

## hw\_2\_Anirudh\_Tunoori

January 17, 2022

```
[ ]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import requests

%matplotlib inline
```

```
[ ]: def get_data(token):
    if token == 'FLOW':
        exchange = 'Kraken'
    else:
        exchange = 'coinbase-pro'
    res = requests.get(
        f'https://api.cryptowat.ch/markets/{exchange}/{token}usd/ohlc',
        params={
            'periods': '3600',
            'after': str(int(pd.Timestamp('2021-11-22').timestamp()))
        }
    )

    df = pd.DataFrame(
        res.json()['result']['3600'],
        columns=['ts', 'open', 'high', 'low', 'close', 'volume', 'volumeUSD']
    )
    df['ts'] = pd.to_datetime(df.ts, unit='s')
    df['token'] = token

    return df
```

```
[ ]: tokens = ['ETH', 'SOL', 'AVAX', 'USDT', 'FLOW']
df_all = pd.concat(get_data(token) for token in tokens)
flattened_df = df_all[['ts', 'token', 'close']].copy()
```

```
[ ]: volume_traded = df_all.groupby('token')['volumeUSD'].sum().
    ↪sort_values(ascending=False).to_frame()
df_all = df_all.rename({'volume': 'volumeBase', 'volumeUSD': 'volumeTerm'},
    ↪axis=1, inplace=True)
```

```
volume_traded
```

```
[ ]: flattened_df = flattened_df.pivot(index='ts',columns='token',values='close').  
    ↪reset_index().rename_axis(None, axis=1)  
    flattened_df['ETH_SOL_Ratio'] = flattened_df['ETH'] / flattened_df['SOL']  
    flattened_df
```

```
[ ]: log_return = flattened_df.apply(lambda x: np.log(x).diff() if x.name in tokens_  
    ↪else x, axis=0)  
    log_return
```

```
[ ]: correlation = log_return.drop(columns=['ETH_SOL_Ratio']).corr()  
    correlation
```

```
[ ]: # I tried two different visualizations for these correlations  
    import seaborn as sns  
    from pandas.plotting import scatter_matrix  
  
    fig = plt.figure(figsize=(5,5))  
    sns.heatmap(correlation, annot = True, fmt = '.2f')  
    plt.show()  
  
    x = scatter_matrix(log_return.drop(columns=['ETH_SOL_Ratio']), figsize=(10,  
    ↪10), diagonal='kde')
```