Abstract

Machine learning (ML) **algorithms** are fast changing the way of how we look and interpret our data to draw meaningful conclusions. Given the increase in data driven decisions**, business optimization** can be achieved by unearthing ever-more-complex information, through **analytics**. These are very helpful for companies /startups to have a better insight into operations and potential threats for disruption. In this project I have tried to analyze some of the very popular regression algorithms to predict the profit value when some of the important predictors such as R&D Spend, Administration, Marketing Spend and also the Profit earned by 50 companies were given in the dataset.

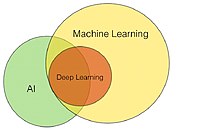
Table of Contents

|  |  |  |
| --- | --- | --- |
| *S.no* | *Topics* | *Page No* |
| 1. | Introduction | 1-2 |
| 2. | Existing Method | 3-12 |
| 3. | Proposed Method with Architecture | 13-24 |
| 4. | Methodology | 25 |
| 5. | Implementation | 26-27 |
| 6. | Conclusion | 28-29 |

Profit value Prediction using Machine Learning

Introduction

For the past 50 years or so we have been able to automate tasks using a computer or more specifically by writing code to achieve what we wanted. But in the recent times there have arose multiple problems where just writing a piece of code won’t help, for example in sales forecasting or in language translation, these are few of the many examples where we need the use of **Machine Learning (ML).** Let’s begin by defining ML, according to Wikipedia,” Machine **learning** (**ML**) is a field of inquiry devoted to understanding and building methods that **‘learn’** that is methods that leverage data to improve performance on some set of tasks. It is seen as a part of Artificial intelligence”.



**Fig: Classification**

Machine learning algorithms/methods build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

**Artificial Intelligence**

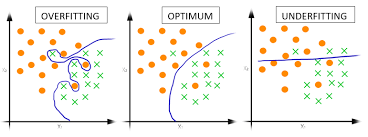
ML is considered as a subset of Artificial intelligence, which can be defined as the science and engineering of creating intelligent machines, AI as an academic discipline in its early years is credited for laying the foundation of the modern day Machine learning. Researchers in those days were interested in having machines learn from data, so they employed various techniques which included: Probabilistic reasoning, neural networks and various other symbolic methods.

But very soon the Probabilistic systems employed were plagued by theoretical and practical problems of data acquisition and representation. This led to the creation of the field of ML, where the goal changed from achieving artificial intelligence to solving problems of more practical nature. This as a result incorporated methods and models borrowed from statistics, fuzzy logic, and probability theory.

**Machine Learning-Objectives**

A core objective of a learner/model is to generalize from its experience. Generalization in this context is the ability of a learning machine to perform accurately on new, unseen examples/tasks after having experienced a learning data set. The training examples come from some generally unknown probability distribution (considered representative of the space of occurrences) and the learner has to build a general model about this space that enables it to produce sufficiently accurate predictions in new cases.

For the best performance in the context of generalization, the complexity of the hypothesis should match the complexity of the function underlying the data. If the hypothesis is less complex than the function, then the model has **under fitted** the data. If the complexity of the model is increased in response, then the training error decreases. But if the hypothesis is too complex, then the model is subject to **over fitting** and generalization will be poorer.

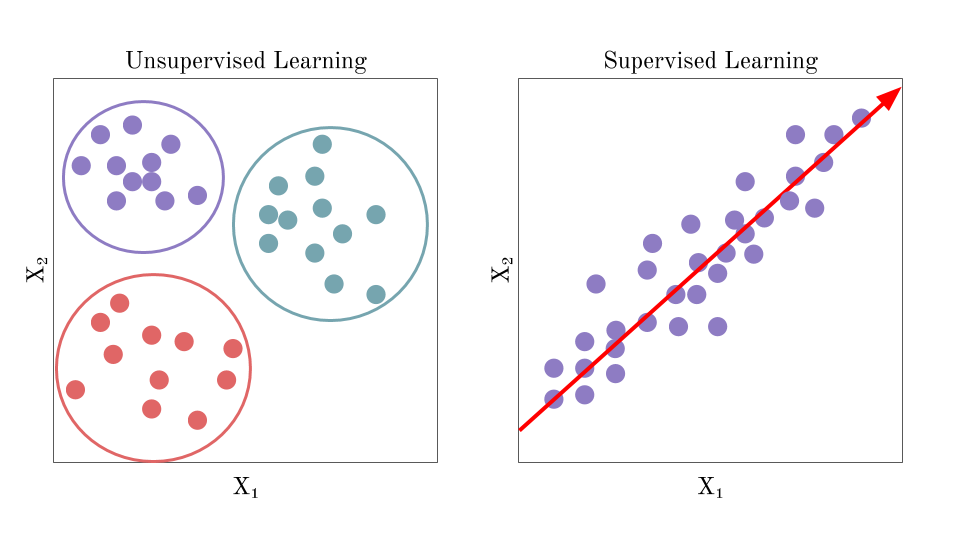
****

**Fig: Over fitting-Optimum- Under fitting**

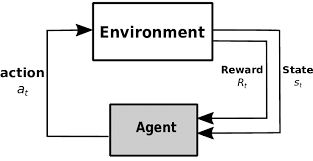
Existing Method

Machine learning approaches are traditionally divided into three broad categories, which correspond to learning paradigms, depending on the nature of the "signal" or "feedback" available to the learning system:

* **Supervised Learning**: The model is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
* **Unsupervised Learning:** No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning)
* **Reinforcement Learning:** A computer program interacts with a dynamic environment in which it must perform a certain goal. As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.



**Fig: Supervised Learning VS Unsupervised Learning**



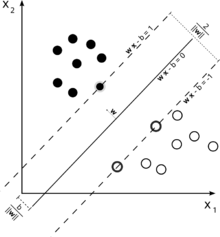
**Fig: Reinforcement Learning**

Supervised Learning

* Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs.
* The data is known as training data and consists of a set of training examples.
* Each training example has one or more inputs and the desired output, also known as a supervisory signal.
* Each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix.
* Through mathematical programming of a loss function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs.
* An algorithm that improves the accuracy of its outputs or predictions over time is said to have **learned** to perform that task.

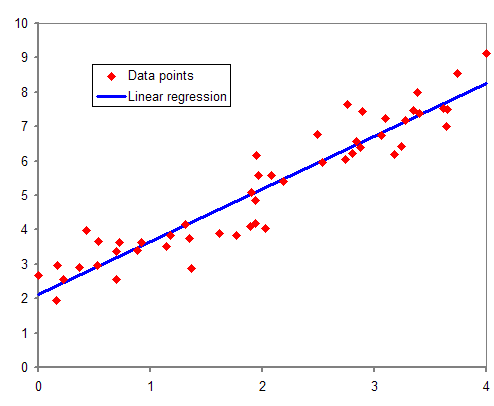
Types of Supervised Learning

* **Classification Algorithm**: An algorithm that implements classification, especially in a concrete implementation, is known as a **classifier.** The term "classifier” also refers to the mathematical function, implemented by a classification algorithm that maps input data to a category. For example the problem of identifying which of a set of categories an observation belongs to such as **spam** or **not spam.** Note that Classification algorithms are used when the outputs are restricted to a limited set of values.



**Fig: Support Vector Machine** **is a supervised learning model that divides the data into regions separated by a linear boundary. Here, the linear boundary divides the black circles from the white.**

* **Regression Algorithm:** The underlying principle of regression algorithm is regression analysis which is a statistical process or estimating the relationships between a dependent variable (often called the 'outcome' or a 'label' in machine learning) and one or more independent variables (often called 'predictors', or 'features'). The most common form of regression analysis is linear regression, in which one finds the line (or a more complex linear combination) that most closely fits the data according to a specific mathematical criterion. Also note that, regression algorithms are used when the outputs may have any numerical value within a range.



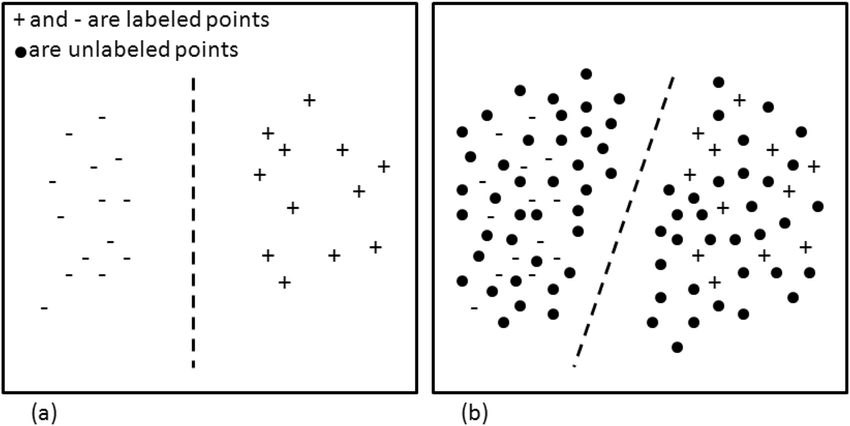
**Fig: Linear Regression (shown in blue), fitted across data points (shown in red)**

Unsupervised Learning

* Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points
* The algorithms, therefore, learn from test data that has not been labelled, classified or categorized.
* Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data
* A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function (PDE)

Semi Supervised Learning

* Semi-supervised learning falls between unsupervised learning (without any labelled training data) and supervised learning (with completely labelled training data).
* Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labelled data, can produce a considerable improvement in learning accuracy
* In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets.



# Fig: To increase Generalization in Semi supervised algorithm decision boundary placed a) in the presence of unlabelled points b) in the presence of labelled points

Reinforcement Learning

* Reinforcement learning is an area of machine learning concerned with how software **agents** ought to take actions in an **environment** so as to maximize some notion of cumulative **reward**.
* Many reinforcement learning algorithms use dynamic programming techniques
* Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent.
* Reinforcement learning algorithms are used when exact models are infeasible.

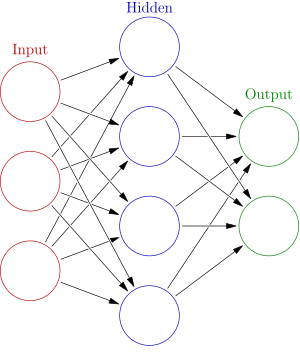
Models

Performing machine learning involves creating a model, which is trained on some training data and then can process additional data to make predictions. Various types of models have been used and researched for machine learning systems.

* **Artificial Neural Networks:** Artificial neural networks (ANNs), are computing systems vaguely inspired by the biological neural networksthat constitute animal brains**.** Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules.

An ANN is a model based on a collection of connected units or nodes called "artificial neurons", which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit information, a "signal", from one artificial neuron to another. An artificial neuron that receives a signal can process it and then signal additional artificial neurons connected to it. The connections between artificial neurons are called "edges". Artificial neurons and edges typically have a weight that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection. Typically, artificial neurons are aggregated into layers. Different layers may perform different kinds of transformations on their inputs. Signals travel from the first layer (the input layer) to the last layer (the output layer), possibly after traversing the layers multiple times.

Artificial neural networks have been used on a variety of tasks, including computer vision, speech recognition, and medical diagnosis.

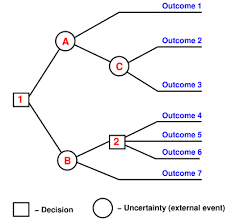


**Fig: Neural network with neurons and layers**

* **Decision Trees:** Decision tree learning uses a decision tree[ a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes] as a predictive model to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves).

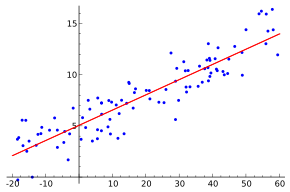
**Types of Decision Trees:**

* **Classification Tree:** analysis is when the predicted outcome is the class (discrete) to which the data belongs.
* **Regression tree** analysis is when the predicted outcome can be considered a real number (e.g. the price of a house, or a patient's length of stay in a hospital).

****

**Fig: Decision Tree**

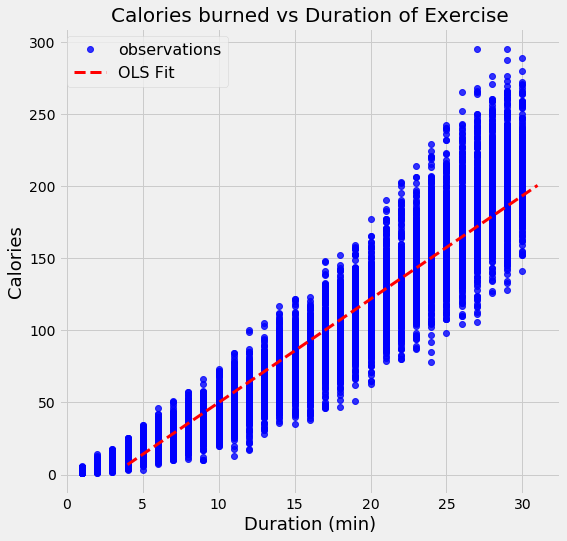
* **Support Vector Machine:** Support-vector machines (SVMs), also known as support-vector networks, are a set of related supervised learning methods used for classification and regression. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that predicts whether a new example falls into one category SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.
* **Regression Analysis:** Regression analysis encompasses a large variety of statistical methods to estimate the relationship between input variables and their associated features. Its most common form is linear regression, where a single line is drawn to best fit the given data according to a mathematical criterion such as ordinary least squares. The latter is often extended by regularization methods to mitigate over fitting and bias, as in ridge regression. When dealing with non-linear problems, go-to models include polynomial regression, logistic regression etc.



**Fig: Linear Regression**

* **Bayesian Regression:** A Bayesian network, belief network, or directed acyclic graphical model is a probabilistic graphical model that represents a set of random variables and their conditional independence with a directed acyclic graph (DAG)

For example, a Bayesian network could represent the probabilistic relationships between diseases and symptoms. Given symptoms, the network can be used to compute the probabilities of the presence of various diseases.

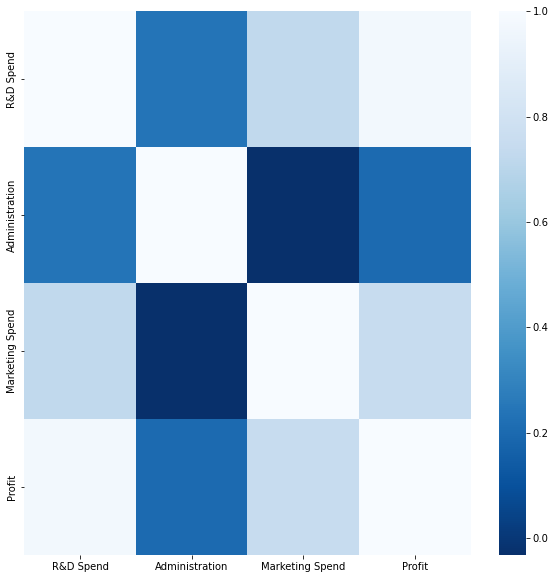


**Fig: An example showing Bayesian Ridge**

Proposed Method with Architecture

The task at hand is to predict the profit value from a data set containing the R&D spends, Administration and Marketing spend and Profit earned. The proposed method after skimming through the existing was of **Regression algorithm**, due to its known top notch performance in prediction based problems.

Owing to the pretty much fair correlation between the predictor variables and the target and also because of the existing linearity between them, the first approach was to go for **linear regression algorithm**, which is the part of supervised learning.



**Fig: Correlation Heatmap: R&D spend and profit have the strongest correlation 0.97 followed by R&D spend and Marketing 0.74**

**Linear Regression- A brief idea**

Linear regression is a quiet and simple statistical regression method used for predictive analysis and shows the relationship between the continuous variables. Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), consequently called linear regression*.*If there is a single input variable (x), such linear regression is called **simple linear regression**. And if there is more than one input variable, such linear regression is called **multiple linear regressions.**  For this particular problem we have employed multiple linear regression technique, as the number of predictors/inputs is more than one.

**Mathematical approach**

For three independent variables, 𝑥₁, 𝑥₂, 𝑥3 here R&D spend, Administration and Marketing spend, the regression function can be defined as (𝑥₁, 𝑥₂, 𝑥3) = 𝑏₀ + 𝑏₁𝑥₁ + 𝑏₂𝑥₂+ 𝑏3𝑥3 . The goal of regression is to determine the values of the weights 𝑏₀, 𝑏₁, 𝑏₂, 𝑏3 and such that regression plane is as close as possible to the actual responses. A more generalised expression for linear regression can be given as: (𝑥₁, …, 𝑥ᵣ) = 𝑏₀ + 𝑏₁𝑥₁ + ⋯ +𝑏ᵣ𝑥ᵣ, where r+1 weights are determined when number of inputs is r.

**Algorithm Performance:**

The best way to gauge an algorithm’s performance and how well it fits and applies to the underlying data is measured by the regression metrics. Since regression predictive modelling involves in predicting a numeric value, therefore the accuracy of a regression model cannot be predicted, but only how close the predicted value was to the expected values

* Mean Absolute Error: 6979.122252370401
* Mean Absolute percentage error: 0.1025215664429128
* Mean squared error: 80926443.7142967
* Root mean squared error: 8995.973749405897
* R2 score: 0.9000639404227893
* Adjusted R-squared: 0.8935463713199276



**Fig: Algorithm performance- Actual VS Predicted value**

**Ridge Regression- A brief idea**

Ridge regression is one of the types of linear regression in which a small amount of bias is introduced so that we can get better long-term predictions. Ridge regression is a regularization technique, which is used to reduce the complexity of the model. It is also called as **L2 regularization**.

In this technique, the cost function is altered by adding the penalty term to it. The amount of bias added to the model is called **Ridge Regression penalty**. We can calculate it by multiplying with the lambda to the squared weight of each individual feature.

**Mathematical Approach:**

Ridge regression is a type of regularization technique, where all variables or features in the model are maintained by reducing the magnitude of the variables. Hence, it maintains accuracy as well as a generalization of the model, reducing over fit.

Regularization works by adding a penalty or complexity term to the complex model. Let's consider the simple linear regression equation:

y= β0+β1x1+β2x2+β3x3+⋯+βnxn +b

Where, X1, X2 … Xn are the features for Y.

β0, β1 .…... βn are the weights or magnitude attached to the features,

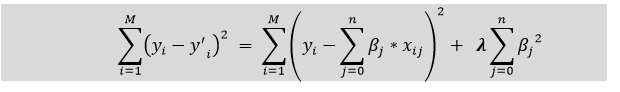
And b represents the intercept.

Linear regression models try to optimize the β0 and b to minimize the cost function. The equation for the cost function for the linear model is given below:

regularization-in-machine-learning.png

In ridge regression the above equation modifies when penalty term **(λ) is added to** regularize the coefficients of the model, and hence ridge regression reduces the amplitudes of the coefficients that decreases the complexity of the model.

The equation becomes:



**Algorithm Performance**

At alpha value 0.001 (penalty term)

* Mean Absolute Error: 6980.591677368231
* Mean Absolute percentage error: 0.10252428 149325576
* Mean squared error: 80948059.90864487
* Root mean squared error: 8996.97330772966
* R2 score: 0.9000417310494496



**Fig: Algorithm performance- Actual VS Predicted value**

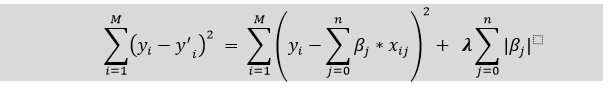
**Lasso Regression- A brief idea**

Lasso regression is another regularization technique to reduce the complexity of the model. It stands for **Least Absolute and Selection Operator.**

It is similar to the Ridge Regression except that the penalty term contains only the absolute weights instead of a square of weights.

Since it takes absolute values, hence, it can shrink the slope to 0, whereas Ridge Regression can only shrink it near to 0.

It is also called as **L1 regularization.** The equation for the cost function of Lasso regression will be:



Hence, the Lasso regression can help us to reduce the over fitting in the model as well as the feature selection.

**Algorithm Performance**

At alpha value 0.001 (penalty term)

* Mean Absolute Error: 6979.121602165693
* Mean Absolute percentage error: 0.10252155690854887
* Mean squared error: 80926426.00783528
* Root mean squared error: 8995.973749405897
* R2 score: 0.9000639404227893



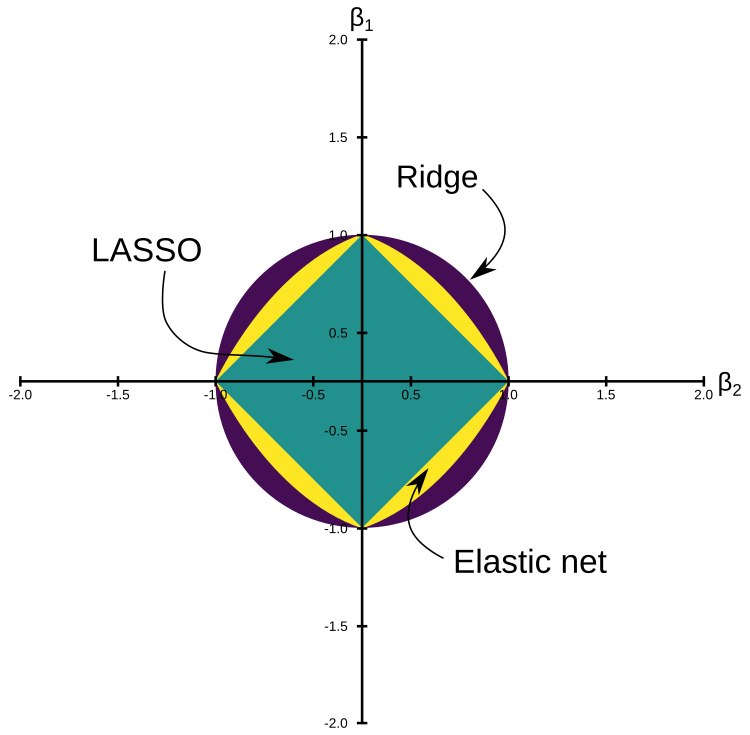
**Fig: Algorithm performance- Actual VS Predicted value**

**Elastic net Regression- A brief idea**

Elastic Net is a regularized regression model that combines l1 and l2 penalties, i.e., lasso and ridge regression.

The elastic net includes the penalty of lasso regression, and when used in isolation, it becomes the ridge regression. In the procedure of regularization with elastic net, first, we find the coefficient of ridge regression. After this, we perform a lasso algorithm on the ridge regression coefficient to shrink the coefficient.

This will be easier to understand by the following diagram

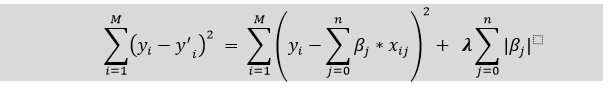


**Fig: Here we can see that after performing the ridge regression, the lasso regression takes part in the procedure that considers all the variables from the dataset.**

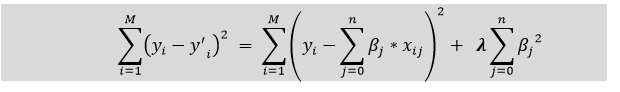
**Mathematical Approach**

Elastic net regressor = Loss function of Ridge regression+ Loss function of Lasso regression:

For Lasso Regression:



For Ridge Regression:



For Elastic net regression adding the above two equations:

image-144.png

**Algorithm Performance**

* Mean Absolute Error: 6979.122245675227
* Mean Absolute percentage error: 0.10252156633356062
* Mean squared error: 80926443.50635484
* Root mean squared error: 8995.973749405897
* R2 score: 0.9000639404227893



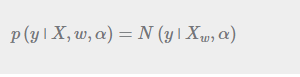
**Fig: Algorithm performance- Actual VS Predicted value**

**Bayesian Ridge Regression- A brief idea**

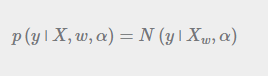
Bayesian regression allows a natural mechanism to survive insufficient data or poorly distributed data by formulating linear regression using probability distributors rather than point estimates. The output or response ‘y’ is assumed to drawn from a probability distribution rather than estimated as a single value.

**Mathematical Approach**

Mathematically, to obtain a fully probabilistic model the response y is assumed to be Gaussian distributed around Xw as follows:



One of the most useful types of Bayesian regression is Bayesian Ridge regression which estimates a probabilistic model of the regression problem. Here the prior for the coefficient w is given by spherical Gaussian as follows –



**Algorithm Performance**

* Mean Absolute Error: 7028.0374662903605
* Mean Absolute percentage error: 0.10293393500991255
* Mean squared error: 81314993.21626456
* Root mean squared error: 9017.482642969964
* R2 score: 0.8995854855461145



**Fig: Algorithm performance- Actual VS Predicted value**

Methodology

A machine learning model creation and deployment follows a strict methodology which consists of many steps each of whose validation is important in the creation of a successful machine learning project. Some of these are:

1. Observing the data set and finding how data is distributed, performing exploratory data analysis

2. Choosing a good algorithm which can actually find the underlying pattern in data which is useful for predictive analysis.

3. Train the algorithm on the data set and observe the actual and the predicted values obtained, watch out for over and under fit, during model fitting

4. If conditions such as over fit and under fit are observed use regularization techniques to improve the model.

5. Again watch the predicted values, how close or how far they are from expected value use regression metrics for this purpose for (regression problems)

6. Tune all the hyper parameters and then implement the model for use

7. Model deployment

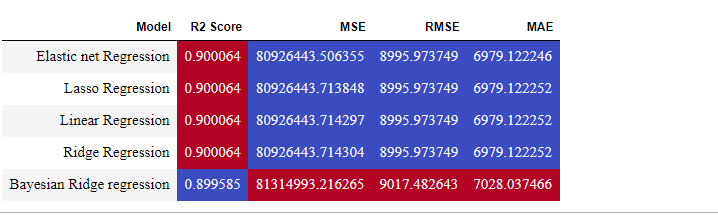
Implementation

For the practical implementation of the machine learning algorithms, I followed these steps:

1. Import the necessary libraries
2. Import the data set using pandas and create a data frame to hold the data
3. Perform exploratory data analysis to find the relationships in the data
4. Check for any null values, supplement them if any
5. Perform feature scaling: Normalization and Standardization
6. Split the data set columns into predictor and target variables
7. Import train test split from scikitlearn library and split the data into train and test data; clearly specifying the amount for test/ train data.
8. Import the algorithm from its respective package
9. Implement/ Run the model to check the predicted values
10. Used regression metrics to check how far the predicted value is from the expected value
11. Checking for over fit and under fit, used regularization techniques to minimise the same.
12. After successfully training and testing the model, implement and then deployed the same.

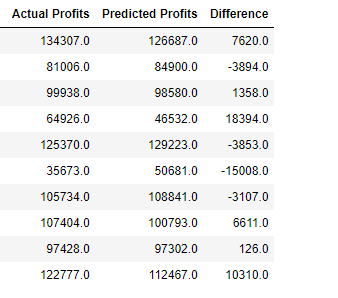
**Conclusion**

Let us again look at our regression metric to draw the final conclusion of the best performing algorithm



Of all the algorithms **Elastic net regression** has the least mean absolute error, which means better prediction since it is the magnitude of difference between the prediction of an observation and the true value of that observation.

**Note**: R2 score is not considered for evaluation since R² can be calculated before even fitting a regression model, which doesn't make sense then to use it for judging prediction ability



**Fig: Actual and Predicted Profit values**