PortfolioMPT2

May 2, 2021

1 Universidad Nacional de Ingeniería

1.1 Facultad de Ingeniería Económica, Estadística y CC.SS

1.1.1 Finanzas Corporativas II - 2021-I

Teoría de Portafolio Docente: Alfonso Chang Medina MSc \ Fecha: Abril~Mayo 2021 \ e-mail: achangm@uni.edu.pe

Trabajaremos portafolios de **9 activos**

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

```
[10]: monthly_returns = pd.read_csv("datamensual2.csv", index_col ="Date")
      monthly_returns
```

```
[10]:            MSFT.Adjusted  SHI.Adjusted  GE.Adjusted  TSN.Adjusted  \
      Date
      01/02/2015       0.081945      0.173933     0.084252      0.056518
      01/03/2015      -0.068438      0.094373    -0.037315     -0.073242
      01/04/2015       0.179201      0.482224     0.087549      0.030850
      01/05/2015      -0.037282     -0.207830     0.006992      0.072026
      01/06/2015      -0.053094      0.092176    -0.026004      0.006574
      01/07/2015       0.056151     -0.286012    -0.009378      0.039554
      01/08/2015      -0.070524     -0.111036    -0.050285     -0.047798
      01/09/2015       0.023434      0.067573     0.015988      0.021573
      01/10/2015       0.173395      0.079168     0.145806      0.028815
      01/11/2015       0.031968     -0.140132     0.034662      0.119685
      01/12/2015       0.027296      0.083184     0.039619      0.067490
      01/01/2016      -0.007054      0.013797    -0.060624      0.000562
      01/02/2016      -0.079498      0.064398     0.001374      0.193472
      01/03/2016       0.089191      0.163616     0.095014      0.031330
      01/04/2016      -0.102087     -0.034709    -0.033262     -0.012681
      01/05/2016       0.060872      0.009808    -0.017055     -0.031484
      01/06/2016      -0.028168     -0.069676     0.040517      0.048420
      01/07/2016       0.102268      0.101159    -0.003312      0.097091
      01/08/2016       0.013668      0.041531     0.003206      0.026414
```

```
01/09/2016       0.008647      0.000782    -0.053250     -0.010006
01/10/2016       0.039488     -0.001760    -0.009937     -0.052513
01/11/2016       0.005658      0.033118     0.055477     -0.220853
01/12/2016       0.037454      0.024877     0.026942      0.086111
01/01/2017       0.039598      0.141215    -0.054512      0.017836
01/02/2017      -0.010417     -0.023042     0.003697     -0.003670
01/03/2017       0.035005     -0.088505     0.007609     -0.010084
01/04/2017       0.038718      0.003938    -0.027557      0.040492
01/05/2017       0.019957      0.051865    -0.057138     -0.113939
01/06/2017      -0.007400     -0.096991    -0.013606      0.092110
01/07/2017       0.053250      0.133872    -0.044824      0.011588
01/08/2017       0.028075      0.090410    -0.042271     -0.000947
01/09/2017       0.001562     -0.047021    -0.015186      0.110643
01/10/2017       0.110342      0.011303    -0.171884      0.034323
01/11/2017       0.011832     -0.020540    -0.097346      0.120538
01/12/2017       0.021155     -0.039052    -0.047015     -0.010786
01/01/2018       0.104998      0.080528    -0.069301     -0.063133
01/02/2018      -0.013137     -0.035253    -0.136274     -0.022993
01/03/2018      -0.022336      0.027780    -0.037435     -0.012097
01/04/2018       0.024353      0.073852     0.042838     -0.043136
01/05/2018       0.055327      0.113567     0.000711     -0.038240
01/06/2018       0.001995     -0.209513    -0.033951      0.024510
01/07/2018       0.073021      0.090541     0.010305     -0.177540
01/08/2018       0.057247     -0.044674    -0.051950      0.085724
01/09/2018       0.021839      0.028403    -0.136406     -0.048866
01/10/2018      -0.068387     -0.319386    -0.101896      0.006530
01/11/2018       0.037487      0.043073    -0.297633     -0.016321
01/12/2018      -0.083480     -0.058933     0.009290     -0.092277
01/01/2019       0.027769      0.102948     0.295640      0.148032
01/02/2019       0.070250      0.052327     0.061606     -0.004208
01/03/2019       0.055671     -0.054629    -0.039259      0.124785
01/04/2019       0.101963     -0.061568     0.018917      0.077302
01/05/2019      -0.054442     -0.049559    -0.074486      0.011664
01/06/2019       0.083538     -0.066804     0.106419      0.066877
01/07/2019       0.017097     -0.045237    -0.003811     -0.015477
01/08/2019       0.011600     -0.193238    -0.236389      0.157273
01/09/2019       0.011775      0.012743     0.080322     -0.072907
01/10/2019       0.030739     -0.066868     0.111128     -0.039667
01/11/2019       0.054365     -0.001465     0.121561      0.082280
01/12/2019       0.044298      0.112844    -0.009808      0.017405

            AAPL.Adjusted  SBUX.Adjusted  GOOG.Adjusted  AMZN.Adjusted  \
Date
01/02/2015       0.092077       0.065873       0.043706       0.069799
01/03/2015      -0.027936       0.016503      -0.018800      -0.021430
01/04/2015       0.005770       0.046021      -0.016902       0.125321
01/05/2015       0.040173       0.046887      -0.009781       0.017509
```

01/06/2015	−0.033770	0.034625	−0.022041	0.011259
01/07/2015	−0.033481	0.077313	0.183918	0.211162
01/08/2015	−0.073005	−0.057189	−0.011834	−0.044353
01/09/2015	−0.017542	0.040947	−0.016028	−0.001952
01/10/2015	0.080113	0.096046	0.155540	0.201081
01/11/2015	−0.010093	−0.019039	0.043752	0.060296
01/12/2015	−0.112519	−0.019170	0.021686	0.016544
01/01/2016	−0.078224	0.012252	−0.021215	−0.141006
01/02/2016	−0.006700	−0.043039	−0.062739	−0.060535
01/03/2016	0.125158	0.028538	0.065428	0.071783
01/04/2016	−0.150731	−0.059882	−0.072273	0.105345
01/05/2016	0.063244	−0.024119	0.059805	0.091500
01/06/2016	−0.037526	0.043316	−0.061119	−0.009969
01/07/2016	0.086235	0.016151	0.105087	0.058602
01/08/2016	0.017974	−0.031855	−0.002266	0.013548
01/09/2016	0.068851	−0.034400	0.013262	0.084895
01/10/2016	0.004325	−0.019962	0.009284	−0.058389
01/11/2016	−0.026959	0.088314	−0.034361	−0.050972
01/12/2016	0.051962	−0.038561	0.018015	−0.000933
01/01/2017	0.046642	−0.005418	0.031840	0.093639
01/02/2017	0.121229	0.029443	0.032620	0.025845
01/03/2017	0.051868	0.030873	0.007684	0.047942
01/04/2017	−0.000069	0.028199	0.088100	0.042457
01/05/2017	0.061488	0.057427	0.062988	0.072578
01/06/2017	−0.054797	−0.082886	−0.059935	−0.027129
01/07/2017	0.032180	−0.077160	0.023674	0.020228
01/08/2017	0.097734	0.016171	0.009445	−0.007295
01/09/2017	−0.058216	−0.016681	0.020839	−0.019826
01/10/2017	0.092404	0.020821	0.058253	0.139515
01/11/2017	0.016487	0.052915	0.004681	0.062658
01/12/2017	−0.011775	−0.001484	0.024172	−0.006206
01/01/2018	−0.010693	−0.010854	0.111597	0.215627
01/02/2018	0.061892	0.005092	−0.057351	0.041554
01/03/2018	−0.055735	0.019150	−0.068306	−0.044003
01/04/2018	−0.015134	−0.005543	−0.014114	0.078880
01/05/2018	0.122893	−0.015757	0.064389	0.039739
01/06/2018	−0.005614	−0.143275	0.027866	0.042164
01/07/2018	0.027599	0.069962	0.087165	0.044663
01/08/2018	0.179172	0.020031	0.000764	0.124308
01/09/2018	−0.004837	0.068431	−0.020501	−0.004836
01/10/2018	−0.030952	0.024847	−0.102899	−0.225887
01/11/2018	−0.203396	0.135417	0.016268	0.056070
01/12/2018	−0.120605	−0.030036	−0.055243	−0.118052
01/01/2019	0.053687	0.056451	0.075092	0.134808
01/02/2019	0.039523	0.030638	0.003175	−0.046993
01/03/2019	0.096882	0.061652	0.046572	0.082442
01/04/2019	0.054901	0.043949	0.012846	0.078681

```
01/05/2019      -0.136476      -0.021075      -0.074070      -0.081875
01/06/2019       0.126520       0.101897      -0.020801       0.064656
01/07/2019       0.073617       0.121817       0.118323      -0.014281
01/08/2019      -0.020390       0.019556      -0.023771      -0.049688
01/09/2019       0.074215      -0.084283       0.025675      -0.022995
01/10/2019       0.104976      -0.044637       0.033168       0.023203
01/11/2019       0.071696       0.010236       0.034973       0.013496
01/12/2019       0.097202       0.033716       0.024271       0.025786

            INTC.Adjusted
Date
01/02/2015       0.006336
01/03/2015      -0.054249
01/04/2015       0.040118
01/05/2015       0.057022
01/06/2015      -0.117494
01/07/2015      -0.049530
01/08/2015      -0.014263
01/09/2015       0.062820
01/10/2015       0.116381
01/11/2015       0.026521
01/12/2015      -0.002227
01/01/2016      -0.104877
01/02/2016      -0.047196
01/03/2016       0.097941
01/04/2016      -0.066127
01/05/2016       0.042353
01/06/2016       0.046188
01/07/2016       0.060912
01/08/2016       0.029119
01/09/2016       0.058078
01/10/2016      -0.079359
01/11/2016      -0.004887
01/12/2016       0.051794
01/01/2017       0.015050
01/02/2017      -0.016982
01/03/2017       0.003516
01/04/2017       0.002216
01/05/2017      -0.001107
01/06/2017      -0.060474
01/07/2017       0.050003
01/08/2017      -0.011341
01/09/2017       0.089822
01/10/2017       0.177803
01/11/2017      -0.014392
01/12/2017       0.034922
01/01/2018       0.042000
```

```
01/02/2018       0.023608
01/03/2018       0.061821
01/04/2018      -0.008872
01/05/2018       0.067054
01/06/2018      -0.099002
01/07/2018      -0.032924
01/08/2018       0.006837
01/09/2018      -0.017758
01/10/2018      -0.008708
01/11/2018       0.050536
01/12/2018      -0.043196
01/01/2019       0.004040
01/02/2019       0.116839
01/03/2019       0.020195
01/04/2019      -0.050803
01/05/2019      -0.147511
01/06/2019       0.089496
01/07/2019       0.054474
01/08/2019      -0.064130
01/09/2019       0.090060
01/10/2019       0.092607
01/11/2019       0.026533
01/12/2019       0.036025
```

[11]:
```
mean_ret = monthly_returns.mean()
mean_ret
```

[11]:
```
MSFT.Adjusted    0.024910
SHI.Adjusted     0.005483
GE.Adjusted     -0.010049
TSN.Adjusted     0.015619
AAPL.Adjusted    0.017009
SBUX.Adjusted    0.013308
GOOG.Adjusted    0.015586
AMZN.Adjusted    0.027983
INTC.Adjusted    0.012434
dtype: float64
```

[23]:
```
#Generating a random matrix of 1000 rows and 4 Columns
matrix = np.random.rand(10000,9)
matrix
```

[23]:
```
array([[0.66244665, 0.47084056, 0.01204378, …, 0.69591026, 0.29215704,
        0.57123252],
       [0.48791289, 0.93155626, 0.69697585, …, 0.84859146, 0.40880258,
        0.82947397],
       [0.92815137, 0.29704817, 0.34691342, …, 0.25894406, 0.77926552,
        0.69997417],
       …,
       [0.25188974, 0.97440287, 0.55736673, …, 0.32230323, 0.61938185,
        0.49692839],
       [0.65751507, 0.11258587, 0.12762429, …, 0.09070256, 0.45763595,
        0.73843696],
       [0.15384535, 0.35879575, 0.05725661, …, 0.73069565, 0.84953198,
        0.13358892]])
```

[24]:
```
#Converting to a data frame
matrix_df = pd.DataFrame(matrix, columns = monthly_returns.columns)
matrix_df
```

[24]:
```
      MSFT.Adjusted  SHI.Adjusted  GE.Adjusted  TSN.Adjusted  AAPL.Adjusted  \
0          0.662447      0.470841     0.012044      0.648247       0.223985
1          0.487913      0.931556     0.696976      0.675902       0.924570
2          0.928151      0.297048     0.346913      0.422725       0.250261
3          0.115060      0.610184     0.208176      0.915334       0.421385
4          0.410319      0.109573     0.584129      0.428427       0.959078
…               …             …           …             …              …
9995       0.372411      0.084083     0.865655      0.652734       0.188740
9996       0.823967      0.748355     0.585127      0.133756       0.433542
9997       0.251890      0.974403     0.557367      0.632622       0.631842
9998       0.657515      0.112586     0.127624      0.425089       0.685080
9999       0.153845      0.358796     0.057257      0.601393       0.644283

      SBUX.Adjusted  GOOG.Adjusted  AMZN.Adjusted  INTC.Adjusted
0          0.019888       0.695910       0.292157       0.571233
1          0.600131       0.848591       0.408803       0.829474
2          0.822432       0.258944       0.779266       0.699974
3          0.621025       0.656126       0.667715       0.921975
4          0.167152       0.808776       0.493998       0.013355
…               …              …              …              …
9995       0.634684       0.533530       0.463360       0.190192
9996       0.149374       0.152657       0.544390       0.535347
9997       0.388706       0.322303       0.619382       0.496928
9998       0.781476       0.090703       0.457636       0.738437
9999       0.463285       0.730696       0.849532       0.133589

[10000 rows x 9 columns]
```

[25]:
```
matrix_sum = matrix_df.sum(axis = 1)
matrix_sum
```

[25]:
```
0       3.596750
1       6.403915
2       4.805715
```

```
3       5.136980
4       3.974805
          …
9995    3.985391
9996    4.106515
9997    4.875443
9998    4.076146
9999    3.992675
Length: 10000, dtype: float64
```

[26]:
```python
#Calculating portfolio weights
weights  = matrix_df.divide(matrix_sum , axis ="rows")
weights
```

[26]:
```
      MSFT.Adjusted  SHI.Adjusted  GE.Adjusted  TSN.Adjusted  AAPL.Adjusted  \
0          0.184179      0.130907     0.003349      0.180231       0.062274
1          0.076190      0.145467     0.108836      0.105545       0.144376
2          0.193135      0.061811     0.072188      0.087963       0.052076
3          0.022398      0.118783     0.040525      0.178185       0.082030
4          0.103230      0.027567     0.146958      0.107786       0.241289
…             …            …           …            …             …
9995       0.093444      0.021098     0.217207      0.163782       0.047358
9996       0.200649      0.182236     0.142487      0.032572       0.105574
9997       0.051665      0.199859     0.114321      0.129757       0.129597
9998       0.161308      0.027621     0.031310      0.104287       0.168070
9999       0.038532      0.089863     0.014340      0.150624       0.161366

      SBUX.Adjusted  GOOG.Adjusted  AMZN.Adjusted  INTC.Adjusted
0          0.005529       0.193483       0.081228       0.158819
1          0.093713       0.132511       0.063836       0.129526
2          0.171136       0.053883       0.162154       0.145655
3          0.120893       0.127726       0.129982       0.179478
4          0.042053       0.203476       0.124282       0.003360
…             …             …              …              …
9995       0.159253       0.133872       0.116265       0.047722
9996       0.036375       0.037174       0.132567       0.130365
9997       0.079727       0.066107       0.127041       0.101925
9998       0.191719       0.022252       0.112272       0.181161
9999       0.116034       0.183009       0.212773       0.033459

[10000 rows x 9 columns]
```

[27]:
```python
#transpose
weights_t= np.transpose(weights)
weights_t
```

[27]:

```
                      0         1         2         3         4         5  \
MSFT.Adjusted  0.184179  0.076190  0.193135  0.022398  0.103230  0.179987
SHI.Adjusted   0.130907  0.145467  0.061811  0.118783  0.027567  0.216021
GE.Adjusted    0.003349  0.108836  0.072188  0.040525  0.146958  0.078620
TSN.Adjusted   0.180231  0.105545  0.087963  0.178185  0.107786  0.123225
AAPL.Adjusted  0.062274  0.144376  0.052076  0.082030  0.241289  0.091216
SBUX.Adjusted  0.005529  0.093713  0.171136  0.120893  0.042053  0.003676
GOOG.Adjusted  0.193483  0.132511  0.053883  0.127726  0.203476  0.096014
AMZN.Adjusted  0.081228  0.063836  0.162154  0.129982  0.124282  0.069287
INTC.Adjusted  0.158819  0.129526  0.145655  0.179478  0.003360  0.141954

                      6         7         8         9  ...      9990  \
MSFT.Adjusted  0.156778  0.119157  0.010010  0.200463  ...  0.206322
SHI.Adjusted   0.218505  0.078206  0.183353  0.107508  ...  0.183210
GE.Adjusted    0.069647  0.126117  0.083461  0.056086  ...  0.027102
TSN.Adjusted   0.198950  0.139205  0.192547  0.164354  ...  0.050657
AAPL.Adjusted  0.043730  0.123172  0.215147  0.204864  ...  0.050052
SBUX.Adjusted  0.095270  0.141240  0.090284  0.066636  ...  0.100597
GOOG.Adjusted  0.155758  0.031365  0.027003  0.018608  ...  0.202124
AMZN.Adjusted  0.054522  0.062625  0.001554  0.020949  ...  0.110098
INTC.Adjusted  0.006839  0.178914  0.196641  0.160532  ...  0.069839

                   9991      9992      9993      9994      9995      9996  \
MSFT.Adjusted  0.070083  0.182354  0.119498  0.098676  0.093444  0.200649
SHI.Adjusted   0.180915  0.030596  0.146527  0.144895  0.021098  0.182236
GE.Adjusted    0.210157  0.023147  0.160173  0.147268  0.217207  0.142487
TSN.Adjusted   0.067367  0.081104  0.048764  0.006121  0.163782  0.032572
AAPL.Adjusted  0.112282  0.165674  0.067838  0.040489  0.047358  0.105574
SBUX.Adjusted  0.088335  0.190897  0.147873  0.140657  0.159253  0.036375
GOOG.Adjusted  0.193191  0.212322  0.151794  0.164883  0.133872  0.037174
AMZN.Adjusted  0.034006  0.069888  0.108468  0.157878  0.116265  0.132567
INTC.Adjusted  0.043664  0.044018  0.049066  0.099131  0.047722  0.130365

                   9997      9998      9999
MSFT.Adjusted  0.051665  0.161308  0.038532
SHI.Adjusted   0.199859  0.027621  0.089863
GE.Adjusted    0.114321  0.031310  0.014340
TSN.Adjusted   0.129757  0.104287  0.150624
AAPL.Adjusted  0.129597  0.168070  0.161366
SBUX.Adjusted  0.079727  0.191719  0.116034
GOOG.Adjusted  0.066107  0.022252  0.183009
AMZN.Adjusted  0.127041  0.112272  0.212773
INTC.Adjusted  0.101925  0.181161  0.033459

[9 rows x 10000 columns]
```

[28]:
```
#Using the portfolio return formula
portfolio_return = np.dot(weights, mean_ret)
portfolio_return
```

[28]:
```
array([0.01648339, 0.01241528, 0.01615012, …, 0.01237861, 0.01663511,
       0.01717225])
```

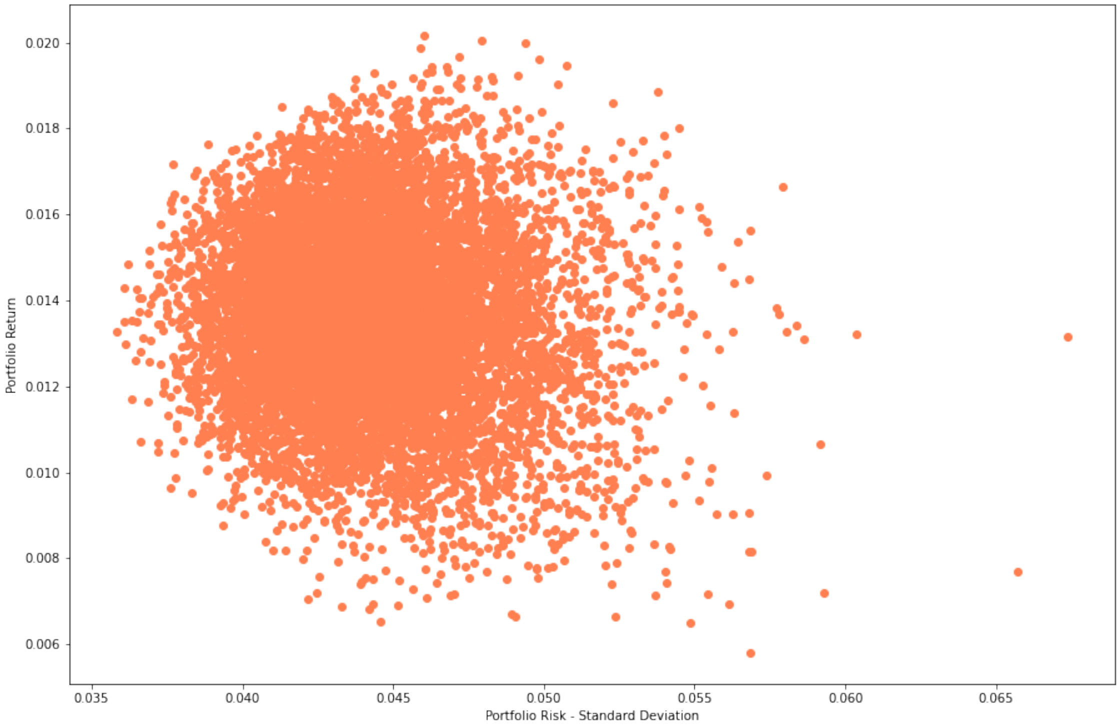
[29]:
```
#Variance covariance
cov_mat = monthly_returns.cov()
cov_mat
```

[29]:
```
               MSFT.Adjusted  SHI.Adjusted  GE.Adjusted  TSN.Adjusted  \
MSFT.Adjusted       0.003257      0.003088     0.001250      0.000395
SHI.Adjusted        0.003088      0.014616     0.002869     -0.001295
GE.Adjusted         0.001250      0.002869     0.007941      0.000547
TSN.Adjusted        0.000395     -0.001295     0.000547      0.005739
AAPL.Adjusted       0.002257      0.001566     0.002369      0.000707
SBUX.Adjusted       0.001023      0.000821     0.000004     -0.000204
GOOG.Adjusted       0.002078      0.000867     0.001188      0.000024
AMZN.Adjusted       0.002856      0.002561     0.001286      0.000782
INTC.Adjusted       0.001824      0.001699     0.001118      0.000194

               AAPL.Adjusted  SBUX.Adjusted  GOOG.Adjusted  AMZN.Adjusted  \
MSFT.Adjusted       0.002257       0.001023       0.002078       0.002856
SHI.Adjusted        0.001566       0.000821       0.000867       0.002561
GE.Adjusted         0.002369       0.000004       0.001188       0.001286
TSN.Adjusted        0.000707      -0.000204       0.000024       0.000782
AAPL.Adjusted       0.005805       0.000545       0.001792       0.002147
SBUX.Adjusted       0.000545       0.002784       0.000805       0.000885
GOOG.Adjusted       0.001792       0.000805       0.003201       0.002966
AMZN.Adjusted       0.002147       0.000885       0.002966       0.006366
INTC.Adjusted       0.002220       0.000635       0.001197       0.001556

               INTC.Adjusted
MSFT.Adjusted       0.001824
SHI.Adjusted        0.001699
GE.Adjusted         0.001118
TSN.Adjusted        0.000194
AAPL.Adjusted       0.002220
SBUX.Adjusted       0.000635
GOOG.Adjusted       0.001197
AMZN.Adjusted       0.001556
INTC.Adjusted       0.003931
```

[30]:
```
portfolio_risk = []
for one_port in range(weights.shape[0]):
    risk = np.sqrt(np.dot(weights.iloc[one_port,:],np.dot(cov_mat,weights_t.
↪iloc[:,one_port])))

    portfolio_risk.append(risk)
```

[37]:

```
plt.figure(figsize = (15,10))
plt.scatter(portfolio_risk, portfolio_return, c="coral")
plt.xlabel("Portfolio Risk - Standard Deviation")
plt.ylabel("Portfolio Return")
plt.show()
```



[36]:

```
#converting to a csv file
portfolio_risk = pd.DataFrame(portfolio_risk, columns = ["portfolio risk"])
portfolio_return = pd.DataFrame(portfolio_return, columns = ["portfolio␣
↪return"])
random_portfolio = pd.concat([portfolio_return, portfolio_risk, weights], axis␣
↪=1)
random_portfolio.to_csv("Random_Portfolios2.csv")
```