

**Project Report**

on

**BITCOIN PRICE PREDICTOR**

Under the guidance of

Aishwarya Saxena (Mentor)

By – Pankaj Kumar

Machine Learning – Python

College – Birsa Institute of Technology, Sindri (B.I.T. Sindri)

Winter Internship: January – February (2021)

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S NO.** | **TOPIC** | **PAGE NO.** |
| 1. | INTRODUCTION | 3 |
| 2. | OBJECTIVE | 4 |
| 3. | BACKGROUND | 5 |
| 4. | HARDWARE AND SOFTWARE REQUIREMENTS | 6 |
| 5. | FUTURE SCOPE | 7 |
| 6. | CONCLUSION | 8 |
| 7. | BIBLIOGRAPHY AND REFERENCES | 9 |

**INTRODUCTION**

Bitcoin is a global cryptocurrency and online payment system that is highly stable and secure. It is Peer-to-peer value transfer and transaction protocol. The Bitcoin’s price varies similarly to a stock albeit in another way. There are some algorithms used on stock market data for price prediction but the parameters affecting Bitcoin are distinctive. Therefore, it is essential to expect the price of Bitcoin in order that correct investment decisions can be made. The price of Bitcoin does now not rely on at the business events or intervening government in contrast to the stock market. Hence, to expect the value we feel it is essential to leverage machine learning technology to expect the rate of Bitcoin.



**OBJECTIVE**

The prime objectives of this project are:

* To predict bitcoin price with maximum efficiency.
* To ensure less risk and more profit for investors.
* To compare between machine learning models to find which one is the best efficient algorithm for predicting bitcoin price.
* Helpful for the customers to decide their investment plans in bitcoins.
* To make correct investment decisions.

**BACKGROUND**

In this project, we have applied some of the models to leverage machine learning technology to predict the real-time price of Bitcoin. However, machine learning literature is lacking verification of whether or not the stock evaluation strategies are legitimate for the cryptocurrencies, and if so, how they may be modified. That is what features want to be eliminated or introduced as a foundation for price prediction, whether current machine learning algorithms work for cryptocurrencies, and which technique yields the excellent outcomes.

In this project we will use the Support Vector Regression function which is a type of Support Vector Machine. A **Support Vector Regression (SVR)** is a type of supervised learning algorithm that analyses data for regression analysis. In 1996, this version of SVM for regression was proposed by Christopher J. C. Burges, Vladimir N. Vapnik, Harris Drucker, Alexander J. Smola and Linda Kaufman. The model produced by SVR depends only on a subset of the training data, because the cost function for building the model ignores any training data close to the model prediction.

We have also used the **Linear Regression** model to predict the Bitcoin prices. Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable. For example, a modeler might want to relate the weights of individuals to their heights using a linear regression model.

**HARDWARE AND SOFTWARE REQUIREMENTS**

* **HARDWARES USED:**

|  |  |
| --- | --- |
| Hardware | Used |
| RAM | 8GB |
| Processor | Intel i5 650 |
| GPU | Nvidia GeForce RTX™ 3080 |

* **SOFTWARE REQUIREMENTS:**

|  |  |
| --- | --- |
| Software | Used |
| OS | Windows 10 |
| Editor and Interpreter | Pycharm, VS Code, Jupyter Notebook |
| Programming Lang. | Python |
| Packages and Libraries | Numpy, Scikit learn, Matplotlib etc. |

**CODING**

* **For SVM Model**

import yfinance as yf

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

df=yf.download("BTC-USD")

df.reset\_index(inplace=True)

prediction\_days=30

df['Prediction']=df[['Close']].shift(-prediction\_days)

X=np.array(df.drop(['Prediction'],1))

X=X[:len(df)-prediction\_days]

Y=np.array(df['Prediction'])

Y=Y[:-prediction\_days]

from sklearn.model\_selection import train\_test\_split

x\_train,x\_test,y\_train,y\_test=train\_test\_split(X,Y,test\_size=0.2)

prediction\_days\_array=np.array(df.drop(['Prediction'],1))[-prediction\_days:]

from sklearn.svm import SVR

svr\_rbf=SVR(kernel='rbf',C=1e3,gamma=0.00001)

svr\_rbf.fit(x\_train,y\_train)

svr\_rbf\_confidence=svr\_rbf.score(x\_test,y\_test)

print("Accuracy is: %.2f"%(svr\_rbf\_confidence\*100),"%")

svm\_prediction=svr\_rbf.predict(prediction\_days\_array)

print("Predicted Price of last 30 days: ",svm\_prediction)

print()

print("Actual values of last 30 days: ",np.array(df['Close'][-prediction\_days:]))

plt.title('Bitcoin prediction - Using SVM')

plt.xlabel('Last 30 days')

plt.ylabel('Price (in USD)')

plt.plot(result\_df\_30days['Prediction'])

plt.plot(result\_df\_30days['Actual']);

plt.legend(['Prediction','Actual']);

* **For SVM Model**

import pandas as pd

import numpy as np

import datetime

from sklearn.linear\_model import LinearRegression

from sklearn import preprocessing, svm

import yfinance as yf

import matplotlib.pyplot as plt

df=yf.download("BTC-USD")

df.reset\_index(inplace=True)

df = df[['Close']]

forecast\_out = int(30)

df['Prediction'] = df[['Close']].shift(-forecast\_out)

X = np.array(df.drop(['Prediction'],1))

X = preprocessing.scale(X)

X\_forecast = X[-forecast\_out:]

X = X[:-forecast\_out]

y = np.array(df['Prediction'])

y = y[:-forecast\_out]

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2)

clf = LinearRegression()

clf.fit(X\_train,y\_train)

confidence = clf.score(X\_test, y\_test)

print("Accuracy is: %.2f"%(confidence\*100),"%")

linear\_regr\_prediction=clf.predict(X\_test)

print(linear\_regr\_prediction)

print()

print(y\_test)

forecast\_prediction = clf.predict(X\_forecast)

print(forecast\_prediction)

data={}

data['Actual']=list(actual\_last\_30)

data['Prediction']=list(forecast\_prediction)

result\_df\_30days=pd.DataFrame(data)

plt.title('Bitcoin prediction - Using SVM')

plt.xlabel('Last 30 days')

plt.ylabel('Price (in USD)')

plt.plot(result\_df\_30days['Prediction'])

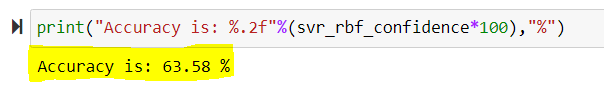
plt.plot(result\_df\_30days['Actual']);

plt.legend(['Prediction','Actual']);

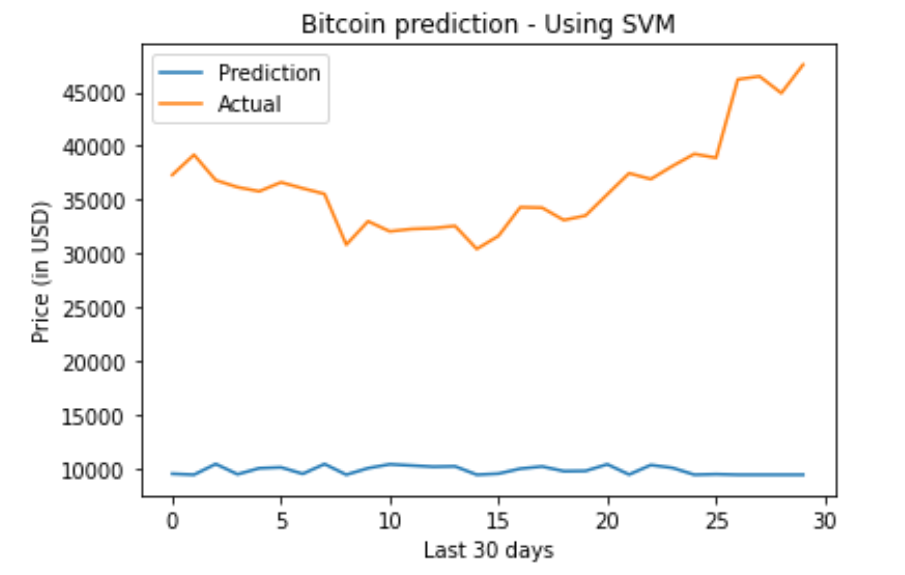
**OUTPUT**

* **For SVM Model**

**Accuracy Score:**

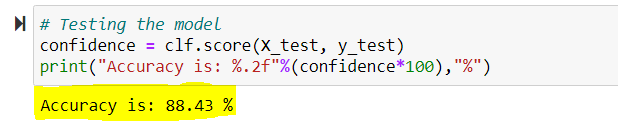


**Comparison plot for comparing 'Actual' and 'Predicted' values:**

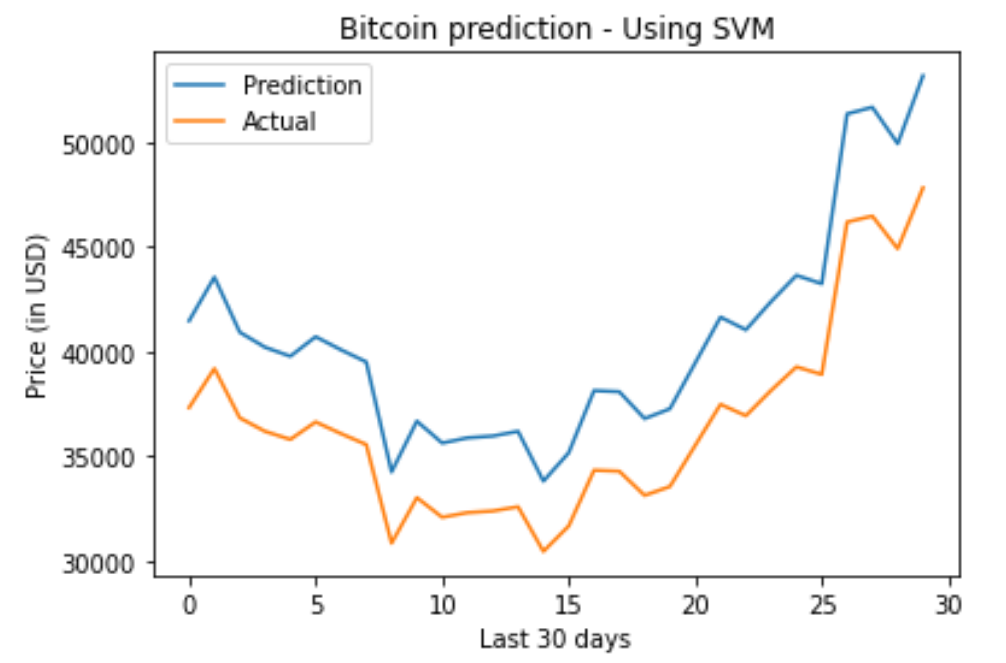


* **For Linear Regression Model**

**Accuracy Score:**



**Comparison plot for comparing 'Actual' and 'Predicted' values:**



**FUTURE SCOPE**

* Demand for crypto currencies like Bitcoin will definitely increase in near future so such projects will be a great help in predicting the Bitcoin prices.
* Market trading will increase using Bitcoin globally, so this project will be a boost in fulfilling the needs of the traders and customers.
* This project will help in maintaining the market equilibrium.
* This project will be helpful in maintaining balance of competition between the other currencies prevailing in the market.
* Encourage in developing better model with the predictive power to beat the market.

**CONCLUSION**

|  |  |
| --- | --- |
| **Model** | **Accuracy Score** |
| SVM | 63.58% |
| Linear Regression | 88.43% |

In this project we conclude that we are just introducing modules of Bitcoin price prediction and machine algorithms. Here, the Comparison table of ML algorithm model accuracy which tells that the linear regression model will have most accuracy (88.43%) than the other algorithms like SVM (Accuracy: 63.58%). Also the comparison plot between predicted and actual values came out to be more close. In this project we conclude that the linear regression algorithm is more efficient than the other algorithms. By taking help from that linear regression algorithm, we can implement the LASSO also. The time complexity reduction in bit coin price prediction using LASSO algorithm is tested by referring all other algorithms and came to a conclusion that LASSO is the best among all. The machine learning algorithms will improves that feature idea of crypto currencies. That will improves the market price of globule investments.

**BIBLIOGRAPHY AND REFERENCES**

* **Data set:** Yahoo Finance
* **Data visualization**: Matplotlib (<https://matplotlib.org/tutorials/index.html>)
* **Exploratory data analysis:** Numpy (https://numpy.org/doc/stable/user/tutorials\_index.html), Pandas, Matplotlib(https://matplotlib.org/tutorials/index.html)
* **Exploratory data analysis:** Numpy (https://numpy.org/doc/stable/user/tutorials\_index.html), Pandas, Matplotlib(<https://matplotlib.org/tutorials/index.html>)
* Mastering Python for Data Science, Sameer Madhavan 2nd Edition.
* Hands-On Machine Learning with Scikit-Learn & TensorFlow (Concepts, Tools and Techniques to Build Intelligent Systems), Aurélien Geron
* Wikipedia (https://en.wikipedia.org/wiki/Bitcoin#:~:text=Throughout%20the%20rest%20of%20the,since%20the%20all%2Dtime%20high.)
* Scikit-Learn (https://scikit-learn.org/stable/modules/svm.html)
* Learning scikit-learn: Machine Learning in Python, By Raúl Garreta, and Guillermo Moncecchi.
* S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

**\*\*\*\*\*** E.O.F. **\*\*\*\*\***