

## EXPERIMENT 6

### Aim :

Develop a Machine Learning Method to Predict Stock Prices based on past Price Variations.

### Introduction :

Stock market prediction is the act of trying to determine the future value of a company stock or other financial instrument traded on an exchange. The successful prediction of a stock's future price could yield significant profit. The efficient-market hypothesis suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information thus are inherently unpredictable. Others disagree and those with this viewpoint possess myriad methods and technologies which purportedly allow them to gain future price information.

Predicting how the stock market will perform is one of the most difficult things to do. There are so many factors involved in the prediction - physical factors vs. psychological, rational and irrational behavior, etc. All these aspects combine to make share prices volatile and very difficult to predict with a high degree of accuracy.

Using features like the latest announcements about an organization, their quarterly revenue results, etc., machine learning techniques have the potential to unearth patterns and insights we didn't see before, and these can be used to make unerringly accurate predictions.

**Regression** is a suitable algorithm for stock market prediction as we are dealing with prediction of continuous values. Linear Regression is simplest form of regression analysis. Regression deals with identifying suitable input parameters and output parameter to get a linear equation of form:

$$Y = b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n$$

The aim is to find the best value of the coefficients  $b_0, b_1, \dots, b_n$  so as to minimize the error between predicted output and actual output.

**Implementing Regression Algorithm :**

```
int k = 0;
for (int j = 0; j < 100; j++)
{
    // number of epochs
    k = 0;
    while (k < 10)
    {
        // number of training instances
        for (int i = 0; i < 3; i++)
        {
            // number of coefficients
            b[i] = b[i] + alpha * (y[k] - prediction) * prediction *
            (1 - prediction)* x[k][i];
            cout << b[i]<<"\n";
        }
        k++; output = 0;
        for (int i = 0; i < 3; i++)
        {
            output += b[i] * x[k][i];
        }
    }
    // end of while loop
}
// end of for loop
```