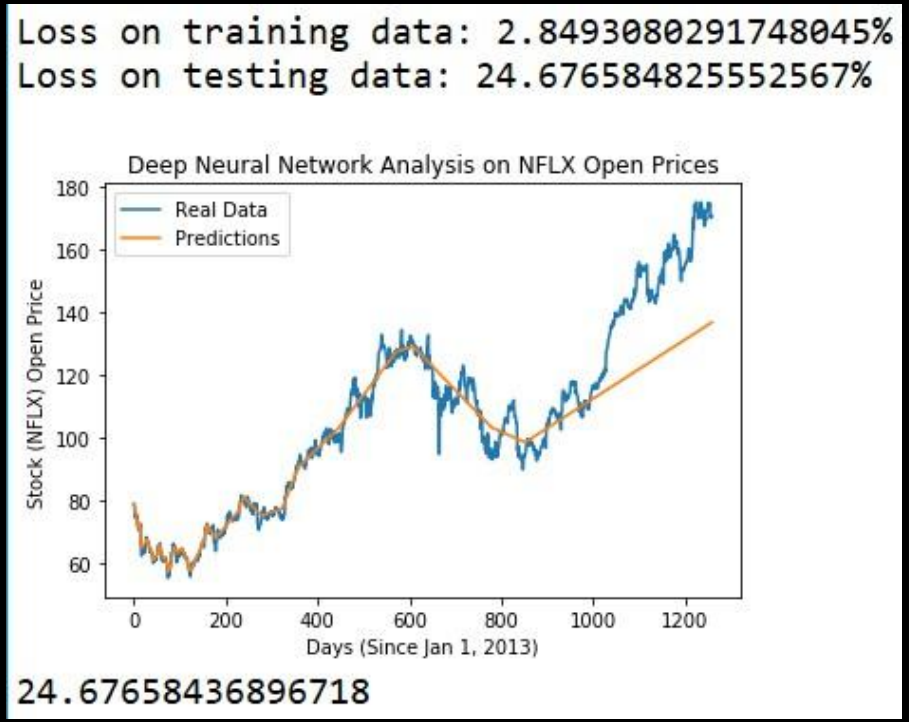
Machine Learning in the Stock Market

Akhil Ganesan & Prabhas Adivi

Introduction

Purpose

- Artificial intelligence is the future
- The stock market is an ideal challenge
- We both enjoy IT, mathematics, and CS



Experimental Design & Hypothesis/Goal

Machine Types:

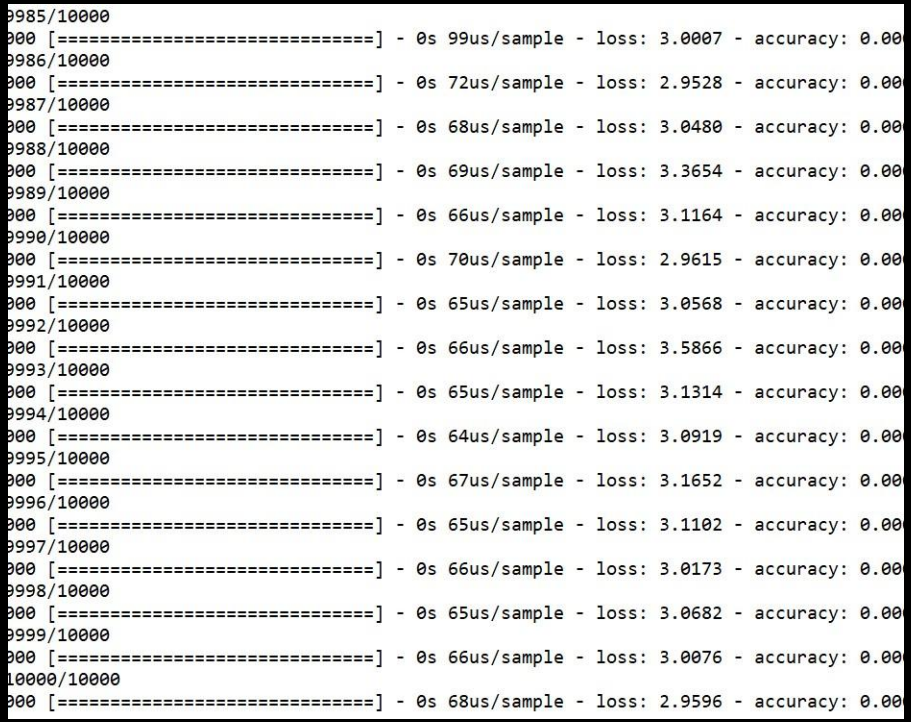
- Statistical Analysis (Lin. Reg.)
- Support Vector Regression (Guided)
- Deep Neural Network (Brain-like)

Testing for Accuracy

Constants: Environment, Return Type

Hypothesis: NN, SA, SVR

Goal: 90%, MAE(x) < 10



Materials & Methods

Materials & Procedure

Critical Materials:

- Testing Hardware/Software
- Data Sets

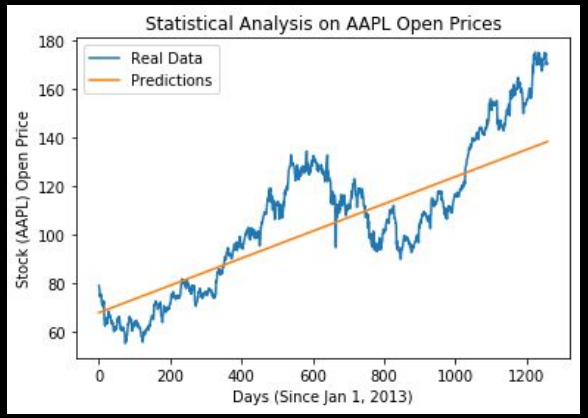
Procedure

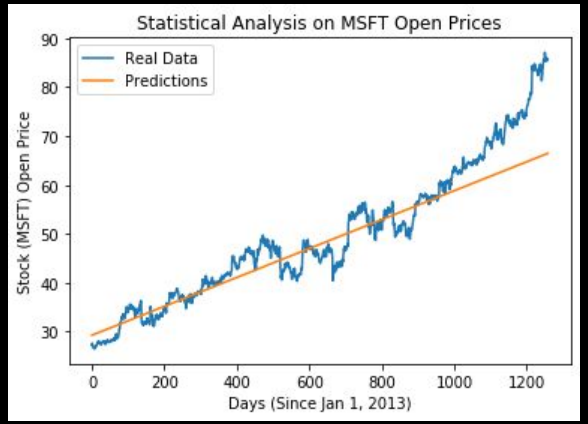
- Set up Experiment/Data
- Run training/testing; record results
    - Repeat per machine

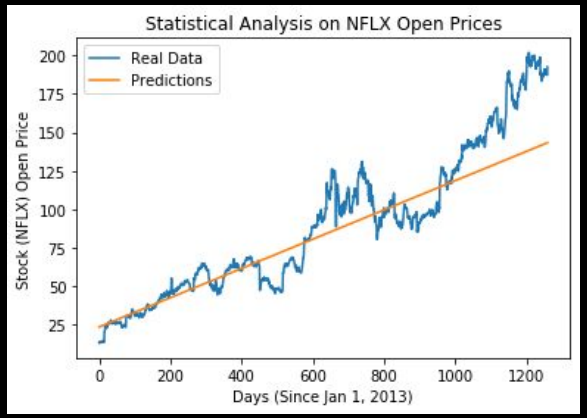
```
            full_y.append(float(row[1]))
        i += 1

train_x = np.reshape(range(0, 1000), (-1, 1))
test_x = np.reshape(range(1000, 1259), (-1, 1))
train_y = np.reshape(full_y[:1000], (-1, 1))
test_y = np.reshape(full_y[1000:], (-1, 1))
full_y = np.reshape(full_y, (-1, 1))

# Buildng the Model & Learning
model = LinearRegression()
model.fit(train_x, train_y)

# Predictions & Results
def graph(x, y):
    plt.plot(x, y, label = "Real Data")
    z = model.predict(x)
    plt.plot(x, z, label = "Predictions")

    plt.xlabel('Days (Since Jan 1, 2013)')
    plt.ylabel('Stock (AAPL) Open Price')
    plt.title('Statistical Analysis on AAPL Open Prices')
    plt.legend(loc="upper left")
    plt.show()

def mean_absolute_error(y, pred):
    a = 0
    b = 0
    for i in keras.losses.mean_absolute_error(y, pred):
        a += i
        b += 1
    return float(a)/b

graph(full_x, full_y)
print(mean_absolute_error(test_y, model.predict(test_x)))
```

Results & Discussion

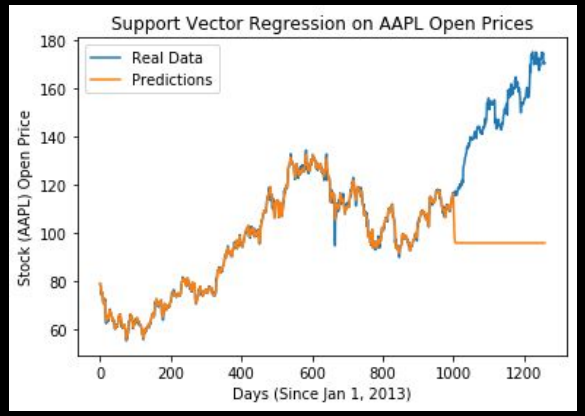
Statistical Analysis

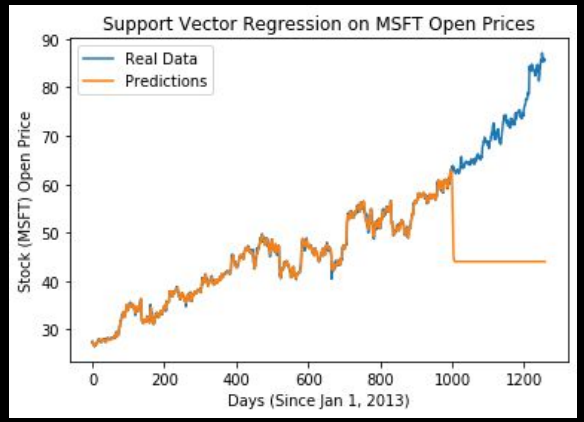


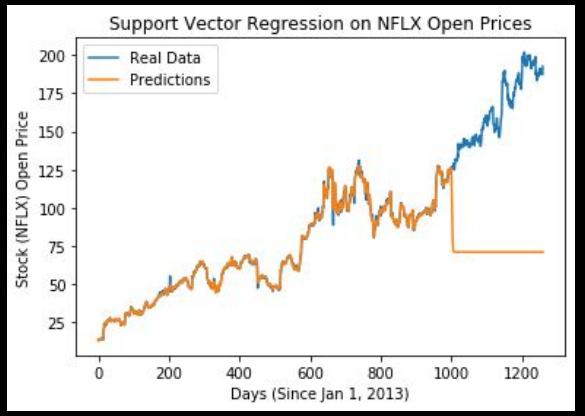




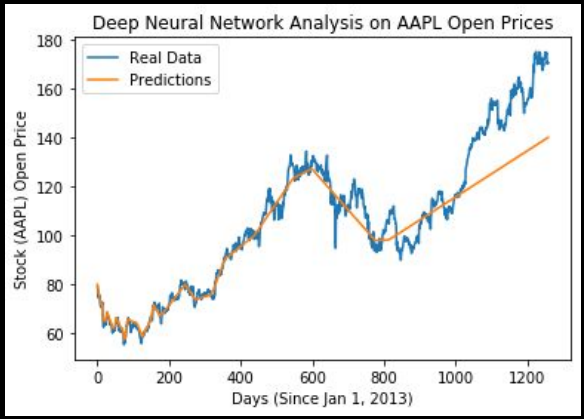
Support Vector Regression

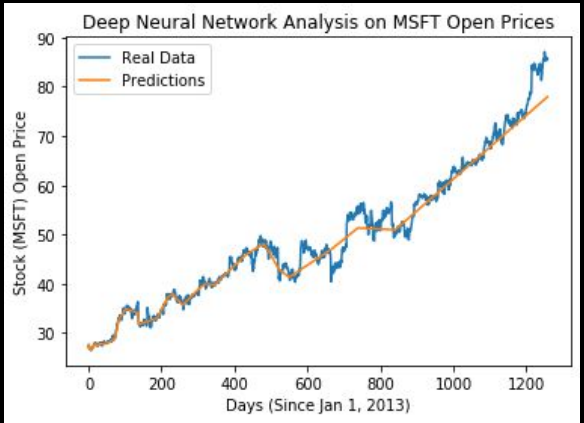


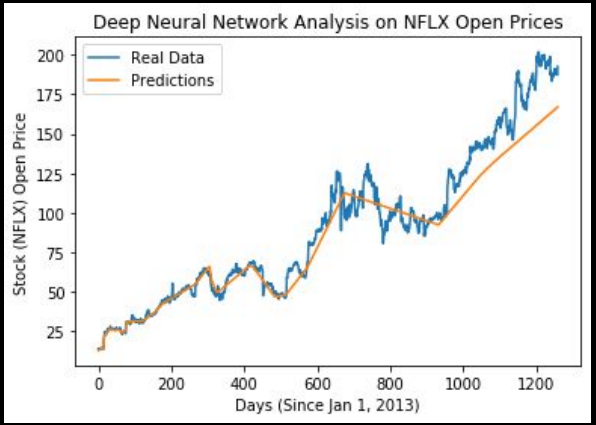




Deep Neural Network





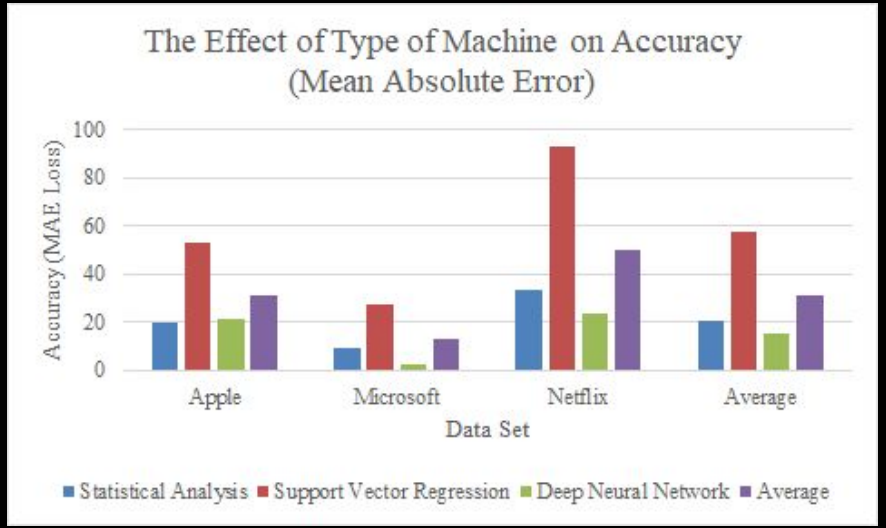


Conclusions, Limitations, & Future Action

1. Deep Neural Network (16.1)
2. Statistical Analysis (20.6)
3. Support Vector Regression (57.8)

Conclusions:

- Toughness of the data sets
- SVR expected values
- SA is best at AAPL
- MSFT NN performance



Science Fair Materials [Public Release]

Google Drive Folder: https://tinyurl.com/2020SFmat
(Contains Timeline, Research Paper, Presentation, and Machines/Datasets)

Research Paper (contained within folder): https://tinyurl.com/2020SFpaper

PowerPoint Presentation (contained within folder): https://tinyurl.com/2020SFpre

Testing Materials (contained within folder): https://tinyurl.com/2020SFmach