Multiple Linear Regression of BTC and Other Market Sectors

In [28]:

```
import pandas as pd
import numpy as np
import seaborn as sb
import matplotlib.pyplot as plt
```

For my final project I am looking to determine the correlation between Bitcoin and other market sectors. There after I am looking to develop a multiple linear regression model to price BTC. I believe bitcoin will be positively correlated to SPY and Gold for these securities also serve as inflation hedges. And BTC will be negatively correlated with VXX.

Looking at the distribution of Bitcoins historical prices, I believe it will be positively skewed as Bitcoin typically grows exponentially after a halving.

In [29]:

```
data=  pd.read_excel(r"C:\Users\mdmoh\Desktop\Historicals_1.xlsx")
```

This data sourced from Bloomberg and requires little cleaning. However, Unnamed columns 2 - 16 needs to be dropped and Unnamedmed column 0 needs to be renamed to 'Date' andd set as an index. Through the isna().sum() funtion I can see this data has no missing values.

In [30]:

```
data.head()
```

Out[30]:

	Unnamed: 0	DXY	Unnamed: 2	SPY	Unnamed: 4	BTC	Unnamed: 6	10 year note	Unnamed: 8	Oil (XOP Equity)	Unnamed: 10	VXX	Unnamed: 12	(Eq
0	2016-01-04	98.856	2016-01-04	182.5136	2016-01-04	434.94	2016-01-04	2.2428	2016-01-04	121.48	2016-01-04	284.0	2016-01-04	10
1	2016-01-05	99.348	2016-01-05	182.8230	2016-01-05	432.60	2016-01-05	2.2357	2016-01-05	121.08	2016-01-05	274.2	2016-01-05	10
2	2016-01-06	99.248	2016-01-06	180.5169	2016-01-06	430.80	2016-01-06	2.1702	2016-01-06	112.76	2016-01-06	283.2	2016-01-06	10
3	2016-01-07	98.305	2016-01-07	176.1860	2016-01-07	455.36	2016-01-07	2.1455	2016-01-07	109.96	2016-01-07	313.6	2016-01-07	10
4	2016-01-08	98.377	2016-01-08	174.2548	2016-01-08	452.42	2016-01-08	2.1156	2016-01-08	110.48	2016-01-08	329.8	2016-01-08	10

In [31]:

```
data.columns
```

Out[31]:

```
Index(['Unnamed: 0', 'DXY ', 'Unnamed: 2', 'SPY ', 'Unnamed: 4', 'BTC',
       'Unnamed: 6', '10 year note', 'Unnamed: 8', 'Oil (XOP Equity)',
       'Unnamed: 10', 'VXX', 'Unnamed: 12', 'Gold (GLD Equity)', 'Unnamed: 14',
       'Real Eatate (VNQ Equity)', 'Unnamed: 16', 'Bonds (AGG Equity)'],
      dtype='object')
```

In [32]:

```
data2 =  data.drop(['Unnamed: 2', 'Unnamed: 4',  'Unnamed: 6',  'Unnamed: 8',  'Unnamed: 10',  'Unnamed: 12',  'U
nnamed: 14',
 'Unnamed: 16'], axis=1)
```

In [33]:

```
data2.head()
```

Out[33]:

	Unnamed: 0	DXY	SPY	BTC	10 year note	Oil (XOP Equity)	VXX	Gold (GLD Equity)	Real Eatate (VNQ Equity)	Bonds (AGG Equity)
0	2016-01-04	98.856	182.5136	434.94	2.2428	121.48	284.0	102.89	78.77	107.97
1	2016-01-05	99.348	182.8230	432.60	2.2357	121.08	274.2	103.18	80.29	108.02
2	2016-01-06	99.248	180.5169	430.80	2.1702	112.76	283.2	104.67	80.07	108.43
3	2016-01-07	98.305	176.1860	455.36	2.1455	109.96	313.6	106.15	78.51	108.42
4	2016-01-08	98.377	174.2548	452.42	2.1156	110.48	329.8	105.68	77.46	108.66

In [34]:

```
data2 = data2.rename(columns={'Unnamed: 0': 'Date'})
data2.head()
```

Out[34]:

	Date	DXY	SPY	BTC	10 year note	Oil (XOP Equity)	VXX	Gold (GLD Equity)	Real Eatate (VNQ Equity)	Bonds (AGG Equity)
0	2016-01-04	98.856	182.5136	434.94	2.2428	121.48	284.0	102.89	78.77	107.97
1	2016-01-05	99.348	182.8230	432.60	2.2357	121.08	274.2	103.18	80.29	108.02
2	2016-01-06	99.248	180.5169	430.80	2.1702	112.76	283.2	104.67	80.07	108.43
3	2016-01-07	98.305	176.1860	455.36	2.1455	109.96	313.6	106.15	78.51	108.42
4	2016-01-08	98.377	174.2548	452.42	2.1156	110.48	329.8	105.68	77.46	108.66

In [35]:

```
data3 = data2.set_index('Date')
data3.head()
```

Out[35]:

	DXY	SPY	BTC	10 year note	Oil (XOP Equity)	VXX	Gold (GLD Equity)	Real Eatate (VNQ Equity)	Bonds (AGG Equity)
Date									
2016-01-04	98.856	182.5136	434.94	2.2428	121.48	284.0	102.89	78.77	107.97
2016-01-05	99.348	182.8230	432.60	2.2357	121.08	274.2	103.18	80.29	108.02
2016-01-06	99.248	180.5169	430.80	2.1702	112.76	283.2	104.67	80.07	108.43
2016-01-07	98.305	176.1860	455.36	2.1455	109.96	313.6	106.15	78.51	108.42
2016-01-08	98.377	174.2548	452.42	2.1156	110.48	329.8	105.68	77.46	108.66

In [36]:

```
data3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1341 entries, 2016-01-04 to 2021-02-22
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   DXY                       1341 non-null   float64
 1   SPY                       1341 non-null   float64
 2   BTC                       1341 non-null   float64
 3   10 year note              1341 non-null   float64
 4   Oil (XOP Equity)          1341 non-null   float64
 5   VXX                       1341 non-null   float64
 6   Gold (GLD Equity)         1341 non-null   float64
 7   Real Eatate (VNQ Equity)  1341 non-null   float64
 8   Bonds (AGG Equity)        1341 non-null   float64
dtypes: float64(9)
memory usage: 104.8 KB
```

In [37]:

```
data3.isna().sum()
```

Out[37]:

```
DXY                         0
SPY                         0
BTC                         0
10 year note                0
Oil (XOP Equity)            0
VXX                         0
Gold (GLD Equity)           0
Real Eatate (VNQ Equity)    0
Bonds (AGG Equity)          0
dtype: int64
```

In [38]:

```
data3.describe()
```

Out[38]:

	DXY	SPY	BTC	10 year note	Oil (XOP Equity)	VXX	Gold (GLD Equity)	Real Eatate (VNQ Equity)	Bonds (AGG Equity)
count	1341.000000	1341.000000	1341.000000	1341.000000	1341.000000	1341.000000	1341.000000	1341.000000	1341.000000
mean	95.894687	260.029184	7021.863490	1.993856	116.136447	59.654499	132.606952	83.200059	110.818538
std	3.061263	50.920385	7243.483629	0.729748	39.180355	73.878088	20.767450	5.697471	3.989754
min	88.627000	166.026100	368.690000	0.506900	30.160000	10.930000	102.890000	56.910000	104.010000
25%	93.722000	220.545000	1251.570000	1.578100	86.880000	21.990000	118.970000	80.000000	108.040000
50%	96.264000	258.854000	6477.560000	2.125600	125.640000	29.490000	123.650000	83.160000	109.820000
75%	97.853000	289.593000	9314.920000	2.536700	145.440000	52.760000	141.390000	85.940000	112.910000
max	103.270000	392.640000	55629.080000	3.237300	178.280000	395.200000	193.890000	99.570000	119.630000

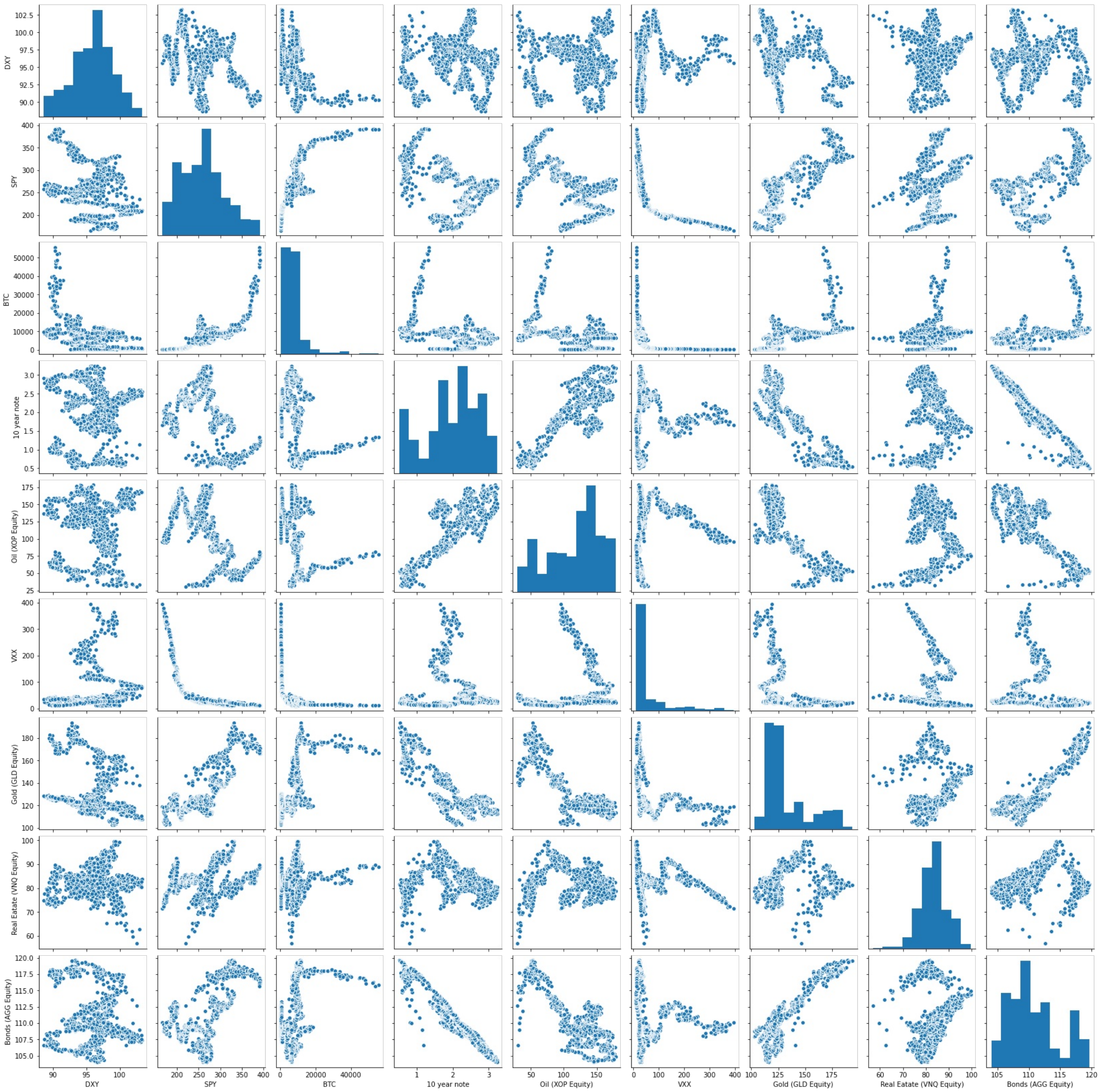
For the correlation I have determined it three ways, pairwise relationships, heatmap and matrix. Herein, it is evident that BTC is positively correlated with SPY, GLD and to a lesser extend Bonds. And BTC is negatively correlated with DXY, Oil, VXX and to a lesser extent US 10 year treasury yields. BTC is not very correlated with Real Estate.

In [39]:

```
sb.pairplot(data3)
```

Out[39]:

```
<seaborn.axisgrid.PairGrid at 0x1b9c871e0c8>
```

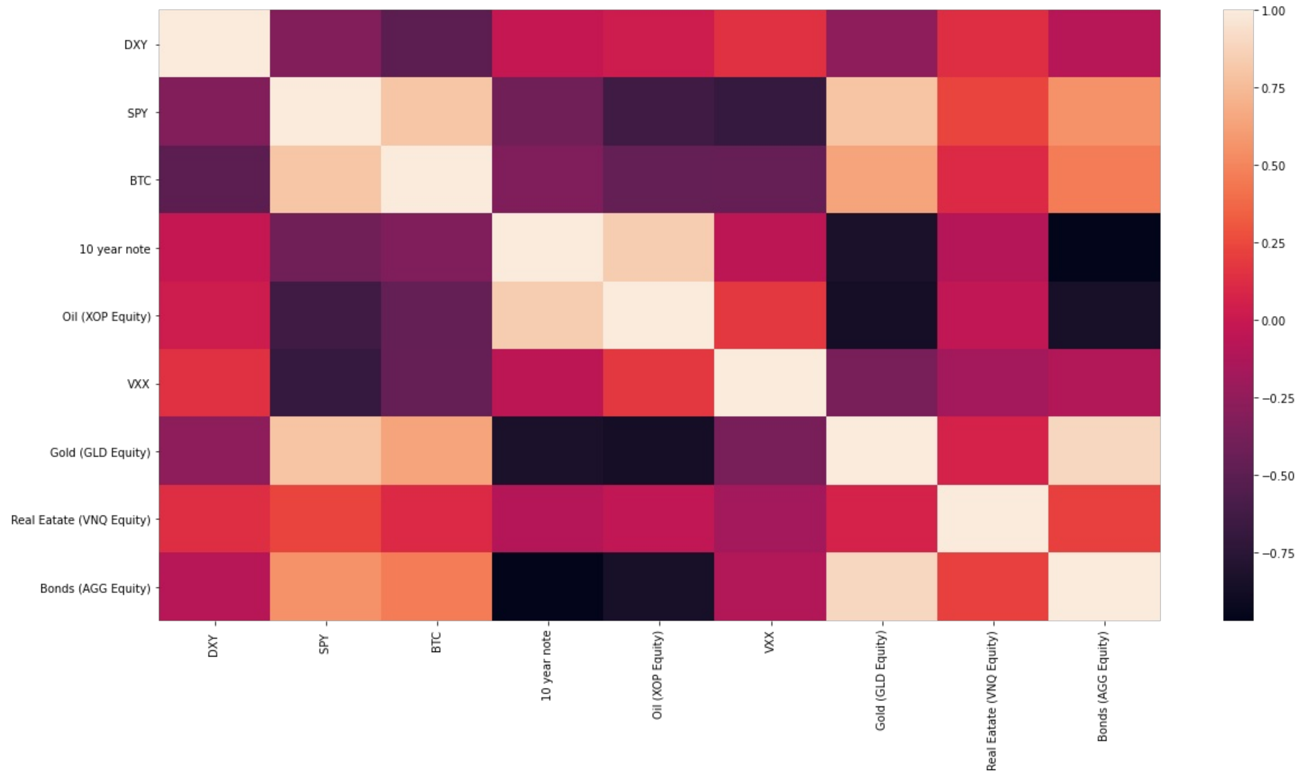


In [40]:

```
corr_data = data3.corr()
sb.heatmap(corr_data)
```

Out[40]:

<AxesSubplot:>



In [41]:

```
data3.corr()
```

Out[41]:

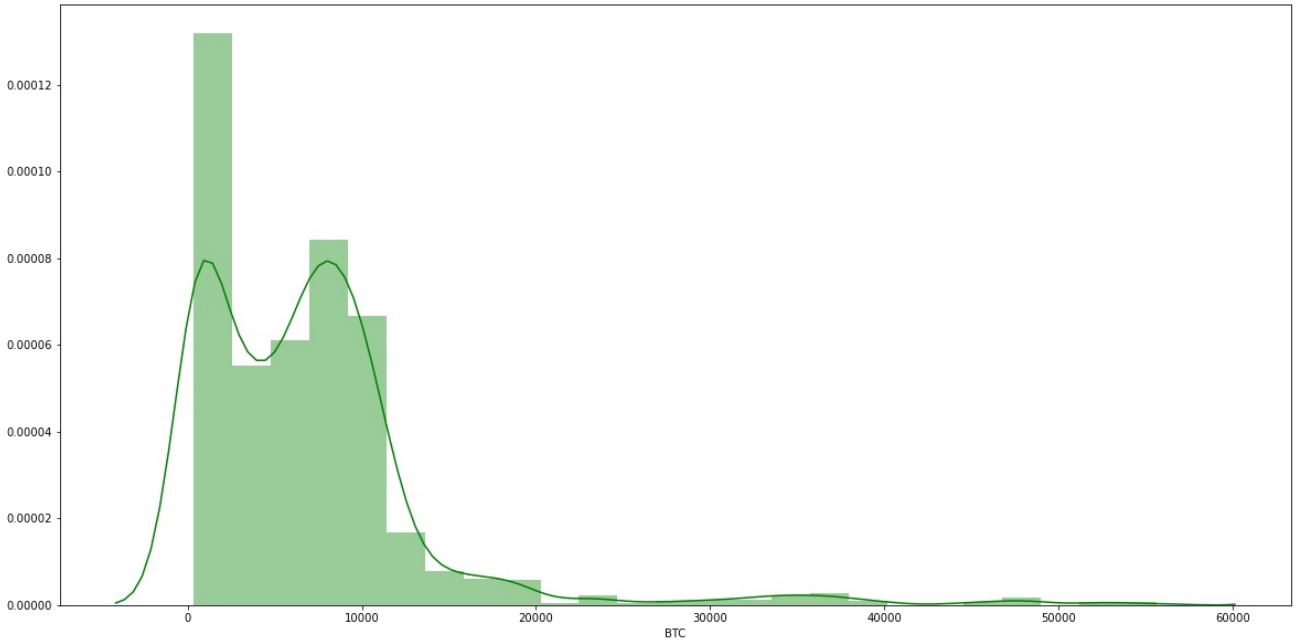
	DXY	SPY	BTC	10 year note	Oil (XOP Equity)	VXX	Gold (GLD Equity)	Real Eatate (VNQ Equity)	Bonds (AGG Equity)
DXY	1.000000	-0.323442	-0.495259	-0.012343	0.022356	0.146825	-0.269577	0.135678	-0.082928
SPY	-0.323442	1.000000	0.804504	-0.404955	-0.636442	-0.693671	0.795333	0.230360	0.556623
BTC	-0.495259	0.804504	1.000000	-0.334848	-0.462188	-0.451688	0.644313	0.111905	0.461312
10 year note	-0.012343	-0.404955	-0.334848	1.000000	0.840300	-0.050804	-0.831707	-0.101244	-0.971594
Oil (XOP Equity)	0.022356	-0.636442	-0.462188	0.840300	1.000000	0.176689	-0.859576	-0.036211	-0.843468
VXX	0.146825	-0.693671	-0.451688	-0.050804	0.176689	1.000000	-0.365836	-0.175142	-0.104347
Gold (GLD Equity)	-0.269577	0.795333	0.644313	-0.831707	-0.859576	-0.365836	1.000000	0.071524	0.897784
Real Eatate (VNQ Equity)	0.135678	0.230360	0.111905	-0.101244	-0.036211	-0.175142	0.071524	1.000000	0.214838
Bonds (AGG Equity)	-0.082928	0.556623	0.461312	-0.971594	-0.843468	-0.104347	0.897784	0.214838	1.000000

For the last 5 years of historical prices Bitcoin has a bimodal distribution. This is explained through its market cycles. After the halving's Bitcoin sees exponential growth and then a prolonged bear market, where bitcoin trades within a range for years until around the next halving (every 210,000 blocks).

In [42]:

```
bit_dist = pd.Series(data3['BTC'])

sb.distplot(bit_dist, bins=25, color="g")
plt.show()
```

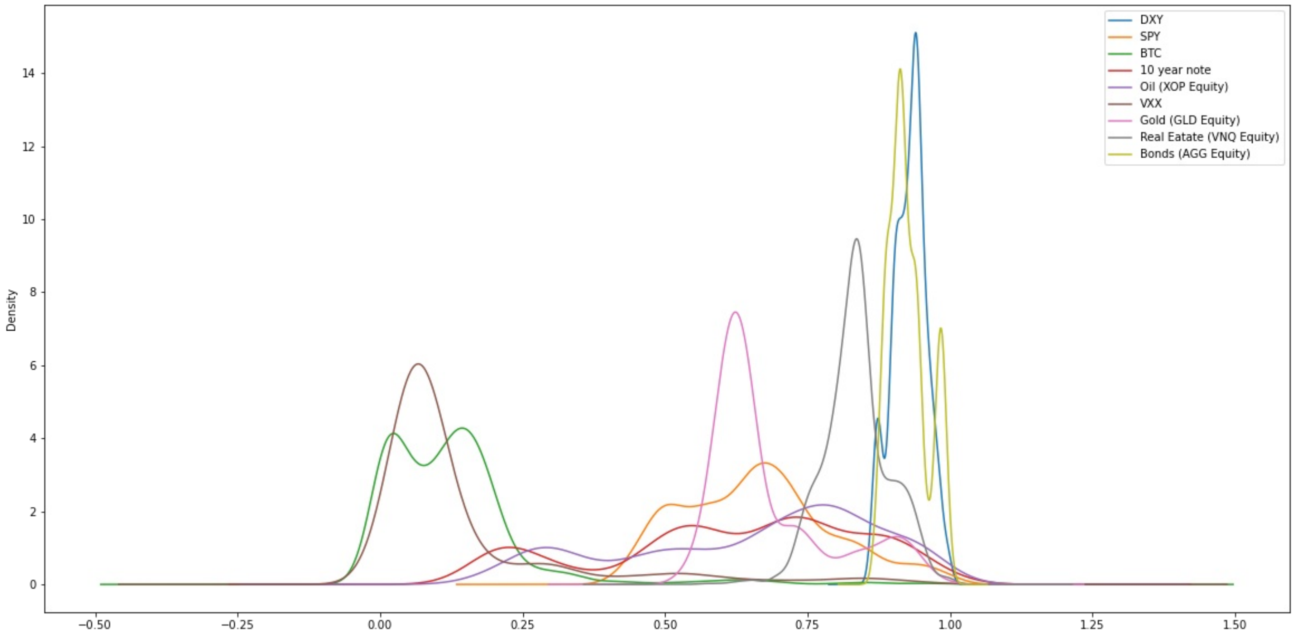


For the following distribution graph, I have normalized the prices for each security. Here by comparing the distribution we can visualize the standard deviation of each security. The security with a steep distribution such as DXY and AGG Equity have a lower standard deviation than the security with a wider distribution such as SPY and BTC.

In [43]:

```
data4 = pd.DataFrame()
for column in data3.columns:
    data4[column] = data3[column]/max(data3[column])

data4.plot.kde()
plt.rcParams['figure.figsize'] = (20,10)
```



The multiple linear regression model here is done using sklearn. The model has an R2 of .75. I have chosen sklearn tfor the linear regression because it makes it quite easy to input the variables and determine the dependent varaible.

In [44]:

```
from sklearn import linear_model

X = data3[['DXY ', 'SPY ', '10 year note', 'Oil (XOP Equity)', 'Gold (GLD Equity)', 'Real Eatate (VNQ Equity)', '
Bonds (AGG Equity)', 'VXX']]
Y = data3['BTC']


regr = linear_model.LinearRegression()
regr.fit(X, Y)

print('Intercept: \n', regr.intercept_)
print('Coefficients: \n', regr.coef_)
regr.score(X,Y)
```

```
Intercept: 
 42615.6465807921
Coefficients: 
 [ -622.70677554   194.85395731 -5301.24162413    53.77320964
  -268.15451381  -201.8810751    263.22491568    16.14506826]
```

Out[44]:

```
0.7561183325994455
```

In [45]:

```
DXY = 80
SPY = 310
ten_year_note = 1.2
Oil_XOP_Equity = 85
Gold_GLD_Equity = 180
Real_Eatate_VNQ_Equity = 90
Bonds_AGG_Equity = 155
VXX = 13

print ('Predicted BTC Price: \n', regr.predict([[DXY, SPY, ten_year_note, Oil_XOP_Equity, Gold_GLD_Equity, Real_E
atate_VNQ_Equity, Bonds_AGG_Equity, VXX]]))
```

```
Predicted BTC Price: 
 [25985.70274713]
```

Here, we are able to see the R2 score of the individual columns comapred to the dependent column BTC. SPY has the higest R2 score and real estate has the lowest R2 score.

In [46]:

```
from sklearn.metrics import r2_score
y_pred = regr.predict(X)
r2_score(Y, y_pred)

for column in data3.columns:
    if column != "BTC":
        X1 = data3[[column]]
        regr1 = linear_model.LinearRegression()
        regr1.fit(X1, Y)
        y_pred1 = regr1.predict(X1)
        print(column, r2_score(Y, y_pred1))
```

```
DXY  0.24528135804570983
SPY  0.6472267090540356
10 year note 0.11212343603043151
Oil (XOP Equity) 0.2136180826414681
VXX 0.204021812528988
Gold (GLD Equity) 0.41513927212748725
Real Eatate (VNQ Equity) 0.012522623397841603
Bonds (AGG Equity) 0.2128084743114531
```

Returns over last 5 years per security

In [47]:

```
cum_return = (data3.iloc[-1] - data3.iloc[0]) / data3.iloc[0]
cum_return*100
```

Out[47]:

```
DXY                              -8.685361
SPY                             113.699144
BTC                           12254.386812
10 year note                    -40.186374
Oil (XOP Equity)                -33.182417
VXX                             -95.552817
Gold (GLD Equity)                64.709884
Real Eatate (VNQ Equity)         13.825060
Bonds (AGG Equity)                7.089932
dtype: float64
```

The multiple linear regression model allows you to price bitcoin using the wider market sectors, herein, allowing you to price it given a comprehensive macro-backdrop and expectations.

Given Bitcoins strong correlation with equities, it may not serve as a portfolio diversifier for those nearing retirement and aiming for cash flow. However, given bitcoins massive over performance over equities over the last 5 years, it may make sense for individuals to replace 1 - 5% of equity holdings with BTC.

For those less risk adverse and understands the cyclical nature of BTC, it may serve as a core component to ones portfolio.