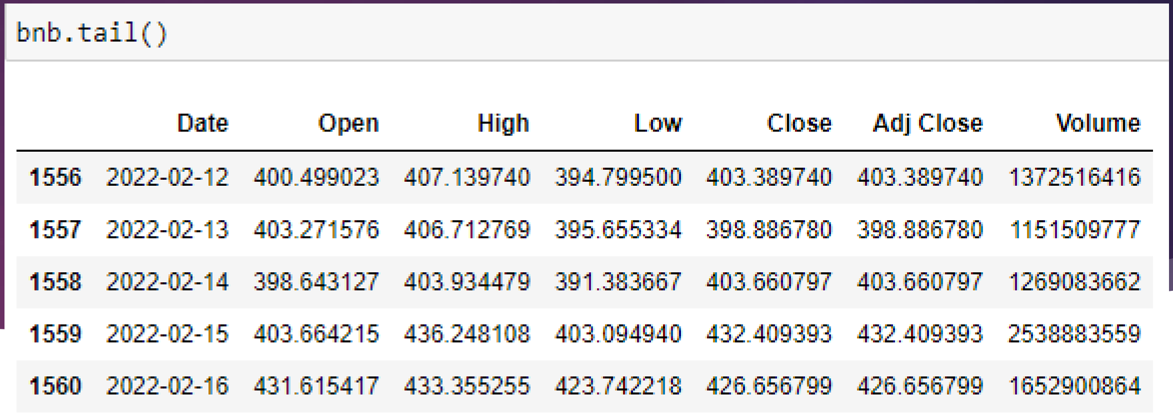
Cryptocurrency Price Prediction with BNB

Table of contents

1. Data Overview
2. Demographic
   1. Show dataset plot graph
3. Predictive Models
   1. Linear Regression
   2. KNearestNeighbors
4. Optimizing Models
5. Comparison

Data Overview

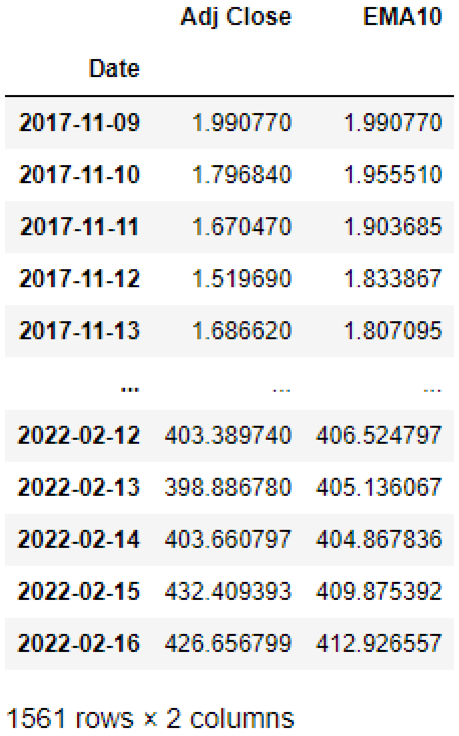


```
bnb.tail()
```

	Date	Open	High	Low	Close	Adj Close	Volume
1556	2022-02-12	400.499023	407.139740	394.799500	403.389740	403.389740	1372516416
1557	2022-02-13	403.271576	406.712769	395.655334	398.886780	398.886780	1151509777
1558	2022-02-14	398.643127	403.934479	391.383667	403.660797	403.660797	1269083662
1559	2022-02-15	403.664215	436.248108	403.094940	432.409393	432.409393	2538883559
1560	2022-02-16	431.615417	433.355255	423.742218	426.656799	426.656799	1652900864

- Dataset source:

  https://www.kaggle.com/varpit94/binancecoin-data
- Original dataset has 1561 rows & 7 columns
  - Consists of BNB coin's Open, High, Low, Close, Adj Close, and Volume
  - From 9 Nov 2017 to 16 Feb 2022
- The dataset was very neat and had no missing values
  - Little to no data cleaning was done
- Cut down dataset to just 3 columns
  - Date as index, Adj Close price and Exponential Moving Average

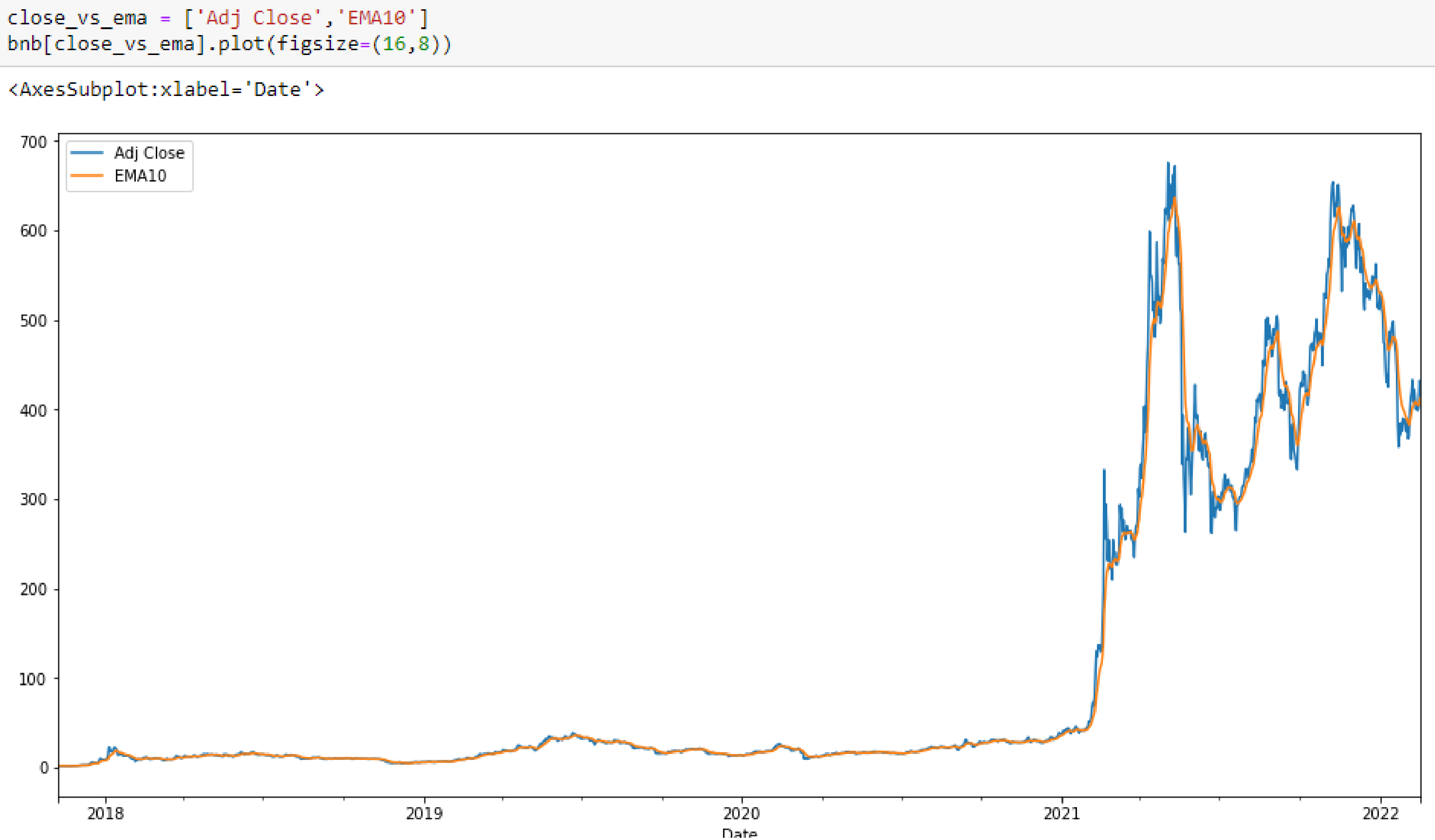


	Adj Close	EMA10
Date		
2017-11-09	1.990770	1.990770
2017-11-10	1.796840	1.955510
2017-11-11	1.670470	1.903685
2017-11-12	1.519690	1.833867
2017-11-13	1.686620	1.807095
...	...	...
2022-02-12	403.389740	406.524797
2022-02-13	398.886780	405.136067
2022-02-14	403.660797	404.867836
2022-02-15	432.409393	409.875392
2022-02-16	426.656799	412.926557

1561 rows × 2 columns

Demographics

- Plot our graph using Adj Close & EMA values over date period in dataset

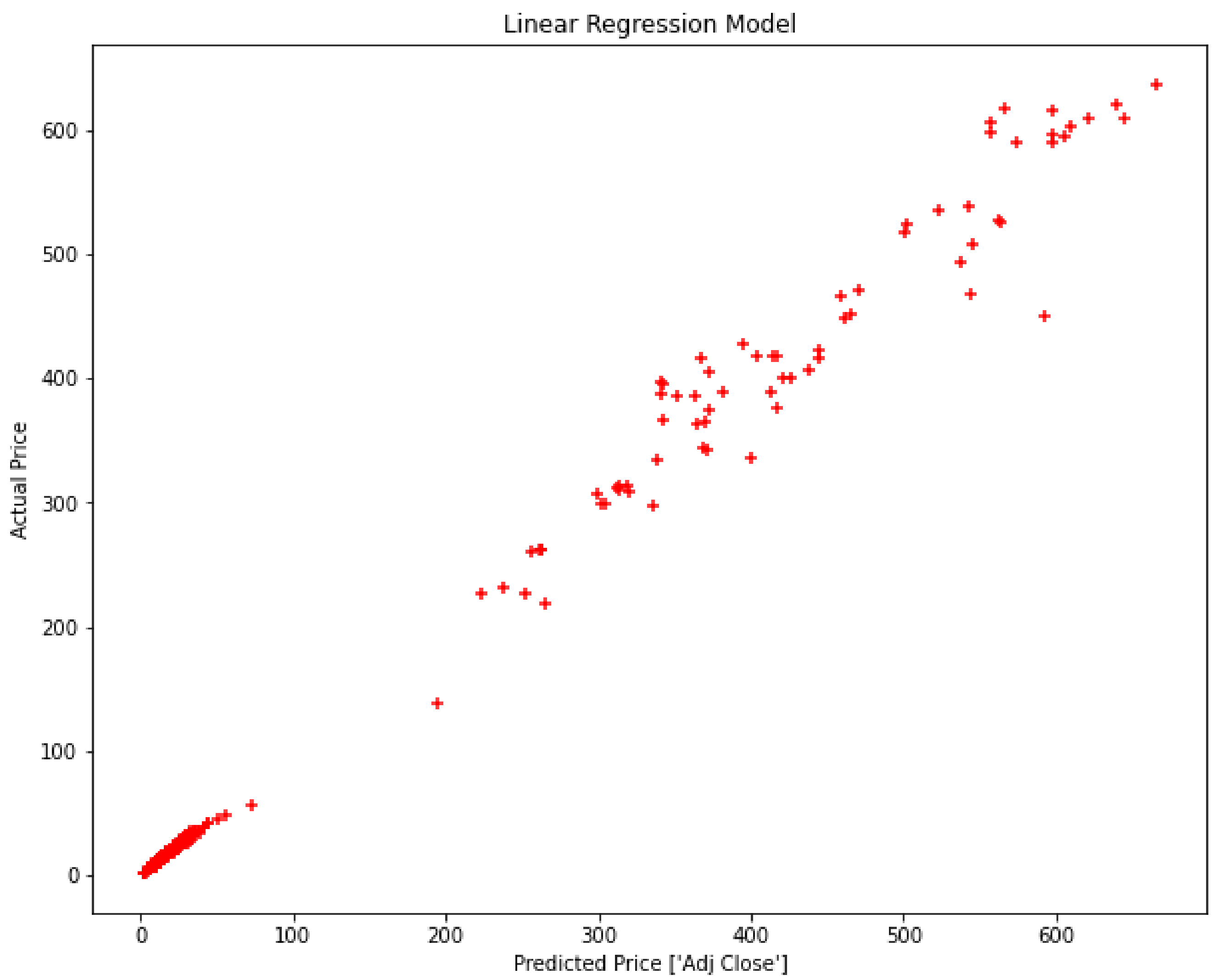


```
close_vs_ema = ['Adj Close','EMA10']
bnb[close_vs_ema].plot(figsize=(16,8))
```

```
<AxesSubplot:xlabel='Date'>
```

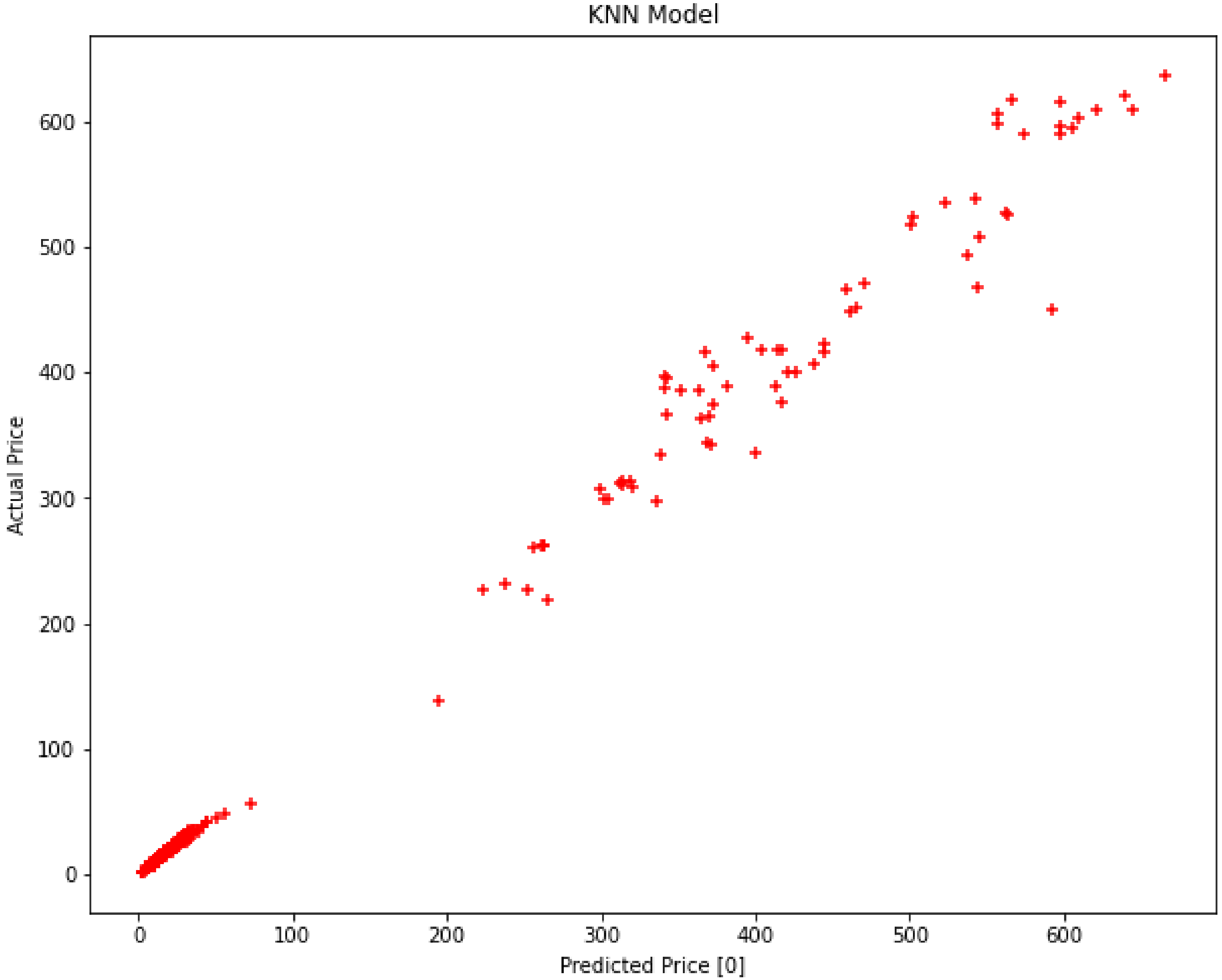
Predictive Models

- Linear Regression
  - RootMSE: 15.8368
  - MSE: 250.803
  - R-Squared(r2): 0.99210
  - Adjusted r2: 0.99207
  - Coefficients: 0.98948
  - Intercept: 0.35736



Predictive Models

- KNearestNeighbors
  - RootMSE: 17.2305
  - MSE: 296.892
  - R-Squared(r2): 0.99065



Optimize Predictive models

- Used GridSearchCV to identify best params to use.
- KNN, n_neighbors = 6, as our new param ,
  - RootMSE: 17.2305
  - MSE: 296.892
  - R-squared(r2): 0.99065

  - Values were same as KNN model without using new params

Comparison

Model/Accuracy	Linear Regression	KNearest Neighbors
RMSE	15.8368	17.2305
MSE	250.803	296.892
R-squared	0.99210	0.99065

As we can see from the table that the Linear Regression Model has better accuracy than the KNN model for predicting our Cryptocurrency price.