Simple Linear Regression Model (CalBank Case Study)

August 31, 2020

1 Question: How Liquid is the CalBank shares on Ghana Stock Exchange ?

2 Data Source : Ghana Stock Exchange - CALBank (2007 - 2019)

3 True Model: Closing Stock Price and Opening Stock Price

3.1 Suppose that Closing Stock Price is determined by the Model

4 Closing Stock Price = B_0 + B_1 Opening Stock Price + U

4.1 y = Closing Stock price and X = Opening Stock Price

4.1.1 y is also known as Dependent variable, Explained variable, Response variable, Predicted variable, Regressand and Target

4.2 X is also known as independent variable, Explanatory variable, Control variable, Predictor variable Regressor and Features

4.3 U is also known as error term or disturbance term: Representing factors other than X that affect y

4.4 We are interested in the effect of Opening Stock Price on Closing Stock Price holding other factors fixed or constant

4.5 This effect is given by B_1

4.6 The error term U contain factors such as Dividend yield, Interest rate, Firm growth, CEO performance and so on

5 Gather Data:

5.1 Cleaning the Data

5.2 Data on Features is Opening Stock Price in GHS

5.3 Data on Targets is Closing Stock Price in GHS

6 Exploring and Visualization:

[28]:
```
import pandas as pd
```

[29]:
```
data = pd.read_csv("CalBank.csv")
```

[30]:
```
print(data)
```

```
            Date Share_Code  Year_High_GHS  Year_Low_GHS  \
0     25/06/2007        CAL         0.2700          0.20
1     26/06/2007        CAL         0.2702          0.20
2     27/06/2007        CAL         0.2705          0.20
3     28/06/2007        CAL         0.2715          0.20
```

```
4     29/06/2007        CAL         0.2720          0.20
…          …        …            …           …
3078  23/12/2019        CAL         1.0800          0.64
3079  24/12/2019        CAL         1.0800          0.64
3080  27/12/2019        CAL         1.0800          0.64
3081  30/12/2019        CAL         1.0800          0.64
3082  31/12/2019        CAL         1.0800          0.64

      Previous_Closing_Price_VWAP_GHS  Opening_Price_GHS  Closing_Price_GHS  \
0                              0.2646             0.2646             0.2700
1                              0.2700             0.2700             0.2702
2                              0.2702             0.2702             0.2705
3                              0.2705             0.2705             0.2715
4                              0.2715             0.2715             0.2720
…                               …                …                …
3078                           0.8100             0.8100             0.8100
3079                           0.8100             0.8100             0.8100
3080                           0.8100             0.8100             0.8200
3081                           0.8200             0.8200             0.8200
3082                           0.8200             0.8200             0.8900

      Price_Change_GHS  Closing_Bid_Price_GHS  Closing_Offer_Price_GHS  \
0               0.0054                 0.2700                     0.00
1               0.0002                 0.2700                     0.00
2               0.0003                 0.2702                     0.00
3               0.0010                 0.2715                     0.00
4               0.0005                 0.2715                     0.00
…                …                    …                      …
3078            0.0000                 0.8100                      NaN
3079            0.0000                 0.8100                      NaN
3080            0.0100                 0.8100                     0.82
3081            0.0000                 0.8200                      NaN
3082            0.0700                    NaN                     0.82

     Total_Shares_Traded Total_Value_Traded Last_Transactio_ Price_GHS
0                2142400               NaN                          0
1                   6500               NaN                          0
2                   6000               NaN                          0
3                 741200               NaN                          0
4                   3000               NaN                          0
…                  …                …                        …
3078                   0                  0                       0.82
3079               1,000                810                       0.82
3080              50,000          41,000.00                       0.82
3081                   0                  0                       0.82
3082             216,200         193,159.00                       0.82

[3083 rows x 13 columns]
```

```
[31]: data.describe()
```

```
[31]:        Year_High_GHS  Year_Low_GHS  Previous_Closing_Price_VWAP_GHS  \
      count    3083.000000   3083.000000                      3083.000000
      mean        0.810767      0.555953                         0.689800
      std         0.401668      0.310753                         0.348338
      min         0.200000      0.170000                         0.170000
      25%         0.390000      0.260000                         0.310000
      50%         0.800000      0.442000                         0.760000
      75%         1.060000      0.850000                         0.980000
      max         1.970000      1.400000                         1.970000

             Opening_Price_GHS  Closing_Price_GHS  Price_Change_GHS  \
      count        3083.000000        3083.000000       3083.000000
      mean            0.689631           0.690109          0.001272
      std             0.348286           0.348606          0.029425
      min             0.170000           0.170000         -0.200000
      25%             0.310000           0.310000          0.000000
      50%             0.760000           0.760000          0.000000
      75%             0.980000           0.980000          0.000000
      max             1.970000           1.970000          0.690000

             Closing_Bid_Price_GHS  Closing_Offer_Price_GHS
      count            2389.000000              2571.000000
      mean                0.626448                 1.596433
      std                 0.385603                38.729524
      min                 0.000000                 0.000000
      25%                 0.273500                 0.290000
      50%                 0.750000                 0.740000
      75%                 0.950000                 0.980000
      max                 3.290000              1900.000000
```

```
[32]: type(data)
```

```
[32]: pandas.core.frame.DataFrame
```

```
[33]: stockPrice = data[["Opening_Price_GHS", "Closing_Price_GHS"]]
```

```
[34]: type(stockPrice)
```

```
[34]: pandas.core.frame.DataFrame
```

```
[35]: print(stockPrice)
```

```
           Opening_Price_GHS  Closing_Price_GHS
     0                0.2646             0.2700
     1                0.2700             0.2702
```

```
2                 0.2702            0.2705
3                 0.2705            0.2715
4                 0.2715            0.2720
…                  …                 …
3078              0.8100            0.8100
3079              0.8100            0.8100
3080              0.8100            0.8200
3081              0.8200            0.8200
3082              0.8200            0.8900

[3083 rows x 2 columns]
```

[36]:

```
stockPrice.describe()
```

[36]:

```
       Opening_Price_GHS  Closing_Price_GHS
count        3083.000000        3083.000000
mean            0.689631           0.690109
std             0.348286           0.348606
min             0.170000           0.170000
25%             0.310000           0.310000
50%             0.760000           0.760000
75%             0.980000           0.980000
max             1.970000           1.970000
```

[37]:

```
#y = data[["Closing_Price_GHS"]]
#X = data[["Opening_Price_GHS"]]
```

[38]:

```
y = stockPrice[["Closing_Price_GHS"]]
X = stockPrice[["Opening_Price_GHS"]]
```

[43]:

```
import matplotlib.pyplot as plt
```

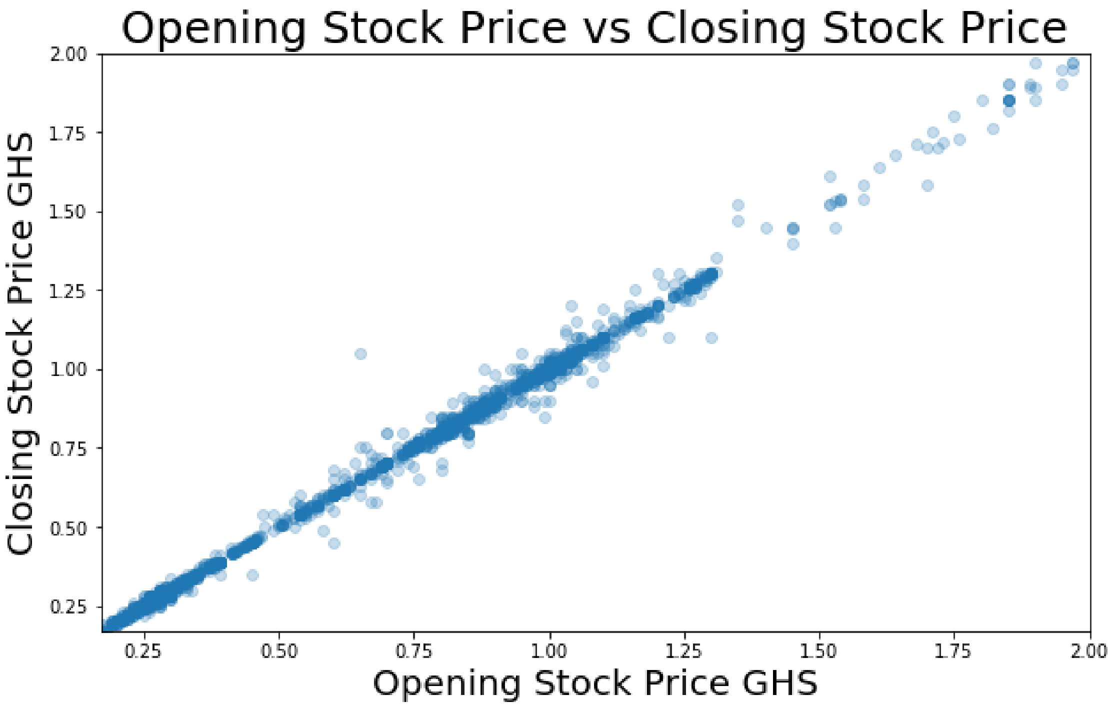
6.1 Scatter Plot

[66]:

```
%matplotlib inline
plt.figure(figsize=(10,6))
plt.scatter(X,y, alpha=0.25)
plt.title("Opening Stock Price vs Closing Stock Price", fontsize=25)
plt.xlabel("Opening Stock Price GHS" , fontsize=20)
plt.ylabel("Closing Stock Price GHS" , fontsize=20)
plt.ylim(0.17,2)
plt.xlim(0.17,2)
plt.show
```

[66]:

```
<function matplotlib.pyplot.show(*args, **kw)>
```



7 Hypothesize or Fitted Model:

7.1 we can draw whole lot of line through the data to estimate the parameter B_0 and B_1 But our job is to find the best possible line that minimize this residual and the method we use is OLS - Ordinary Least Square

8 Run and Evaluate Rgression

[67]:
```
from sklearn.linear_model import LinearRegression
```

[68]:
```
reg = LinearRegression()
```

[69]:
```
reg.fit(X,y)
```

[69]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

8.1 slope coeficient: B_1

[71]:
```
reg.coef_
```

[71]:
```
array([[0.99950251]])
```

8.2 Intercept: B_0

[72]:

```
reg.intercept_
```

[72]:

```
array([0.00082142])
```

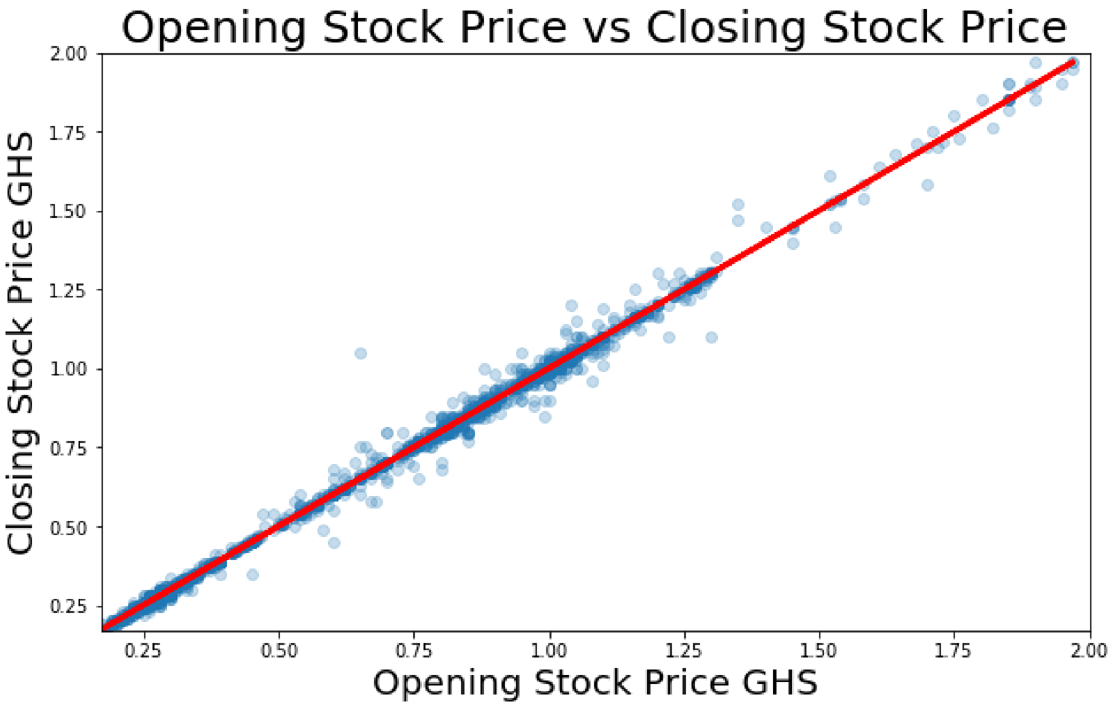
9 Plotting

[74]:

```
%matplotlib inline
plt.figure(figsize=(10,6))
plt.scatter(X,y, alpha=0.25)
plt.title("Opening Stock Price vs Closing Stock Price", fontsize=25)
plt.xlabel("Opening Stock Price GHS" , fontsize=20)
plt.ylabel("Closing Stock Price GHS" , fontsize=20)
plt.ylim(0.17,2)
plt.xlim(0.17,2)
plt.plot(X,reg.predict(X), color="red",linewidth=3)
plt.show
```

[74]:

```
<function matplotlib.pyplot.show(*args, **kw)>
```



10 Estimated or Fitted Model

11 y = 0.00 + 0.99X

11.1 where y = Closing Stock Price , X = Opening Stock Price , $B_0 = 0.00$ and $B_1 = 0.99$

11.2 The intercept B_0 literally means an investor with zero investment in CalBank shares or equities will earn zero returns

11.3 The slope B_1 implies that an investor with 1GHS investment on CalBank shares will loss money after going short. Thus wont make profit on his investment

12 Goodness of fit of the model: R^2

```
[75]: reg.score(X,y)
```

```
[75]: 0.9971702603458394
```

12.1 $R^2 = 0.997$

12.2 The Opening Stock Price explains over 99% of the variation in the Closing Stock Price of CalBank

13 Conclusion

13.1 In conclusion liquidity is not permenant, CalBank shares is not actively traded on the Ghana Stock Exchange(GSE) and as a result investor would hold their stocks beacuse they not willing to traded at fall price. my advice to investor is to invest in money market intrument for short term investment

14 Prepared by PAUL OFFEI

```
[ ]:
```