CRISP-DM Method (Danso, 2022).

Owing to the CRISP-DM method, finance companies can also take up this approach for transitioning into better service providing and rolling out capable products, thereby, relying on the powerful ML algorithms and capitalizing on their stance in the competition. Let’s look at how ML algorithms function.

**Briefing on Regression, Classification and Clustering**

In the world of AI and ML, there are plenty of algorithms that an individual can use and solve complex tasks. As entitled, regression, classification, and clustering are a few of many widely used ML algorithms to tackle data related problems. Let’s briefly observe what each algorithm does:

1. **Regression:** According to Corporate Finance Institute (CFI) (N.D.), regression is a method that uses statistical methods to develop a relation between what is called a dependent variable and one or many independent variables. It can be used to determine the strength of the relationship between the said variables. It has various forms like simple, multiple, and non-linear. A simple regression model tries to establish a relationship between a dependent variable and an independent variable using the below mathematical equation:

Where,

Y = Dependent Variable

X= Independent Variable

m= Slope of the line

c= Intercept

e= residual error.

When dealing with multiple linear regression, there are multiple independent variables present in the equation, thereby leading to many possible slope coefficients. It looks like this:

Where,

Y = Dependent Variable

X1, X2, …= Independent Variables

m1, m2, …= Slopes of the lines

c= Intercept

e= residual error.

In multiple linear regression, since there are many independent variables, there is an additional condition that is to be taken care of, which is, the collinearity between the independent variables. Higher the collinearity, difficult is the process to form a relation between the variables. Rest assured, it is a similar process to that of simple linear regression. You can find more details about regression and how finance sector can benefit from it [here.](https://corporatefinanceinstitute.com/resources/knowledge/finance/regression-analysis/)

[This YouTube channel](https://www.youtube.com/c/KhanAcademyProbabilityandStatistics/playlists) is another great resource for simple ML algorithms.

1. **Classification:** JavaTPoint (N.D) define classification as a supervised learning technique that is used to identify or categorize new instances of data into their respective “category” or “class” from a given dataset. This technique is used to predict the output of new data after learning the characteristics of the input data. There are quite a few classification algorithms due to which depicting a general equation for this technique is hard.

Wolff (2020) has given various types of classification techniques and how they function. Let’s briefly look at them below:

* **Logistic Regression:** This is used to obtain a logical outcome for a given problem which is generally binary in nature like 0 or 1, yes or no, etc.
* **Naïve Bayes:** This technique determines whether a data point belongs to a certain category or not. It operates in the lines of the below formula:

Where,

P(A|B) = Probability of event A occurring, given B has occurred

P(B|A) = Probability of event B occurring, given A has occurred

P(A)= Probability of event A and P(B)= Probability of event B.

This can also be termed as conditional probability Bayes’ Theorem.

* **Decision Tree:** As entitled, this supervised learning algorithm is capable of classifying outcomes on a precise level. It looks like a flow chart, where in, as the tree develops, classifying the outcome becomes much more organic and precise. Below is a depiction of a decision tree:

Diagram

Description automatically generatedFigure 2: Depiction of a Decision Tree (Wolff, 2020)

* **Random Forest:** This algorithm builds numerous decision trees in using the input data and then tries to fit or classify the new data within those multitude of decision trees. It is often used as an alternate if decision tree fails to offer satisfactory results.

Read the full articles on classification techniques by [JavaTPoint](https://www.javatpoint.com/classification-algorithm-in-machine-learning) & [Wolff](https://monkeylearn.com/blog/classification-algorithms/). Also, go through [this playlist](https://www.youtube.com/playlist?list=PLEiEAq2VkUULNa6MHQAZSOBxzB6HHFXj4) for thorough info on classification algorithms. These are a few of many techniques that financial organizations utilize to identify fraudulent customers, determine customers eligible for loans and so on.

1. **Clustering:** Priy (2022) defined clustering as dividing the whole data (sometimes referred to as population) into several groups, where in, each group of data points portray specific behaviour. To put it in simple terms, grouping of data points based on their similarities and dissimilarities when analysed is called clustering. Clusters can be formed in multiple shapes and sizes. It is not necessary that a cluster should assume a specific shape. After all, it is the characteristics of the data points, due to which, clusters are formed. A pictorial representation of the same is given below:

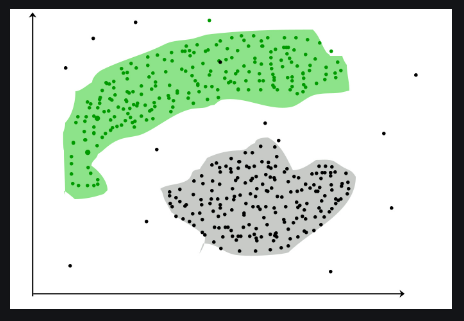
****

Figure 3: Pictorial representation of Clustering (Priy, 2022).

There are several clustering algorithms that perform in their own ways. McGregor (2020), has given a few clustering types and algorithms:

* **Density Based:** In this method, the data is grouped or clustered by the density of data points. The algorithm tries to find areas with high concentration of data points and groups them into clusters.
* **Distribution Based:** In this method, data is grouped as a cluster depending on each instance’s probability of being in a cluster. A centre point is assumed; the probability of a data point being in the cluster depends on the distance of the data point from the assumed centre point. Higher the distance, lesser is the probability of that data point being grouped into the cluster.
* **Centroid Based:** The most common, efficient, buta sensitive technique for clustering. Multiple centroids are assumed in the data; each data point is then clustered based on the squared distance of it from the centroid.
* **Hierarchical Based:** As entitled, it is used to data that is available in a hierarchical format, usually, taxonomy databases. A tree of clusters is built and organized from the top-down.

Apart from this, the commonly heard clustering algorithms are enlisted below:

* **K-Means Clustering:** This is the most commonly used centroid based clustering algorithm and the simplest unsupervised learning method. This is a handy algorithm for small datasets.
* **DBSCAN Clustering:** unlike K-Means, this is a density-based algorithm that helps in identifying unnecessary data points that hinder a model’s performance.
* **BIRCH clustering:**  It stands for Balance Iterative Reducing and Clustering using Hierarchies. Also, used on large datasets unlike K-Means. This method breaks large data into smaller chunks, which, is then clustered.

Please refer to <Priy's> article and [McGregor's](https://www.freecodecamp.org/news/8-clustering-algorithms-in-machine-learning-that-all-data-scientists-should-know/) articles on clustering algorithms. Also, here is [Marc Whalley's](https://www.youtube.com/playlist?list=PL2l879VLd5ulj1Kqd7DXcuqm67qqW2mqm) YouTube playlist for clustering algorithms.

Financial companies extensively use clustering algorithms in their work to identify customers with good credit score, salary distribution between tiers of their employees and so on.

These are a few of many more capabilities of AI and ML concepts that companies can benefit form and contribute extensively to their performance. Further from now, we shall continue onto the experimentation part where in we test out a few of the discussed ML algorithms and observe how they do. We will also discuss how the application of these powerful algorithms can be transitioned into the financial domain and how employees in the finance domain can tackle multiple problems with these algorithms and bring value and efficiency to their performance.

**Experimentation With ML Algorithms**

Here, we will observe, experiment, model and compare various ML algorithms and their performances on data sourced from the developer website, GitHub. Two separate datasets are considered to help differentiate between the regression and classification algorithms. Below are the details of the experiments conducted:

1. **Regression Experiment:**

Name of the dataset: delaney\_solubility\_with\_descriptors.csv

Link to the Dataset: <https://github.com/dataprofessor/data>

Description: This dataset contains 4 independent variables (MolLogP, MolWt, NumRotatableBonds, and AromaticProportion) and a dependent variable (logS). All of them are numeric in nature. The aim is to predict the logS value using the other 4 variables. Since regression works best with numeric data, the above dataset was considered.

The data pre-processing steps involved normalization of the independent variables to make the algorithm comprehend data more simply (WEKA automatically leaves the dependent variable out while performing this step). Then, the following algorithms were run and here are their respective outputs:

Graphical user interface, application

Description automatically generated

Figure 4: WEKA Output for Simple Linear Regression.

Graphical user interface, application

Description automatically generated

Figure 5: WEKA Output for Cross Validation Linear Regression.

Graphical user interface, text, application

Description automatically generated

Figure 6: WEKA Output for 80-20 Train-Test Split Linear Regression.

The final equations can be seen under the Linear Regression Model line in figures 4, 5, and 6 respectively, and there are various metrics on which the model’s performance can be judged. The GUI also gives you the time taken to run each model, which, is evidently quick. Based on the screenshots, the simple linear regression model gave the best result out of all three models.

1. **Classification Experiment:**

Name of Dataset: penguins\_cleaned.csv

Link to the dataset: <https://github.com/dataprofessor/data>

Description: There are 6 independent variables (mixture of numeric and nominal variables) and one dependent or class variable (nominal variable) that is to be predicted. Unlike other datasets where there is an imbalance in the class variable, this has almost equal number of instances of both classes (male and female). The aim is to classify the outcome as a male or a female. It also has a few numeric variables, which have a wide range of values. Hence, as a pre-processing step, normalization was done for the numeric variables. We have then moved on to modelling and here are outputs from various models:

Graphical user interface, application

Description automatically generated

Figure 7.1: WEKA output for a simple Decision Tree.

Diagram, timeline

Description automatically generated

Figure 7.2: Tree Diagram for the Simple Decision Tree.

Graphical user interface, application

Description automatically generated

Figure 8.1: WEKA Output for Cross Validation Decision Tree.

Diagram

Description automatically generated

Figure 8.2: Tree Diagram for Cross Validation Decision Tree.

Graphical user interface

Description automatically generated

Figure 9.1: WEKA Output for 80-20 Train-Test Split Decision Tree.

Diagram

Description automatically generated

Figure 9.2: Tree Diagram for 80-20 Train-Test Split Decision Tree.

The following are the outputs for Random Forest Model:

Graphical user interface, text, application

Description automatically generated

Figure 10: WEKA Output for Random Forest.

Graphical user interface, text, application

Description automatically generated

Figure 11: WEKA Output for Cross Validation Random Forest.

Graphical user interface, text

Description automatically generated

Figure 12: WEKA Output for 80-20 Train-Test Split Random Forest.

In all the classification experiment screenshots, the summary section provides us the information on how well the model is performing. Based on the above screenshots, the random forest with built with the 80-20 split gives the best performance.

In this manner, AI and ML algorithms can come in handy to help accomplish data related tasks.

**Conclusion**

This is just the tip of the iceberg when it comes to model building and analysis using data. This can be taken much further if companies dig deep into bringing out insights. As discussed, the purpose of this paper was to make the finance organizations understand the capacity in which AI and ML algorithms perform and persuade them to transition into this approach of tackling problems and the above experimental analysis should encourage finance companies to make the switch.

**References**

Danso, S. (2022) *Understanding Artificial Intelligence Week 8 Seminar*. UAI\_PCOM7E March 2022. University of Essex Online.

Institute, C.F. (N.D) Regression Analysis. Available from: https://corporatefinanceinstitute.com/resources/knowledge/finance/regression-analysis/ [Accessed 27 May 2022]

Khan Academy Probability and Statistics (2018) Regression| Probability and Statistics| Khan Academy. Available at: https://www.youtube.com/c/KhanAcademyProbabilityandStatistics/playlists [Accessed 27 May 2022]

JavaTPoint. (N.D) Classification Algorithm in Machine Learning. Available from: https://www.javatpoint.com/classification-algorithm-in-machine-learning [Accessed 27 May 2022]

Wolff, R. (2020) 5 Types of Classification Algorithms in Machine Learning. 27 August 2020. *MonkeyLearn Blog*. Available from: https://monkeylearn.com/blog/classification-algorithms/ [Accessed 27 May 2022]

Simplilearn (2022) Machine Learning Algorithms [2022 updated] Available from: https://www.youtube.com/playlist?list=PLEiEAq2VkUULNa6MHQAZSOBxzB6HHFXj4 [Accessed 27 May 2022]

Priy, S. (2022) Clustering in Machine Learning. 18 May 2022. *GeeksForGeeks Blog*. Available from: https://www.geeksforgeeks.org/clustering-in-machine-learning/ [Accessed 27 May 2022]

McGregor, M. (2020) 8 Clustering Algorithms in Machine Learning that All Data Scientists Should Know. 21 September 2020. *freeCodeCamp Blogs.* Available from: https://www.freecodecamp.org/news/8-clustering-algorithms-in-machine-learning-that-all-data-scientists-should-know/ [Accessed 27 May 2022]

Whalley, M. (2019) clustering. Available from: https://www.youtube.com/playlist?list=PL2l879VLd5ulj1Kqd7DXcuqm67qqW2mqm [Accessed 27 May 2022]

**Appendices**

Appendix 1: CRISP-DM Method (Danso, 2022). [Image]

Appendix 2: Depiction of a Decision Tree (Wolff, 2020). [Image]

Appendix 3: Pictorial Representation of Clustering (Priy, 2022). [Image]

Appendix 4: WEKA Output for Simple Linear Regression. [Image]

Appendix 5: WEKA Output for Cross Validation Linear Regression. [Image]

Appendix 6: WEKA Output for 80-20 Train-Test Split Linear Regression. [Image]

Appendix 7.1: WEKA Output for Simple Decision Tree. [Image]

Appendix 7.2: Tree Diagram for Simple Decision Tree [Image]

Appendix 8.1: WEKA Output for Cross Validation Decision Tree. [Image]

Appendix 8.2: Tree Diagram for Cross Validation Decision Tree [Image]

Appendix 9.1: WEKA Output for 80-20 Train-Test Split Decision Tree. [Image]

Appendix 9.2: Tree Diagram for 80-20 Train-Test Split Decision Tree [Image]

Appendix 10: WEKA Output for Simple Random Forest. [Image]

Appendix 11: WEKA Output for Cross Validation Random Forest. [Image]

Appendix 12: WEKA Output for 80-20 Train-Test Split Decision Tree. [Image]