



INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

# Estimating Tail Risk

Dakota Wixom

Quantitative Analyst | [QuantCourse.com](https://QuantCourse.com)



# Estimating Tail Risk

**Tail risk** is the risk of extreme investment outcomes, most notably on the negative side of a distribution.

- Historical Drawdown
- Value at Risk
- Conditional Value at Risk
- Monte-Carlo Simulation

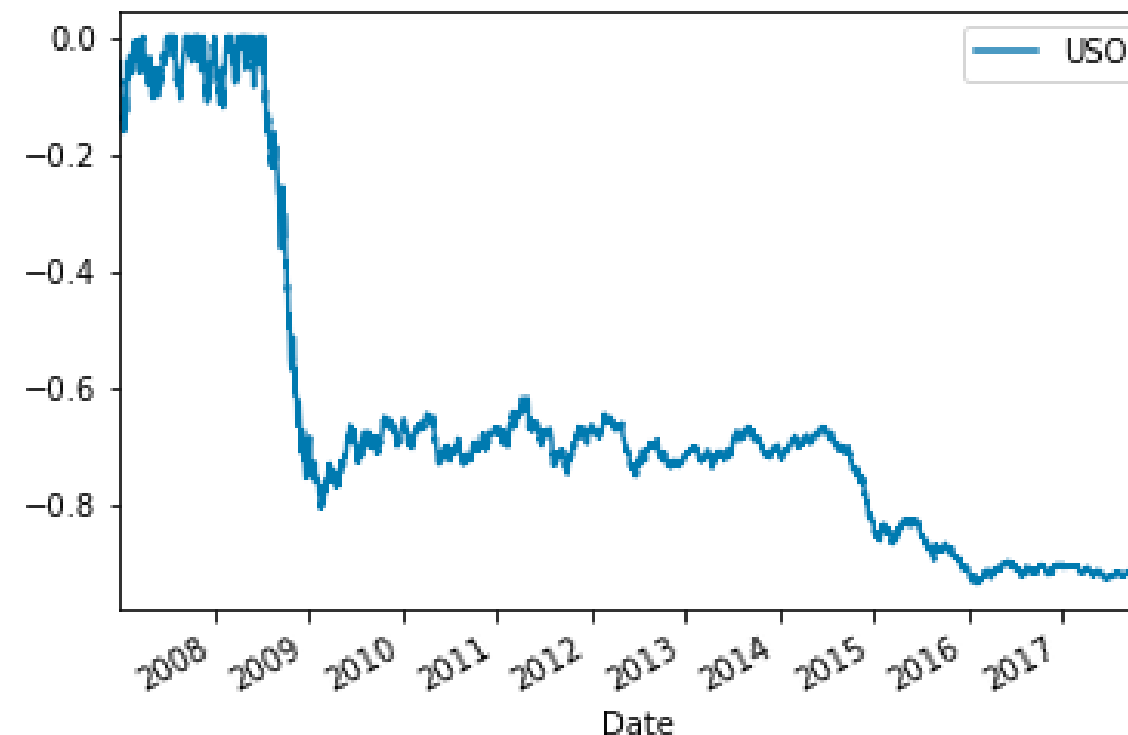
# Historical Drawdown

**Drawdown** is the percentage loss from the highest cumulative historical point.

$$\text{Drawdown} = \frac{r_t}{RM} - 1$$

- $r_t$ : Cumulative return at time  $t$
- $RM$ : Running maximum

## HISTORICAL DRAWDOWN OF THE USO OIL ETF





# Historical Drawdown in Python

Assuming `cum_rets` is an `np.array` of cumulative returns over time

```
In [1]: running_max = np.maximum.accumulate(cum_rets)
In [2]: running_max[running_max < 1] = 1
In [3]: drawdown = (cum_rets)/running_max - 1
In [4]: drawdown
Out [4]:
```

Date	Return
2007-01-03	-0.042636
2007-01-04	-0.081589
2007-01-05	-0.073062



# Historical Value at Risk

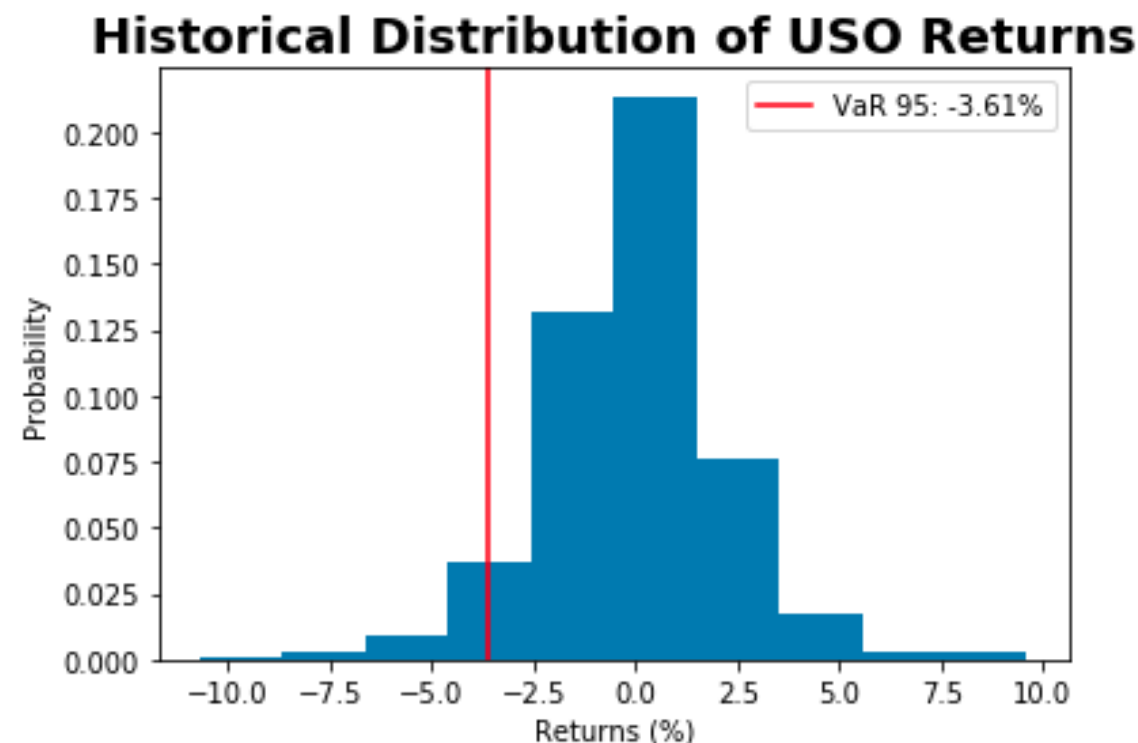
**Value at Risk**, or VaR, is a threshold with a given confidence level that losses will not (or more accurately, will not historically) exceed a certain level.

VaR is commonly quoted with quantiles such as 95, 99, and 99.9.

Example:

$$\text{VaR}(95) = -2.3\%$$

95% certain that **losses will not exceed** -2.3% in a given day based on historical values.





# Historical Value at Risk in Python

```
In [1]: var_level = 95
In [2]: var_95 = np.percentile(StockReturns, 100 - var_level)
In [3]: var_95
Out [3]: -.023
```



# Historical Expected Shortfall

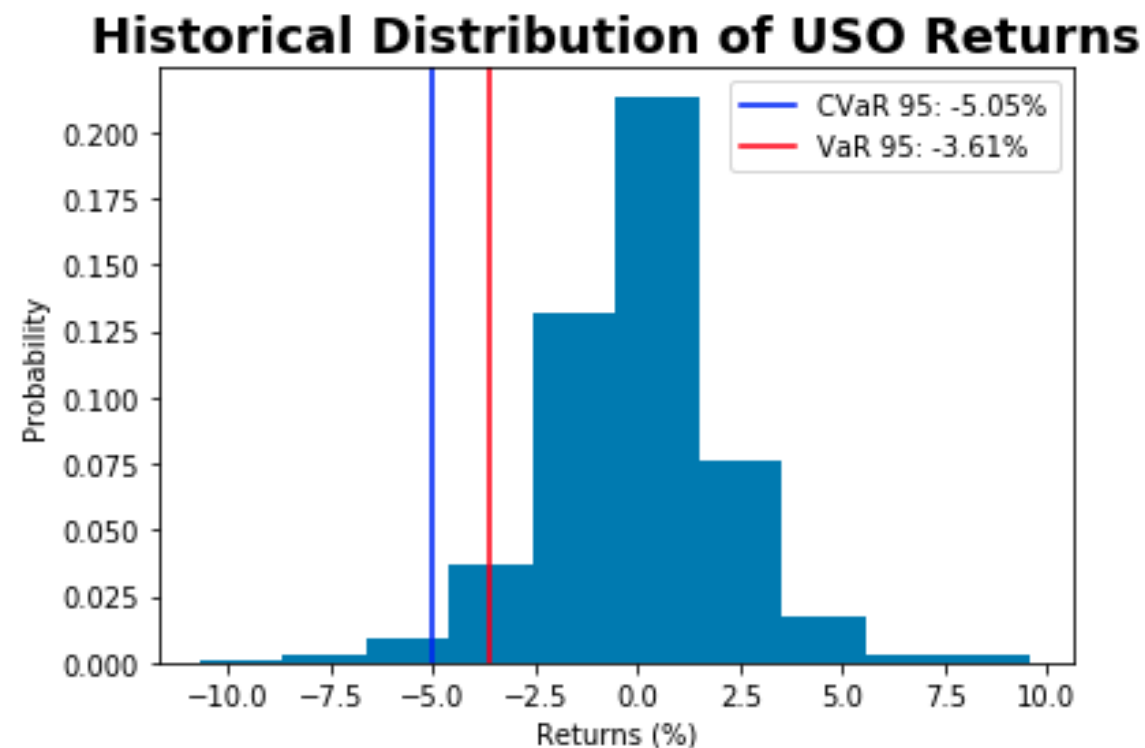
**Conditional Value at Risk**, or CVaR, is an estimate of expected losses sustained in the worst 1 - x% of scenarios.

CVaR is commonly quoted with quantiles such as 95, 99, and 99.9.

Example:

$$\text{CVaR}(95) = -2.5\%$$

In the worst 5% of cases, **losses were on average exceed -2.5%** historically.





# Historical Expected Shortfall in Python

Assuming you have an object `StockReturns` which is a time series of stock returns.

To calculate historical CVaR(95):

```
In [1]: var_level = 95
In [2]: var_95 = np.percentile(StockReturns, 100 - var_level)
In [3]: cvar_95 = StockReturns[StockReturns <= var_95].mean()
In [3]: cvar_95
Out [3]: -.025
```





## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

**Let's practice!**



## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

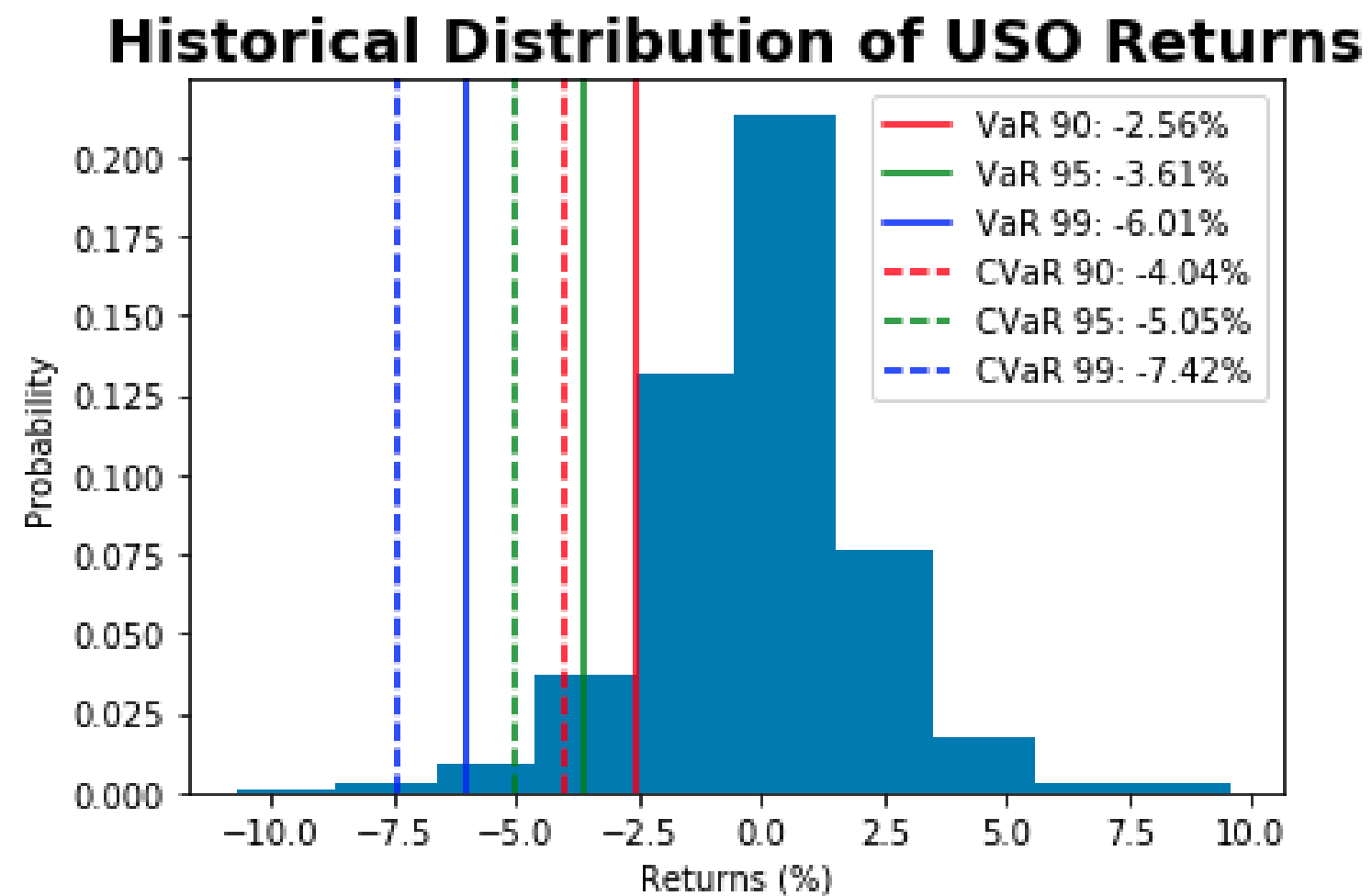
# VaR Extensions

Dakota Wixom

Quantitative Analyst | [QuantCourse.com](https://QuantCourse.com)



# VaR Quantiles





# Empirical Assumptions

Empirical Historical values are those that have *actually occurred*.

How do you simulate the probability of a value that has never occurred historically before?

**Sample from a probability distribution**



# Parametric VaR in Python

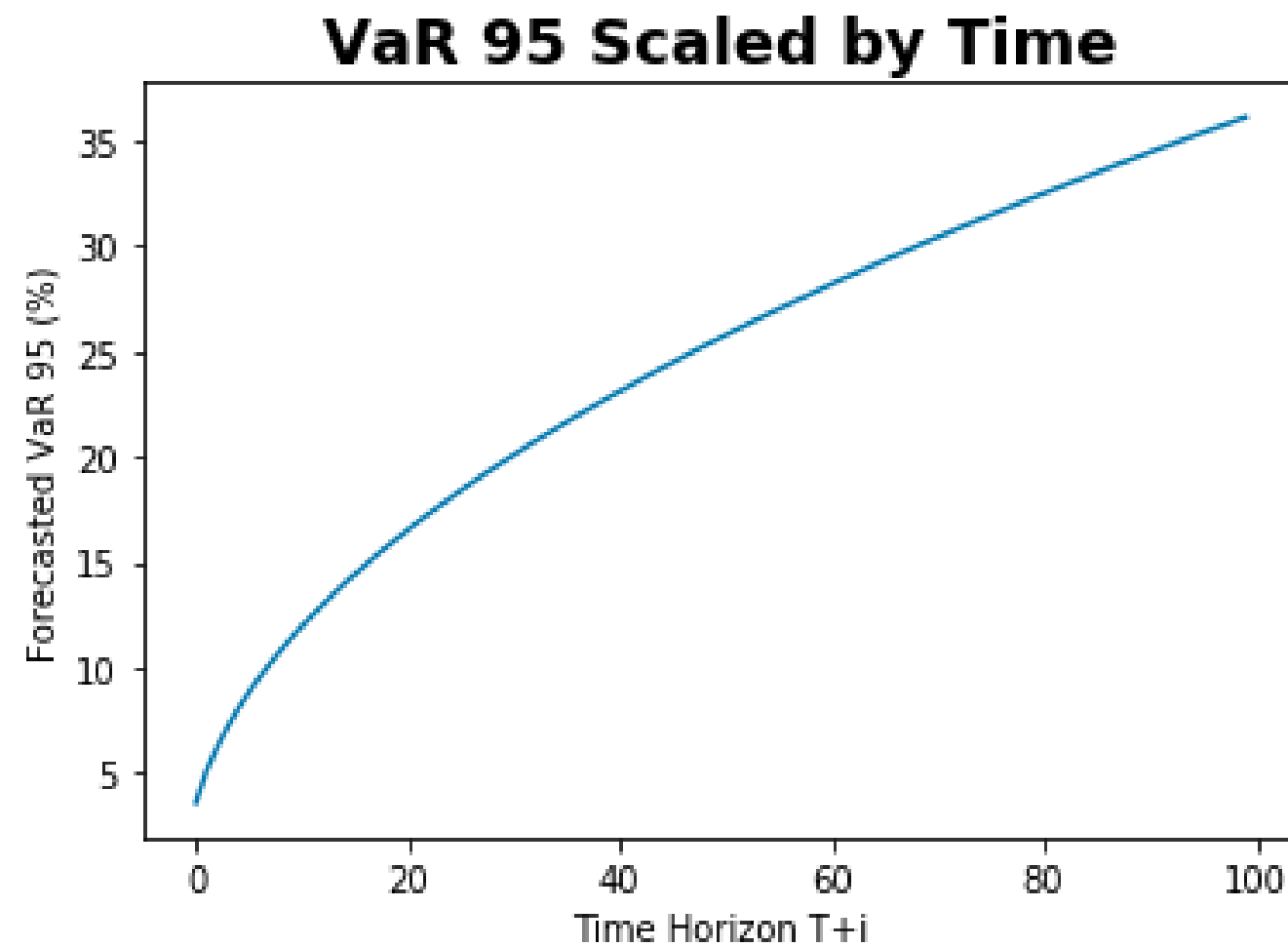
Assuming you have an object `StockReturns` which is a time series of stock returns.

To calculate parametric VaR(95):

```
In [1]: mu = np.mean(StockReturns)
In [2]: std = np.std(StockReturns)
In [3]: confidence_level = 0.05
In [4]: VaR = norm.ppf(confidence_level, mu, std)
In [5]: VaR
Out [5]: -0.0235
```



# Scaling Risk





# Scaling Risk in Python

Assuming you have a one-day estimate of VaR(95) `var_95`.

To estimate 5-day VaR(95):

```
In [1]: forecast_days = 5
In [2]: forecast_var95_5day = var_95*np.sqrt(forecast_days)
In [3]: forecast_var95_5day
Out [3]: -0.0525
```



## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

**Let's practice!**





## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

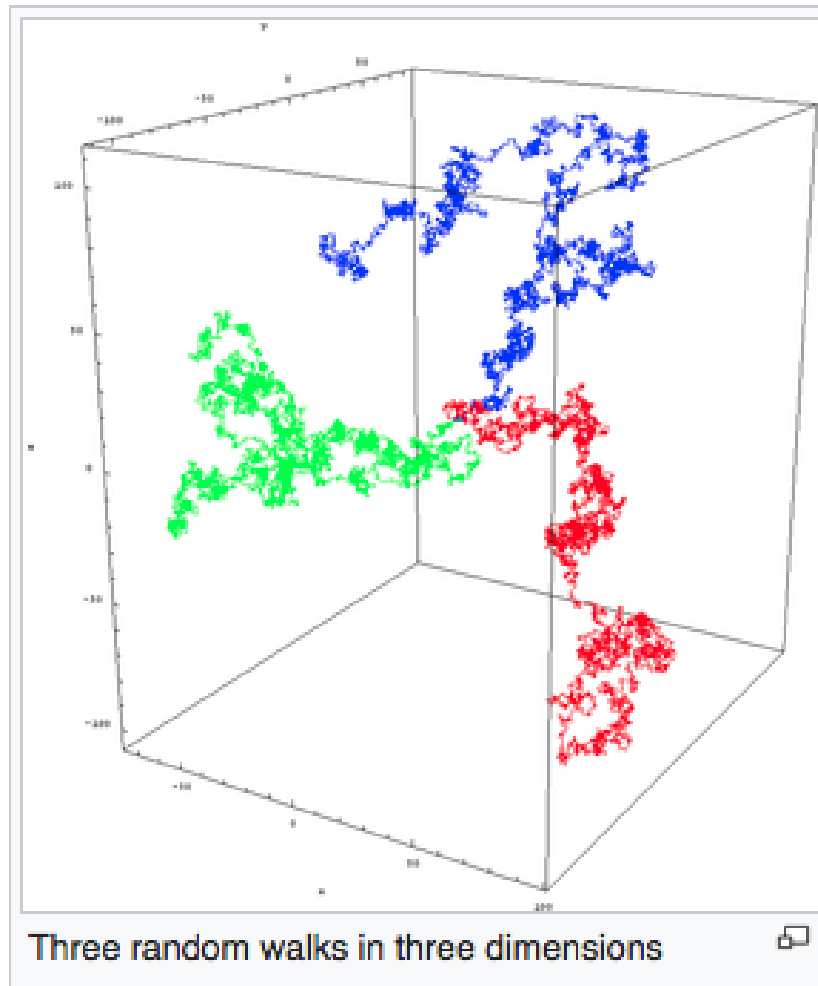
# Random Walks

Dakota Wixom

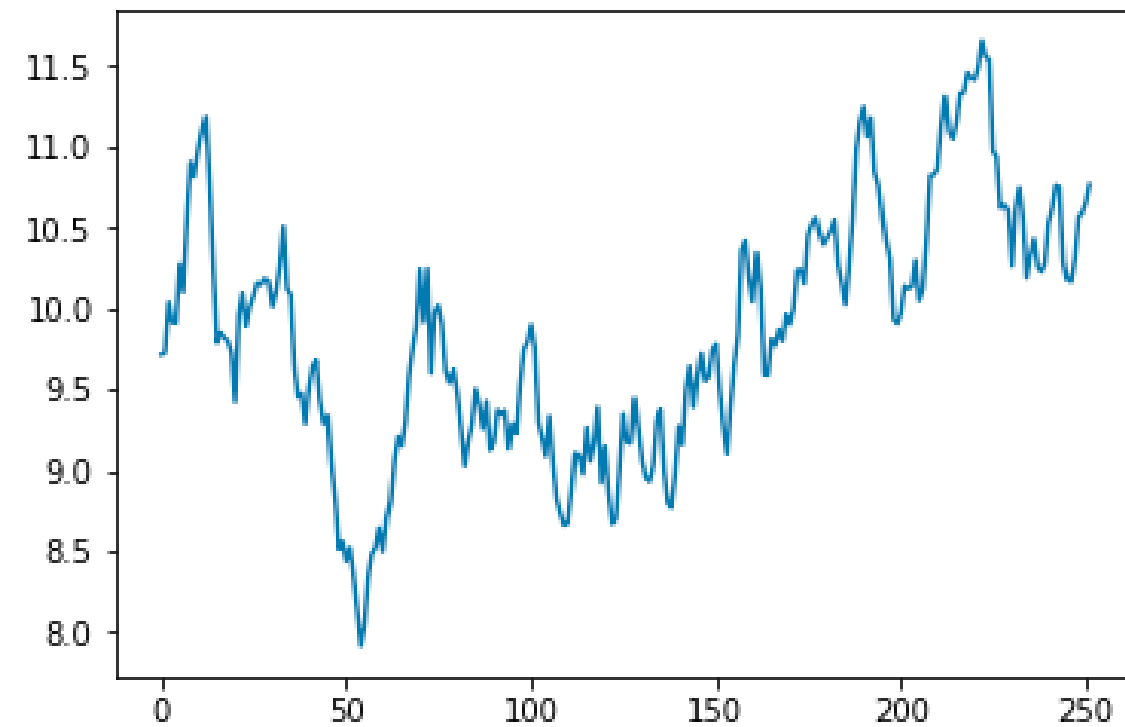
Quantitative Analyst | [QuantCourse.com](https://QuantCourse.com)



# Random Walks



Most often, random walks in finance are rather simple compared to physics:





# Random Walks in Python

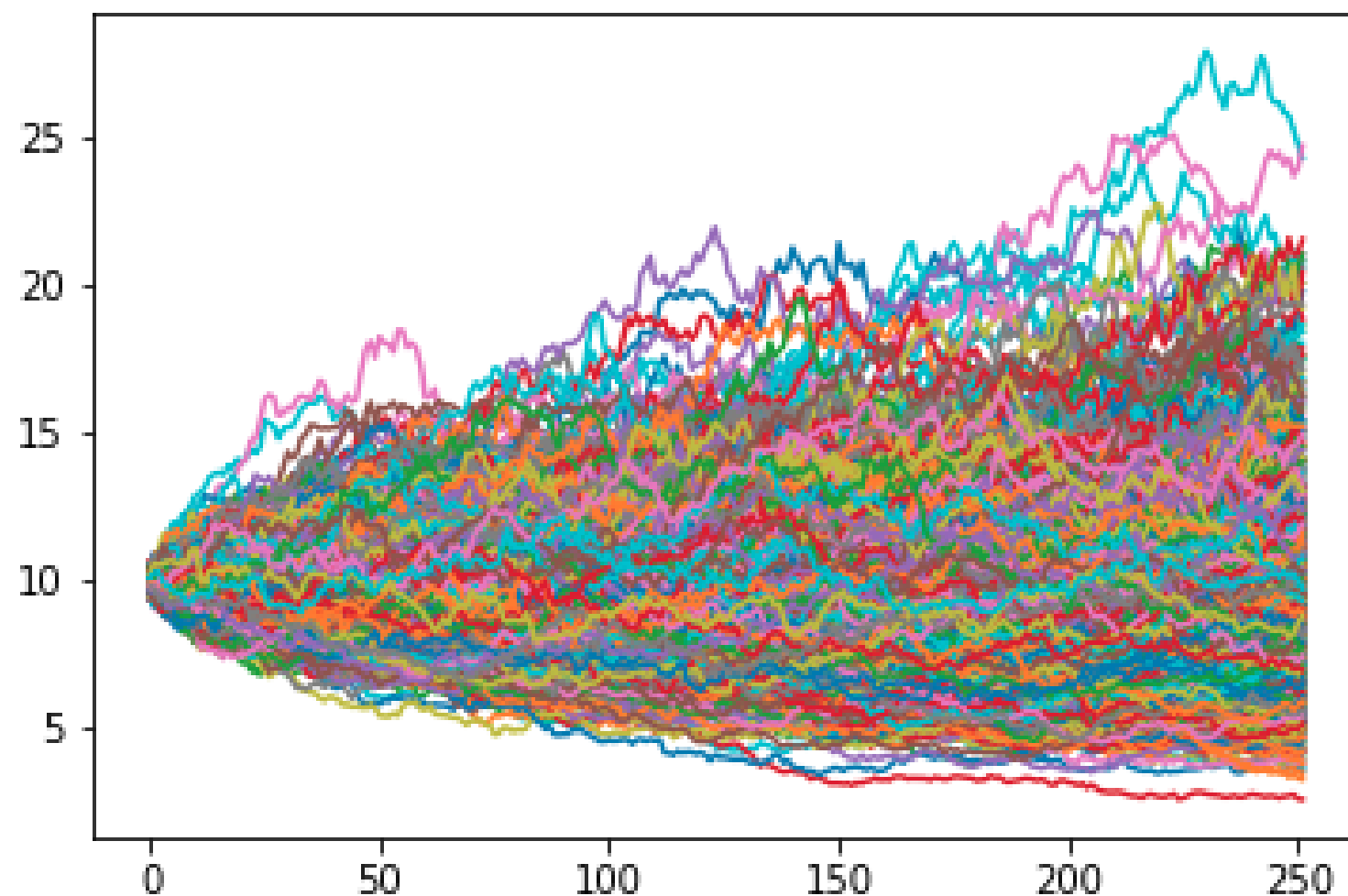
Assuming you have an object `StockReturns` which is a time series of stock returns.

To simulate a random walk:

```
In [1]: mu = np.mean(StockReturns)
In [2]: std = np.std(StockReturns)
In [3]: T = 252
In [4]: S0 = 10
In [5]: rand_rets = np.random.normal(mu, std, T) + 1
In [6]: forecasted_values = S0*(rand_rets.cumprod())
In [7]: forecasted_values
Out [7]: array([ 9.71274884,  9.72536923, 10.03605425 ... ])
```

# Monte Carlo Simulations

A series of Monte Carlo simulations of a single asset starting at stock price \$10 at T0. Forecasted for 1 year (252 trading days along the x-axis):





# Monte Carlo VaR in Python

To calculate the VaR(95) of 100 Monte Carlo simulations:

```
In [1]: mu = 0.0005
In [2]: vol = 0.001
In [3]: T = 252
In [4]: sim_returns = []
In [5]: for i in range(100):
In [6]:     rand_rets = np.random.normal(mu, vol, T)
In [7]:     sim_returns.append(rand_rets)
In [8]: var_95 = np.percentile(sim_returns, 5)
In [9]: var_95
Out [9]: -0.028
```



## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

**Let's practice!**



INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

# Understanding Risk

Dakota Wixom

Quantitative Analyst | [QuantCourse.com](https://QuantCourse.com)



# Summary

- Moments and Distributions
- Portfolio Composition
- Correlation and Co-Variance
- Markowitz Optimization
- Beta & CAPM
- FAMA French Factor Modeling
- Alpha
- Value at Risk
- Monte Carlo Simulations





## INTRO TO PORTFOLIO RISK MANAGEMENT IN PYTHON

**Good luck!**