# 2_Simulations

June 20, 2022

## 0.1 With the data ready, portfolios are simulated

[2]:
```
import bt
import yfinance as yf
import pandas as pd
import numpy as np
```

[24]:
```
monthly_returns = pd.read_excel('monthly_returns.
 ↪xlsx',parse_dates=True,index_col=0)
price_index = pd.read_excel('price_index.xlsx',parse_dates=True,index_col=0)
```

### 0.1.1 A function is created to simulate portfolios

[19]:
```
def generate_random_portfolios(number_of_portfolios=5,number_of_companies=20):

    random_portfolios = pd.DataFrame(index=monthly_returns.index)

    for portfolio in range(0,number_of_portfolios):

        s = bt.Strategy('s1', [bt.algos.RunMonthly(),
                       bt.algos.SelectRandomly(number_of_companies),
                       bt.algos.WeighEqually(),
                       bt.algos.Rebalance()])

        test = bt.Backtest(s, price_index,initial_capital=10000000)
        res = bt.run(test)

        random_portfolios[portfolio] = res.prices

    return random_portfolios
```

### 0.1.2 Ten thousand portfolios are simulated

**This computations is computationally expensive, and may take more than 12 hours**

[27]:
```
number_of_portfolios = 10000

results = generate_random_portfolios(number_of_portfolios=number_of_portfolios)
```

### 0.1.3 S&P 500 index data is added for reference

[ ]:
```
spy = yf.download("SPY")["Adj Close"].pct_change()

spy = spy["1996":]

spy = spy["1996":].resample("M").agg(lambda x: (x + 1).prod() - 1)

spy[0] = 0
spy[1] = 0

results["SPX"] = (1+spy).cumprod()*100

results.to_excel("Simulations.xlsx")
```

### 0.1.4 Data is saved for future access

[44]:
```
results = pd.read_excel("Simulations.xlsx",index_col=0)
```