ESTIMATION AND CI FOR VALUE AT RISK & EXPECTED SHORTFALL

Dr. Aric LaBarr

Institute for Advanced Analytics

VALUE AT RISK ESTIMATION

VaR Estimation

- Main Steps:
  1. Identify the variable of interest (asset value, portfolio value, credit losses, insurance claims, etc.)
  2. Identify the key risk factors that impact the variable of interest (assets prices, interest rates, duration, volatility, default probabilities, etc.)
  3. Perform deviations in the risk factors to calculate the impact in the variable of interest

VaR Estimation

- 3 Main Approaches
  1. Delta-Normal or Variance-Covariance Approach
  2. Historical Simulation (variety of approaches)
  3. Monte Carlo Simulation

DELTA-NORMAL APPROACH

Delta – Normal (Distribution)

- Suppose that the value, *V*, of an asset is a function of a Normally distributed risk factor, *RF*.
- What if the relationship between the two is linear?

$$V = \beta_0 + \beta_1 RF$$

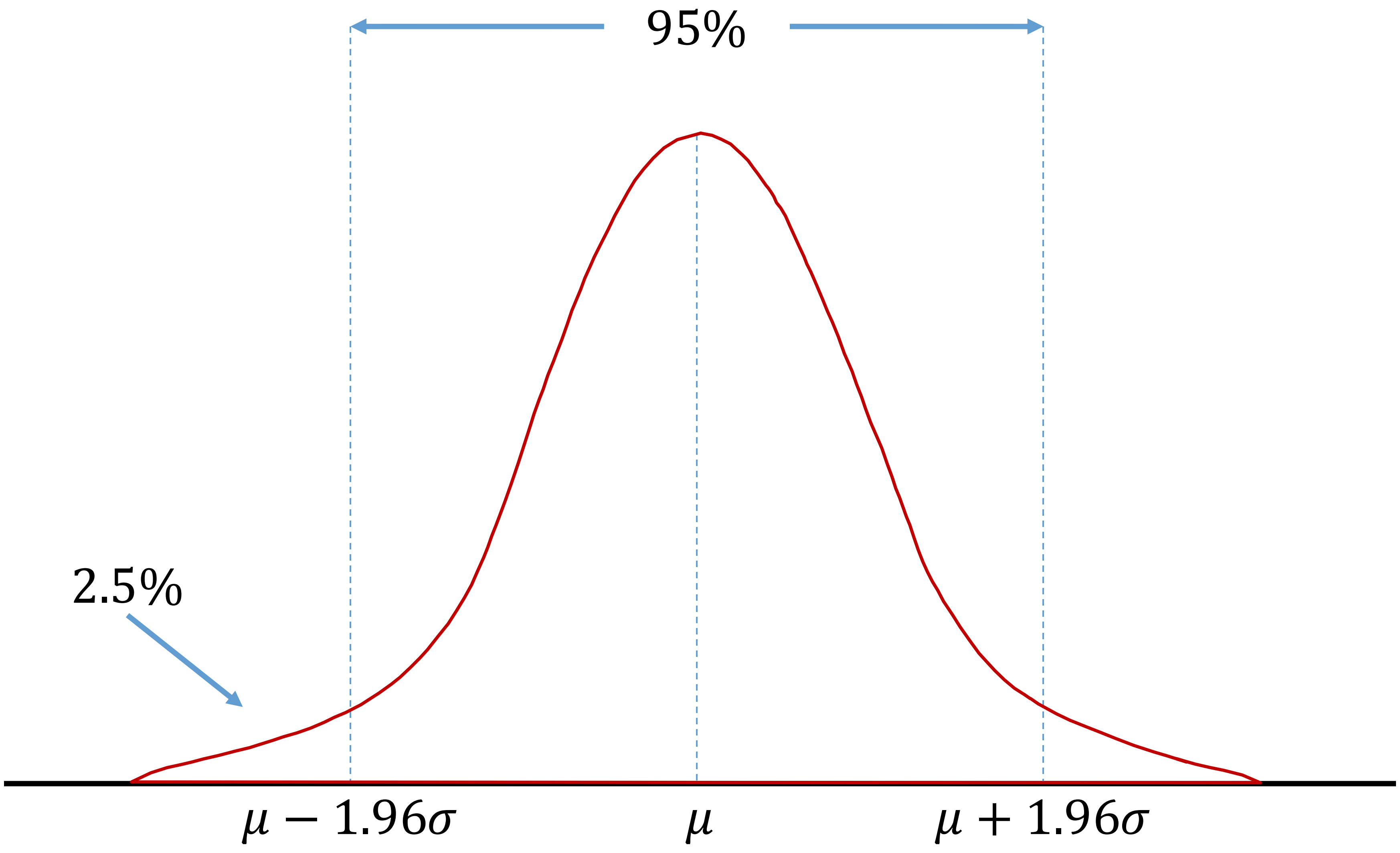
Delta – Normal (Distribution)

- Suppose that the value, *V*, of an asset is a function of a Normally distributed risk factor, *RF*.
- What if the relationship between the two is linear?

$$V = \beta_0 + \beta_1 RF$$

- Easy to calculate VaR!
- If *RF* is Normally distributed, then *V* would be as well.
- What is the 2.5% VaR on any Normal distribution?

Normal Distribution



Delta – Normal (Distribution)

- Suppose that the value, *V*, of an asset is a function of a Normally distributed risk factor, *RF*.
- What if the relationship between the two is linear?

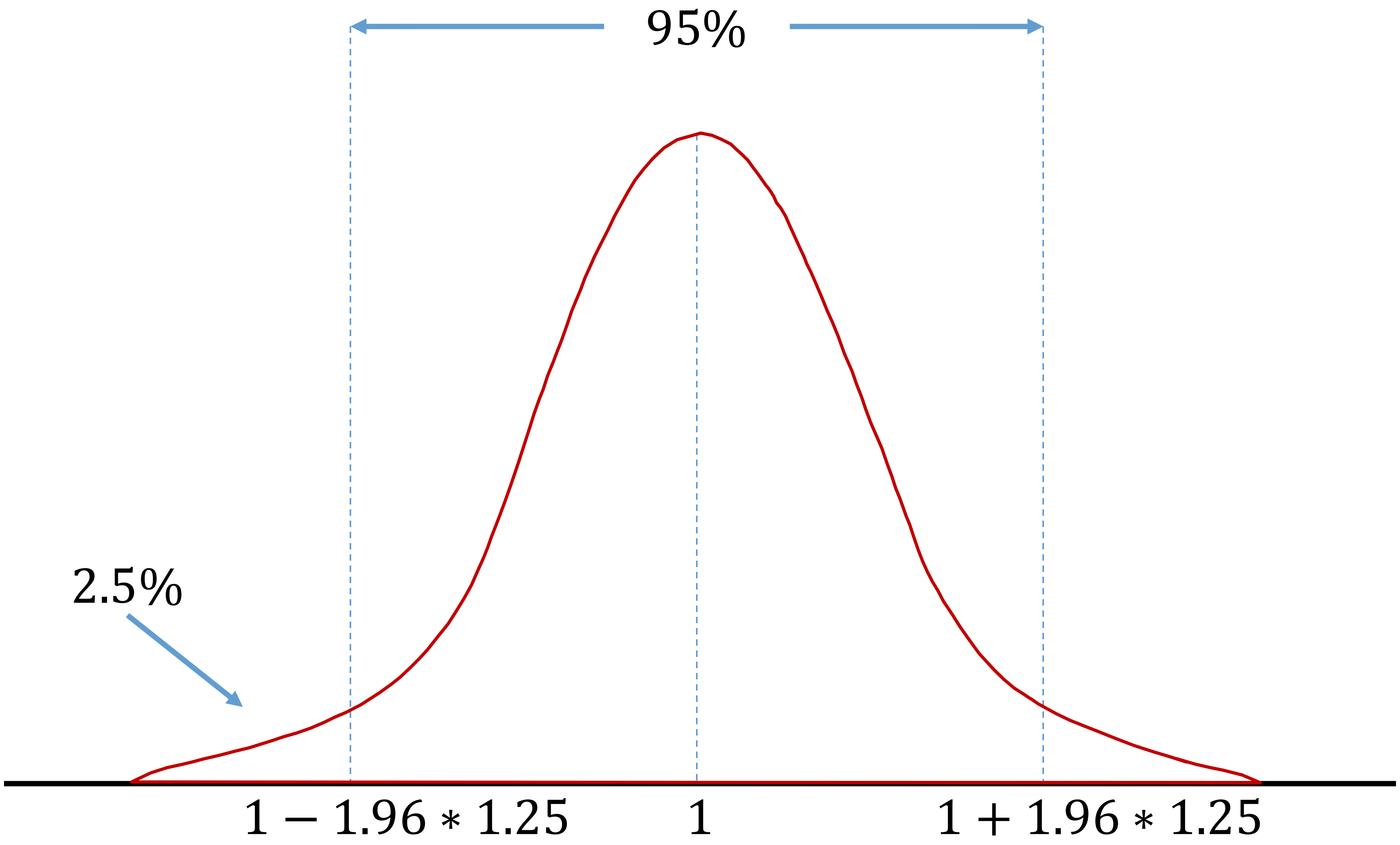
$$V = \beta_0 + \beta_1 RF$$

- Easy to calculate VaR!
- If *RF* is Normally distributed, then *V* would be as well.
- What is the 2.5% VaR on any Normal distribution?

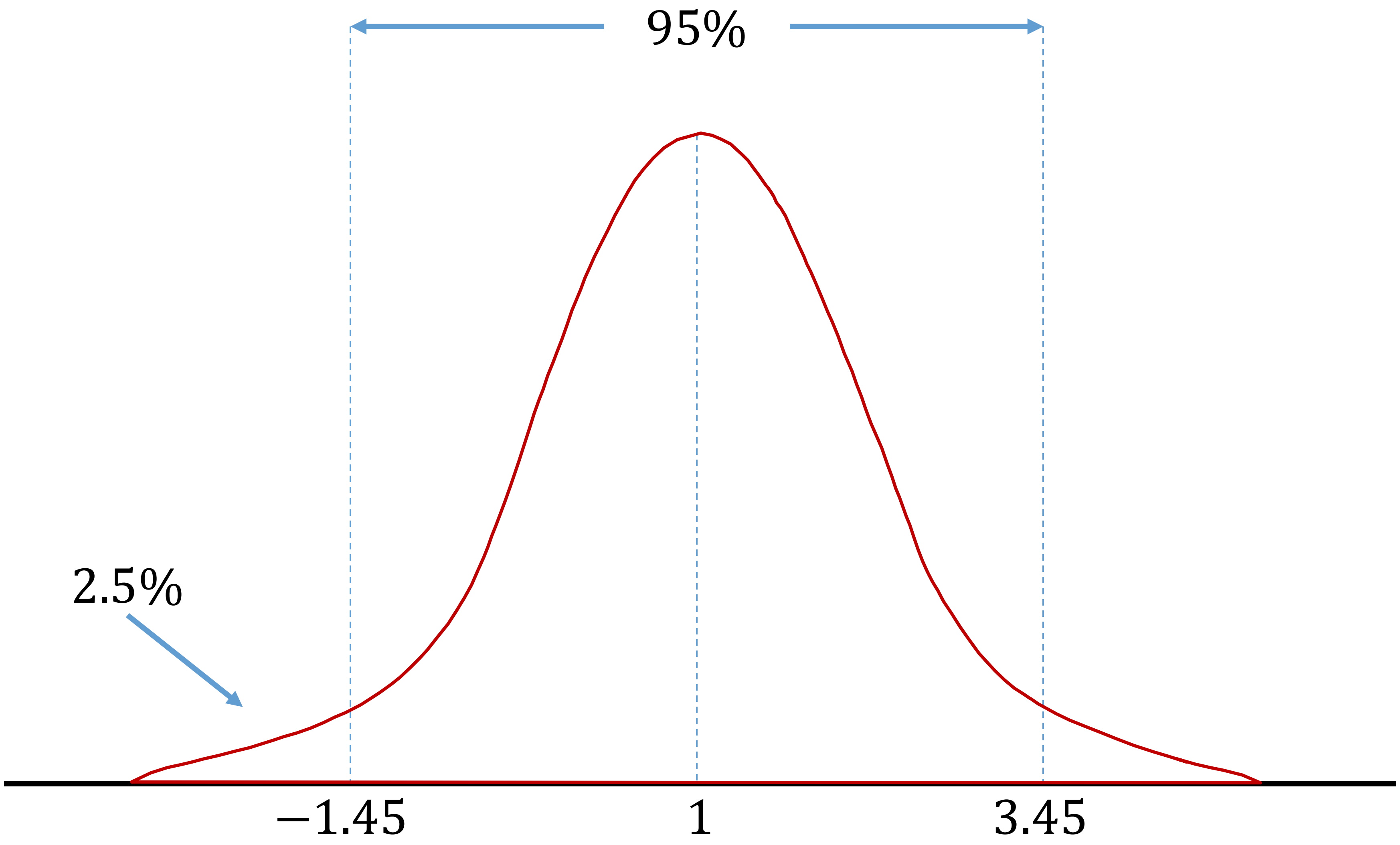
$$VaR_{2.5\%} = \mu - 1.96 * \sigma$$

- Just need to estimate μ and σ!

Normal Distribution – Example



Normal Distribution – Example



Delta – Normal

- Suppose that the value, *V*, of an asset is a function of a Normally distributed risk factor, *RF*.
- What if the relationship between the two is **non-linear**?

$$V = \beta_0 + \beta_1 RF^2$$

- How can we calculate the Value at Risk by taking advantage of the Normality assumption?

Delta – Normal

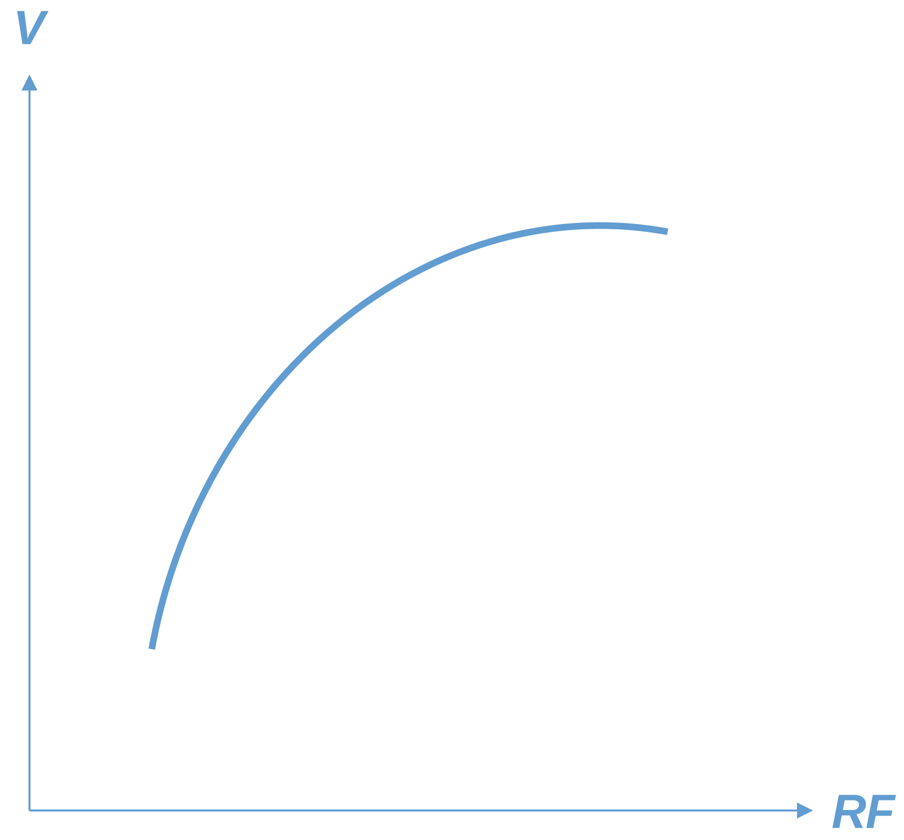
- Suppose that the value, *V*, of an asset is a function of a Normally distributed risk factor, *RF*.
- What if the relationship between the two is **non-linear**?

$$V = \beta_0 + \beta_1 RF^2$$

- How can we calculate the Value at Risk by taking advantage of the Normality assumption?
- Finding the extreme of a Normally distributed value and squaring that **DOES NOT EQUAL** the extreme value for the squared risk factor.

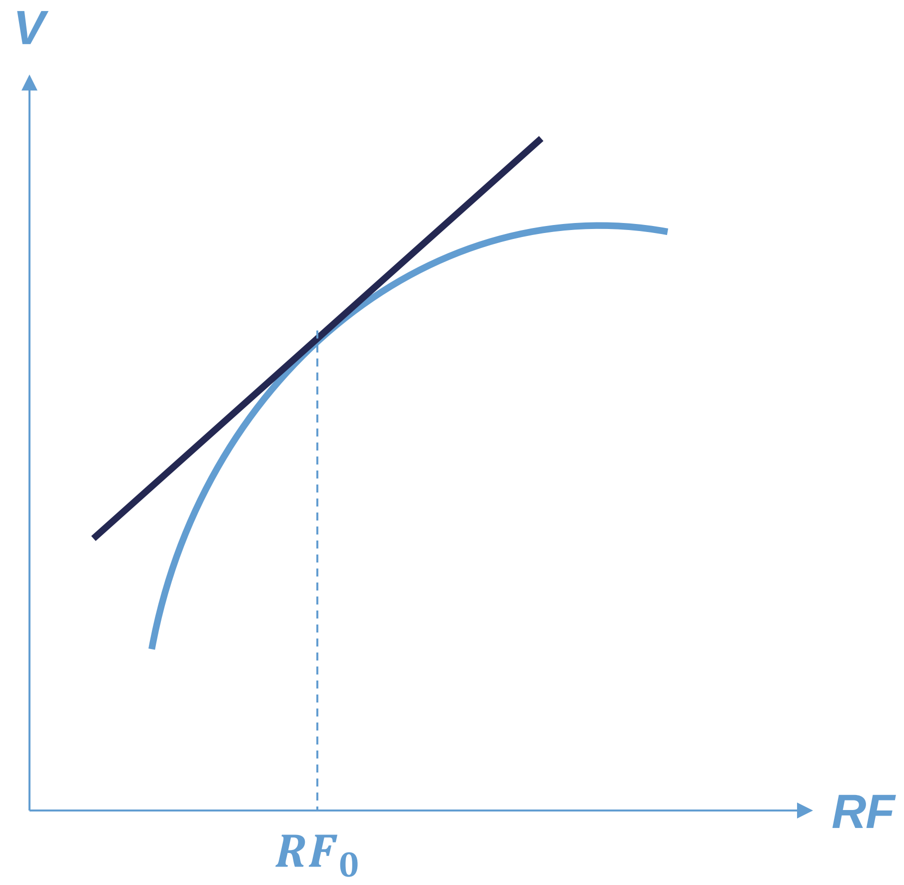
Delta (Derivative) – Normal

- The derivative of the relationship will help us approximate what we need.



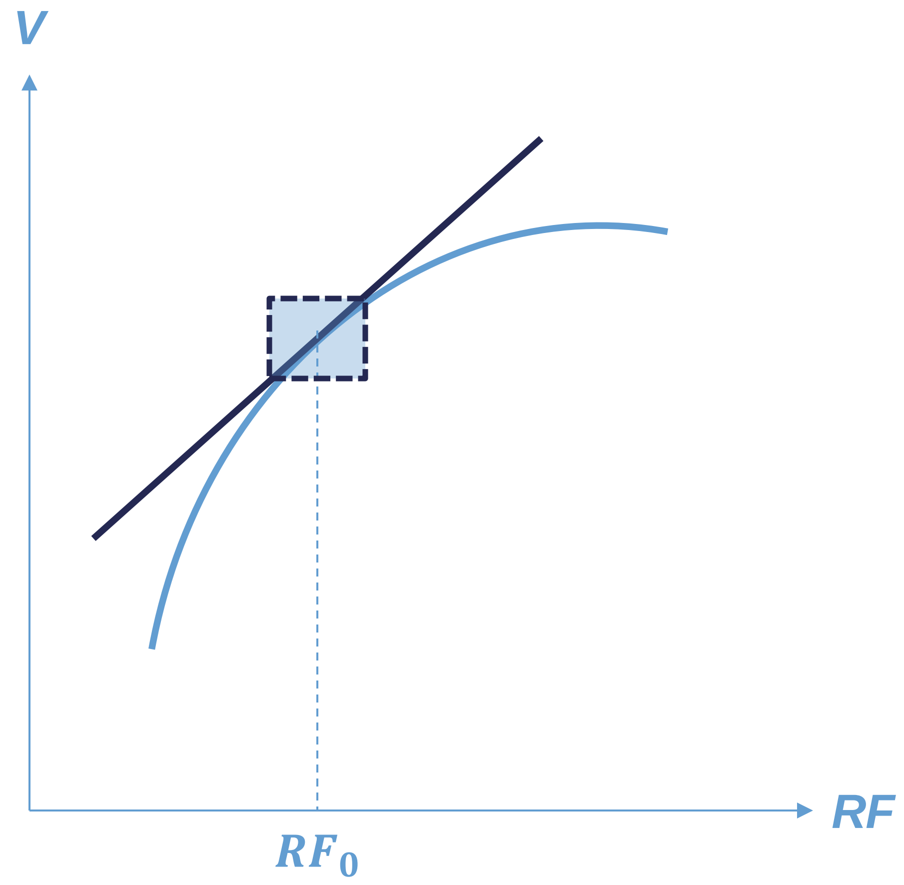
Delta (Derivative) – Normal

- The derivative of the relationship will help us approximate what we need.
- Remember, the derivative at a point (RF_0) is the tangent line at that point.



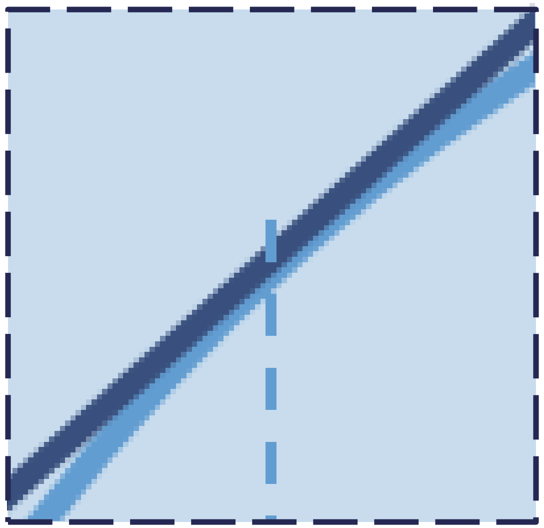
Delta (Derivative) – Normal

- The derivative of the relationship will help us approximate what we need.
- Remember, the derivative at a point (RF_0) is the tangent line at that point.
- But what if we zoom in closer since we think that RF_0 will only have small changes…?



Delta (Derivative) – Normal

- **This is approximately linear!**
- Small changes of the risk factor result in small changes of the value – **approximate using the slope!**
- Hence the name Delta – Normal.



Delta (Derivative) – Normal

- How to calculate the first derivative?
- If you know the formula relating the *RF* with the *V* then it is easy.

- Taylor-Series explansion:
- The change in any value is a function of all of the derivatives of that function:

$$dV = \frac{\partial V}{\partial RF} \cdot dRF + \frac{1}{2} \cdot \frac{\partial^2 V}{\partial RF^2} \cdot dRF^2 + \cdots$$

Delta (Derivative) – Normal

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF} \cdot dRF + \cancel{\frac{1}{2} \cdot \frac{\partial^2 V}{\partial RF^2} \cdot dRF^2 + \cdots}$$

- Evaluate the first derivative at a specific point RF_0, typically some initial value:

$$dV = \boxed{\left.\frac{\partial V}{\partial RF}\right|_{RF_0}} \cdot dRF$$

Delta (Derivative) – Normal

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF} \cdot dRF + \cancel{\frac{1}{2} \cdot \frac{\partial^2 V}{\partial RF^2} \cdot dRF^2 + \cdots}$$

- Evaluate the first derivative at a specific point RF_0, typically some initial value:

$$dV = \boxed{\left.\frac{\partial V}{\partial RF}\right|_{RF_0}} \cdot dRF \quad \Rightarrow \quad \Delta V = \delta_0 \cdot \Delta RF$$

Delta (Derivative) – Normal

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF}\bigg|_{RF_0} \cdot dRF \quad \Rightarrow \quad \Delta V = \delta_0 \cdot \Delta RF$$

- Change in value of the portfolio is a constant (δ_0) times the change in the risk factor.

Delta (Derivative) – Normal

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF}\bigg|_{RF_0} \cdot dRF \qquad \Rightarrow \qquad \Delta V = \delta_0 \cdot \Delta RF$$

- Change in value of the portfolio is a constant (δ_0) times the change in the risk factor.
- This is a **linear function!**

Delta (Derivative) – Normal (Distribution)

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF}\bigg|_{RF_0} \cdot dRF \quad \Rightarrow \quad \Delta V = \delta_0 \cdot \Delta RF$$

- What is the distribution of the change in *RF*?

$$\Delta RF = RF_t - RF_{t-1}$$

Delta (Derivative) – Normal (Distribution)

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \left.\frac{\partial V}{\partial RF}\right|_{RF_0} \cdot dRF \qquad \Rightarrow \qquad \Delta V = \delta_0 \cdot \Delta RF$$

- What is the distribution of the change in *RF*?

$$\Delta RF = RF_t - RF_{t-1}$$

Both are Normal

Delta (Derivative) – Normal (Distribution)

- Delta – Normal approach assumes that only the first derivative is actually important:

$$dV = \frac{\partial V}{\partial RF}\bigg|_{RF_0} \cdot dRF \quad \Rightarrow \quad \Delta V = \delta_0 \cdot \Delta RF$$

- What is the distribution of the change in *RF*?

$$\Delta RF = RF_t - RF_{t-1}$$

Both are Normal

- Difference of Normal distributions = Normal distribution!

Delta – Normal

- Difference of Normal distributions = Normal distribution!

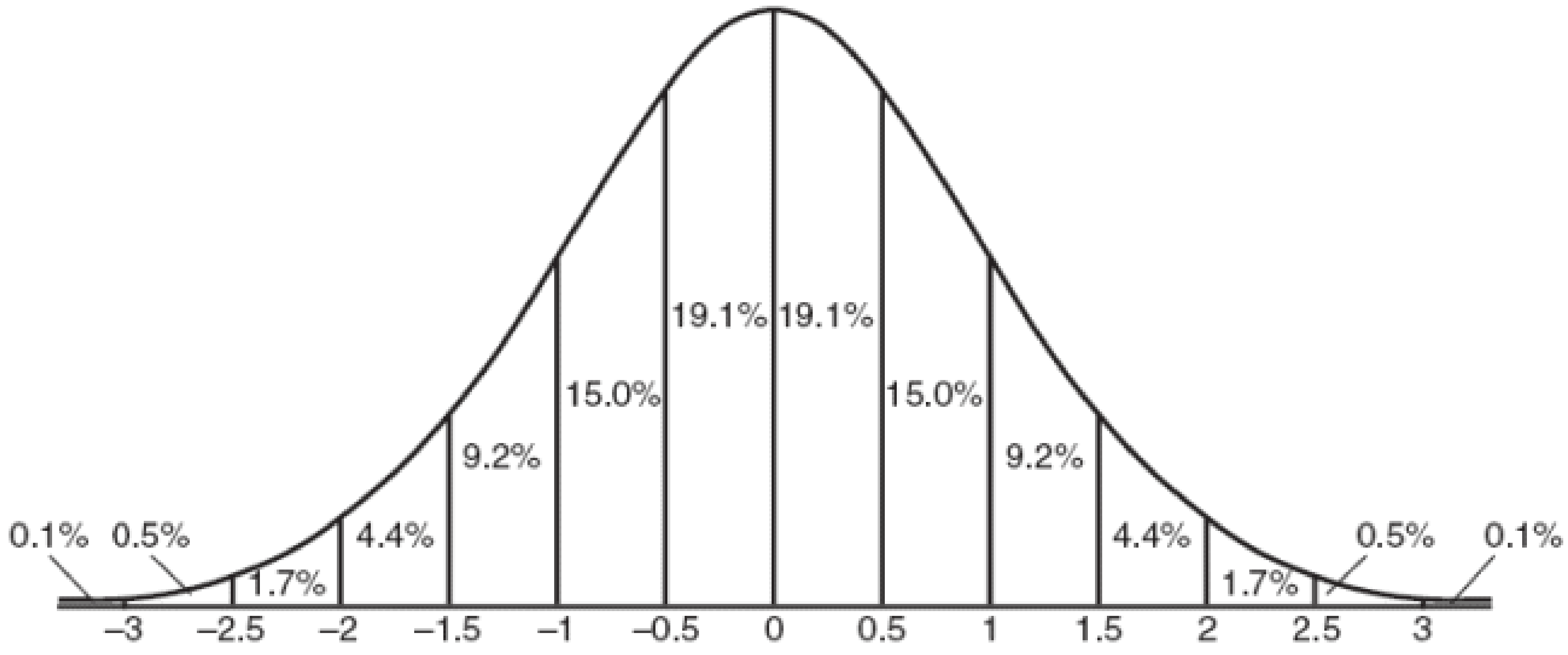
$$\Delta V = \delta_0 \cdot \Delta RF$$

Constant Normal distribution

- Therefore, the change in V, ΔV, also follows a Normal distribution.

Variance – Covariance

- The Delta – Normal approach is sometimes called the **variance-covariance approach** because you rely on the known variance relationships on the Normal distribution.



Variance – Covariance

- Popular percentiles for Value at Risk calculations (left tail):
  - 0.1% (99.9% confidence level) – $VaR = -3.09\sigma$
  - 0.5% (99.5% confidence level) – $VaR = -2.58\sigma$
  - 1.0% (99% confidence level) – $VaR = -2.33\sigma$
  - 5.0% (95% confidence level) – $VaR = -1.64\sigma$

Variance – Covariance

- The variance piece of “variance-covariance” is rather apparent from our last example, but what about the covariance piece?
- If all you have is a single portfolio or asset (or an independent one) then all you need is the variance.
- However, if you have multiple portfolio’s with a dependence structure then you need the covariance as well.

V – C: Single Position Portfolio

- Suppose you invested $100,000 in Apple today (bought Apple stock).
- Daily standard deviation of Apply return = 1.36%.
- Daily mean of Apply return = 0.1%.
- Data gathered from 2/08/2023 – 2/05/2025.

- Assume the Normal distribution on Apple returns (like the previous example).
- What is the daily VaR of your position at 1% confidence?

V – C: Single Position Portfolio

- What is the daily VaR of your position at 1% confidence?
- The percentile of the returns is -2.33 standard deviations below the mean of 0.

$$VaR = \$100{,}000 \times (0.001 + (-2.33) \times 0.0136) = -\$3{,}263.83$$

V – C: Single Position Portfolio

- What is the daily VaR of your position at 1% confidence?
- The percentile of the returns is -2.33 standard deviations below the mean of 0.

$$VaR = \$100{,}000 \times (0.001 + (-2.33) \times 0.0136) = -\$3{,}263.83$$

- With 99% confidence, you expect not to lose more than $3,263.83 by holding Apple stock for one day.

OR

- There is a 1% chance of losing at least $3,263.83 by holding Apple stock for one day.

V – C: Two Position Portfolio

- Suppose you invest $300,000 as follows:
  - $200,000 in MSFT and $100,000 in Apple
- Daily Returns:
  - $\bar{x}_{MSFT} = 0.1\%, \sigma_{MSFT} = 1.40\%$
  - $\bar{x}_{APPLE} = 0.1\%, \sigma_{APPLE} = 1.36\%$
  - Correlation of returns = 0.488
  - Assume Normal distribution for both.
- What is the 1% VaR of the portfolio?

V – C: Two Position Portfolio

- What is the 1% VaR of the portfolio?
- Need to find the variance of the portfolio's return!
- Portfolio's return = 2/3(MSFT Return) + 1/3(Apple Return) = 0.1%
- Portfolio's variance:

$$\sigma_P^2 = \left(\frac{2}{3}\right)^2 \sigma_M^2 + \left(\frac{1}{3}\right)^2 \sigma_A^2 + 2 * \left(\frac{2}{3}\right)\left(\frac{1}{3}\right) Cov(Apple, MSFT)$$

$$\sigma_P^2 = \left(\frac{2}{3}\right)^2 0.0140^2 + \left(\frac{1}{3}\right)^2 0.0136^2 + \frac{4}{9} * 0.488 * 0.0140 * 0.0136$$

$$\sigma_P^2 = 0.00015 \Rightarrow \sigma_P = 0.0122$$

V – C: Two Position Portfolio

- What is the 1% VaR of the portfolio?
- Need to find the variance of the portfolio's return!
- Portfolio's return = 2/3(MSFT Return) + 1/3(Apple Return) = 0.1%
- Portfolio's variance:

$$\sigma_P^2 = 0.00015 \Rightarrow \quad \sigma_P = 0.0122$$

- The 1% VaR is given by:

$$VaR = \$300{,}000 \times (0.001 + (-2.33) \times 0.0122) = -\$8{,}814.43$$

Getting Stock Data – Python

```
import yfinance as yf

stocks = yf.download("AAPL MSFT", start = "2007-01-01", group_by = 'tickers')

stocks = stocks.tail(500)
stocks['msft_r'] = stocks['MSFT'].pct_change()['Close']
stocks['aapl_r'] = stocks['AAPL'].pct_change()['Close']
```

Two Position Portfolio – Python

```
import numpy as np
import scipy.stats as sps
import locale

var_msft = np.cov(np.array([stocks['msft_r'][1:], stocks['aapl_r'][1:]]))[0,0]
var_aapl = np.cov(np.array([stocks['msft_r'][1:], stocks['aapl_r'][1:]]))[1,1]
cov_m_a = np.cov(np.array([stocks['msft_r'][1:], stocks['aapl_r'][1:]]))[0,1]
cor_m_a = np.corrcoef(np.array([stocks['msft_r'][1:], stocks['aapl_r'][1:]]))[0,1]

VaR_percentile = 0.01
AAPL_inv = 100000
MSFT_inv = 200000

var_port = (MSFT_inv / (MSFT_inv + AAPL_inv))**2*var_msft +
(AAPL_inv/(MSFT_inv+AAPL_inv))**2*var_aapl +
2*(AAPL_inv/(MSFT_inv+AAPL_inv))*(MSFT_inv/(MSFT_inv+AAPL_inv))*cov_m_a

VaR_DN_port = (AAPL_inv+MSFT_inv)*sps.norm.ppf(VaR_percentile)*np.sqrt(var_port)

locale.currency(VaR_DN_port, grouping = True)
```

Using Normality

- Under the assumption of Normality, we can get the following relationship between 1-day and n-day VaR:

$$VaR_N = \sqrt{N} \times VaR_1$$

- The general relation between a and b periods VaR is:

$$VaR_a = \sqrt{\frac{a}{b}} \times VaR_b$$

HISTORICAL SIMULATION

Historical Simulation Idea

- Non-parametric methodology (distribution free).
- Based solely on historical data.
- If history suggests that only 1% of the time Apple's daily returns were below -4%, what do you think the VaR at a 1% confidence level should be?

Historical: Single Position Portfolio

- $100,000 invested in Apple today.
- You have 500 observations on Apple's daily returns. You want to compute the daily VaR of your portfolio at the 1% confidence level.
- The 1% VaR will be a loss value that will not be exceeded 99% of the time **OR** the loss will be exceeded only 1% of the time.
- Find the 1% quantile of your data!

Historical: Single Position Portfolio

- Find the 1% quantile of your data!
- Using the 500 observations on daily returns, calculate the portfolio's value ($\$100{,}000 \times R_A$).
- Sort the 500 observations from worst to best.
- The 1% of 500 days is 5 – find a loss observation in our data set that is only **exceeded** 5 times.
- The 1% VaR will be the 6th observation of your sorted data set.

Historical: Single Position Portfolio

- The 1% VaR will be the 6th observation of your sorted data set.

Observation Number	Date	Return
374	8/5/2024	-4.82%
122	8/4/2023	-4.80%
281	3/21/2024	-4.09%
487	1/16/2025	-4.04%
145	9/6/2023	-3.58%
226	1/2/2024	-3.58%

Historical: Single Portfolio – Python

```
VaR_H_AAPL = AAPL_inv * stocks['aapl_r'].sort_values()[5]

locale.currency(VaR_H_AAPL, grouping = True)
```

```
'-$3,578.67'
```

Historical: Two Position Portfolio

- $200,000 invested in MSFT & $100,000 in Apple today.
- You have 500 observations on both returns.
- Calculate the portfolio's value using each one of the historical daily returns:

$$\$200{,}000 \times R_M + \$100{,}000 \times R_A$$

- Sort the 500 portfolio values from worst to best.
- The 1% VaR will be the 6th observation.

Historical: Two Position Portfolio

- The 1% VaR will be the 6th observation of your sorted data set.

Observation Number	Date	Portfolio Value
436	10/31/2024	-$13,926.46
496	1/30/2025	-$13,101.28
374	8/5/2024	-$11,348.08
366	7/24/2024	-$10,046.39
181	10/26/2023	-$9,963.40
469	12/18/2024	-$9,654.38

Historical: Two Position Portfolio – Python

```
stocks['port_v'] = MSFT_inv*stocks['msft_r'] + AAPL_inv*stocks['aapl_r’]

VaR_H_port = stocks['port_v'].sort_values()[5]

locale.currency(VaR_H_port, grouping = True)
```

```
'-$9,654.38'
```

Historical Simulation Assumptions

- There are some key assumptions we are making with this approach:
  1. The past will repeat itself.
  2. The historical period covered is long enough to get a good representation of “tail” events.

Historical Simulation Assumptions

- There are some key assumptions we are making with this approach:
  1. The past will repeat itself.
  2. The historical period covered is long enough to get a good representation of “tail” events.

- These have led to “alternative” historical simulation approaches…

Stressed VaR (and ES)

- Instead of basing calculations on the movements in market variables over the last n days, we can base calculations on movements during a period in the past that would have been particularly bad for the current portfolio.
- This produces measures known as “stressed VaR” and “stressed ES.”

Stressed VaR: Two Position Portfolio

- The 1% Stressed VaR will be the 6th observation of your sorted 500 observation data set from 3/15/2007 – 3/9/2009.

Observation Number	Date	Portfolio Value
390	9/29/2008	-$35,364.71
396	10/7/2008	-$22,638.74
434	12/1/2008	-$19,960.71
382	9/17/2008	-$19,541.83
417	11/5/2008	-$19,253.15
406	10/21/2008	-$18,063.40

Stressed VaR: Two Position – Python

```
stocks_stressed = yf.download("AAPL MSFT", start = "2007-01-01", group_by = 'tickers')
stocks_stressed['msft_r'] = stocks_stressed['MSFT'].pct_change()['Close']
stocks_stressed['aapl_r'] = stocks_stressed['AAPL'].pct_change()['Close’]

stocks_stressed['port_v'] = MSFT_inv*stocks_stressed['msft_r'] + AAPL_inv*stocks_stressed['aapl_r']
stocks_stressed['ma'] = stocks_stressed['port_v'].rolling(500).mean()
stocks_stressed = stocks_stressed.loc[:stocks_stressed.sort_values(by = ['ma']).index[0]].tail(500)

stressed_VaR_H_port = stocks_stressed['port_v'].sort_values()[5]
locale.currency(stressed_VaR_H_port, grouping = True)
```

```
'-$18,063.40'
```

Extending to Multiple Day Return

- When extending to multiple days you could do one of either of the following:
  1. Calculate *n* day returns first, then run historical simulation.
  2. Use any of the historical simulation approaches, but consider the daily returns as **starting points**. Record the return of the next consecutive *n* days to get a real example of *n* day returns for simulation.

MONTE CARLO SIMULATION

MC Simulation: Main Idea

- Estimate the VaR through the simulation of results of statistical / mathematical models.
- Simulate the value of the portfolio using some statistical / financial model that explains the behavior of the random variables of interest.
- If we have “enough” simulations then we have simulated the distribution of the portfolio’s value.
- Use the empirical distribution to find the VaR at any point you wish.

MC Simulation

- Monte Carlo simulation is not easy to use, but able to handle the following details:
  - Non-normal models
  - Nonlinear models
  - Multidimensional problems
  - History changing

MC: Single Position Portfolio

- You want to invest $100,000 in Apple stock for one day.

$$P_1 = P_0 + r_{0,1} * P_0$$

OR

$$P_1 = P_0 * (1 + r_{0,1})$$

Variable of Interest

Return (Risk Factor)

MC: Single Position Portfolio

- Let’s use 10,000 simulations.
- In each simulation:
  - Draw a value from Normal distribution to get $r_{0,1}$.
  - Get the portfolio’s value: $V = 100{,}000 \times \left(1 + r_{0,1}\right)$

MC: Single Position Portfolio

- Let's use 10,000 simulations.
- In each simulation:
  - Draw a value from Normal distribution to get $r_{0,1}$.
  - Get the portfolio's value: $V = 100{,}000 \times \left(1 + r_{0,1}\right)$
- Now we have 10,000 simulated portfolio values.
- Create an empirical distribution of portfolio changes to look at quantiles and calculate VaR.
- 1% VaR would be 101st observation.

MC: Two Position Portfolio

- Suppose you invest \$300,000 as follows:
  - \$200,000 in MSFT and \$100,000 in Apple
- Daily Returns:
  - $\bar{x}_{MSFT} = 0.1\%, \sigma_{MSFT} = 1.40\%$
  - $\bar{x}_{APPLE} = 0.1\%, \sigma_{APPLE} = 1.36\%$
  - Correlation of returns = 0.488
  - Assume Normal distribution for both.
- What is the 1% VaR of the portfolio?

MC: Two Position Portfolio

- Let's use 10,000 simulations.
- In each simulation:
  - Draw a value from bivariate Normal distribution with covariance matrix to get $r_{0,1}$ for each Apple and Microsoft.

OR

  - Draw a value from a Normal distribution for each Apple and Microsoft and add correlation structure after (Cholesky decomposition).
  - Portfolio's value: $V = 100{,}000 \times \left(1 + r_{AAPL,0,1}\right) + 200{,}000 \times \left(1 + r_{MSFT,0,1}\right)$

MC: Two Position Portfolio

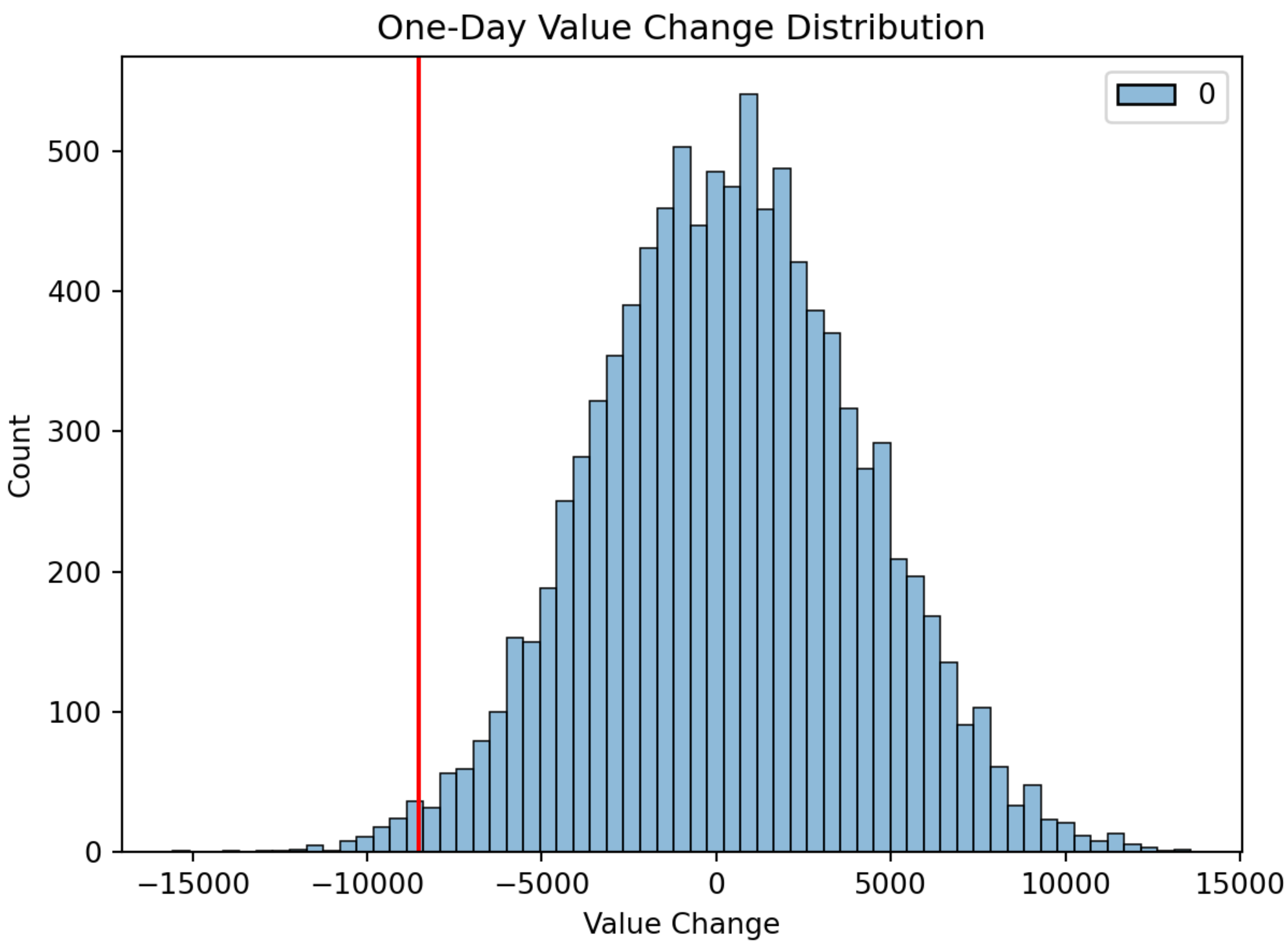
- Now we have 10,000 simulated portfolio values.
- Create an empirical distribution of portfolio changes to look at quantiles and calculate VaR.
- 1% VaR would be 101st observation.

MC: Two Position Portfolio – Python

```
n_sim = 10000
R = np.array([[1, cor_m_a], [cor_m_a, 1]])
U = sp.linalg.cholesky(R, lower = False)


msft_r = np.random.normal(loc = np.mean(stocks['msft_r']), scale = np.sqrt(var_msft), size = n_sim)
aapl_r = np.random.normal(loc = np.mean(stocks['aapl_r']), scale = np.sqrt(var_aapl), size = n_sim)
Both_r = np.array([msft_r, aapl_r])

port_r = U @ Both_r
port_r = np.transpose(port_r)


port_v = MSFT_inv*port_r[:,0] + AAPL_inv*port_r[:,1]
VaR = np.quantile(port_v, VaR_percentile)
locale.currency(VaR, grouping = True)
```

```
'-$8,553.97'
```

MC: Two Position Portfolio – R



Extensions

- Due to the Normality assumption in the last example we could have also used the Delta – Normal (Variance-covariance) approach.
- However, this illustrates the Monte Carlo principle and shows how easily it can be extended.
- You can use a mixture of distributions and variance structures that change over time with the Monte Carlo simulation approach.

Monte Carlo Simulation Assumptions

- You are assuming a couple of key things with this approach:
  1. The model used is an accurate representation of the reality.
  2. The number of draws is enough to capture the tail behavior.

?

Comparison of Three Approaches

	Delta – Normal / Variance – Cov.	Historical Simulation	Monte Carlo Simulation
Attractions	Intuitive	Intuitive and easy to explain	Extremely powerful and flexible
	Easy formula for VaR	Non-parametric	Handles non-linearity, non-normality, etc.
	Ideal for linear and Normal factors	Easy to implement	Ideal for complex problems
Limitations	Normality assumption	Problems obtaining data	Hard to explain
	Linearity assumption	Complete dependence on past	Computer-time intensive
	Covariance might not be well behaved	Length of estimation	Considerable investment

CONFIDENCE INTERVAL ESTIMATION

Confidence Intervals for VaR

- Under the Normality assumption:

$$VaR = q_\alpha \times \sigma$$

$$SE(VaR) = SE(\hat{\sigma})$$

$$CI(\hat{\sigma}) = \left(\sqrt{\frac{(n-1)\hat{\sigma}^2}{\chi^2_{\frac{\alpha}{2},n-1}}}, \sqrt{\frac{(n-1)\hat{\sigma}^2}{\chi^2_{1-\frac{\alpha}{2},n-1}}} \right)$$

Confidence Intervals for VaR – Python

```
sigma_low = np.sqrt(var_port * (stocks.shape[0] - 1) / sps.chi2.ppf((1 - VaR_percentile/2), (stocks.shape[0] - 1)))

sigma_up = np.sqrt(var_port * (stocks.shape[0] - 1) / sps.chi2.ppf((VaR_percentile/2), (stocks.shape[0] - 1)))


VaR_DN_port = (AAPL_inv+MSFT_inv)*sps.norm.ppf(VaR_percentile)*np.sqrt(var_port)

VaR_L = (AAPL_inv+MSFT_inv)*sps.norm.ppf(VaR_percentile)*sigma_low

VaR_U = (AAPL_inv+MSFT_inv)*sps.norm.ppf(VaR_percentile)*sigma_up


print("The VaR is", locale.currency(VaR_DN_port, grouping = True), "with confidence interval (", locale.currency(VaR_L, grouping = True), ",", locale.currency(VaR_U, grouping = True), " )" )
```

```
The VaR is -$8,535.11 with confidence interval ( -$7,888.72 , -$9,287.95 )
```

Bootstrapping

- Steps of Bootstrapping:
  1. Resample from the simulated data using their empirical distribution; or rerun the simulation several times.
  2. In each new sample (from either approach in step 1) calculate the VaR.
  3. Repeat steps 1 and 2 many times to get several VaR estimates; use these estimates to get the expected VaR and its confidence interval.

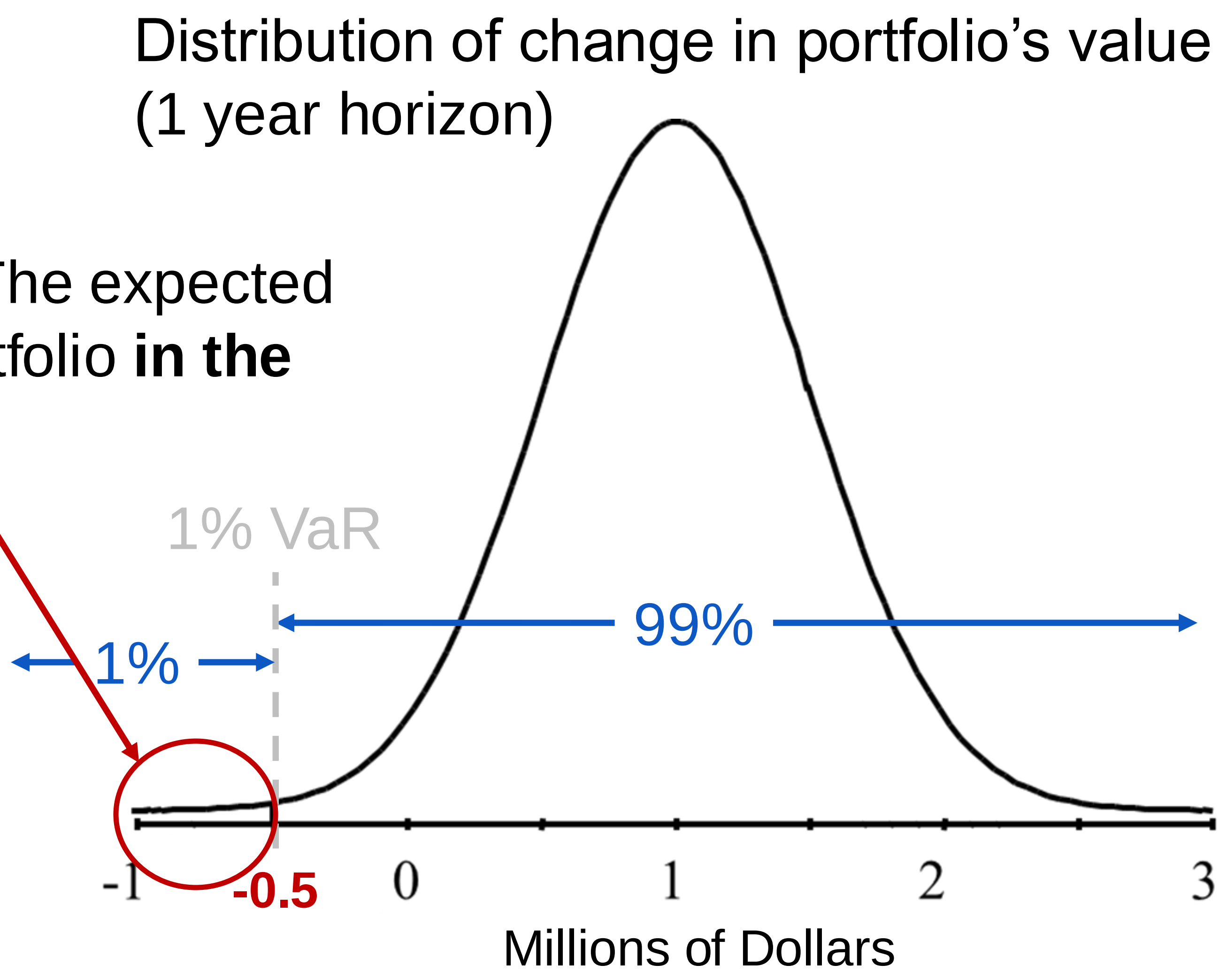
Bootstrapping – Python

```
n_bootstraps = 1000
sample_size = 1000

data = []
for i in range(n_bootstraps):
  bootstrap_sample = value_df.sample(n = sample_size)
  VaR_boot = np.quantile(bootstrap_sample, VaR_percentile)
  data.append(VaR_boot)

VaR_boot_U = np.quantile(data, 0.975)
VaR_boot_L = np.quantile(data, 0.025)
```

CONDITIONAL VALUE AT RISK (EXPECTED SHORTFALL) ESTIMATION

Visualizing CVaR (ES) – Left Tail

CVaR (or ES): The expected value of the portfolio **in the circled region.**



CVaR (ES) Estimation

- 3 Main Approaches
  1. Delta-Normal or Variance-Covariance Approach
  2. Historical Simulation
  3. Monte Carlo Simulation

CVaR: Variance – Covariance

- In the case of the variance-covariance approach, the CVaR can be calculated as follows:

$$CVaR = \mu - \sigma \times \frac{e^{\left(\frac{-q_\alpha^2}{2}\right)}}{\alpha\sqrt{2\pi}}$$

- σ: standard deviation
- α: percentile we are working on (e.g. 1%)
- q_α: tail 100α percentile of the standard Normal distribution

(e.g. -2.33)

CVaR: Variance – Covariance – Python

```
ES_DN_port = (0 - np.sqrt(var_port) * np.exp(-(sps.norm.ppf(VaR_percentile)**2)/2)/
(VaR_percentile * np.sqrt(2 * np.pi)))*(AAPL_inv + MSFT_inv)


locale.currency(ES_DN_port, grouping = True)
```

```
'-$9,778.38'
```

CVaR: Historical Simulation

- Suppose you have 500 observations for the daily return on Apple and Microsfot.
- In order to find CVaR at the 1% confidence level, you need to do the following:
  - Sort the data from worst to best
  - Calculate the VaR (6^{th} value in this example)
  - The CVaR is the **average** of the values that are worst than the VaR (the average of the first 5 values in our example)

Historical: Two Position Portfolio

- The 1% CVaR will be the average of the first 5 observations of your sorted data set.

Observation Number	Date	Portfolio Value
436	10/31/2024	-$13,926.46
496	1/30/2025	-$13,101.28
374	8/5/2024	-$11,348.08
366	7/24/2024	-$10,046.39
181	10/26/2023	-$9,963.40
469	12/18/2024	-$9,654.38

$$\text{CVaR} = -\$11{,}677.12$$

CVaR: Variance – Covariance – R

```
ES_H_port = np.mean(stocks['port_v'].sort_values().head(5))


locale.currency(ES_H_port, grouping = True)
```

```
'-$11,677.12'
```

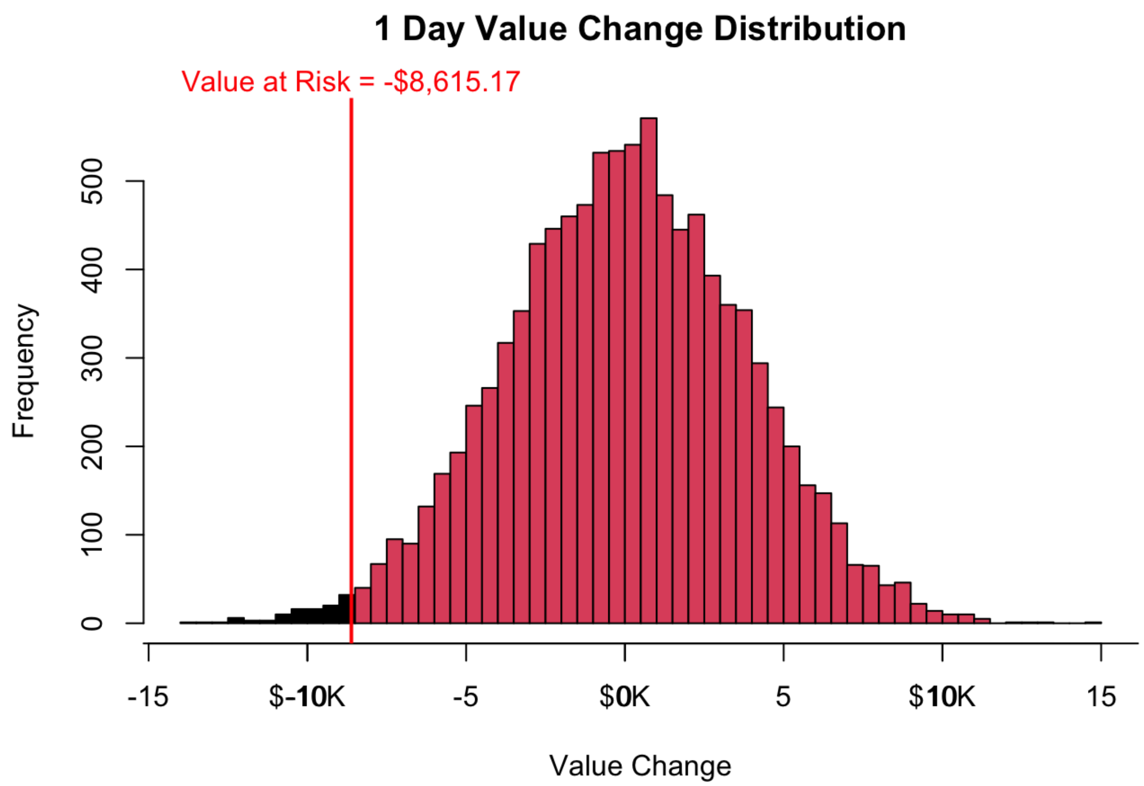
CVaR: MC Simulation

- Follow the steps described earlier to create the 10,000 simulated, sorted, portfolio values for the VaR calculation.
- Take the average of all the values that are worst than the VaR.
- Example – average of first 100 observations is 1% CVaR.

CVaR: MC Simulation

```
ES = np.mean(port_v[port_v < VaR])

locale.currency(ES, grouping = True)
```

```
'-$9,847.63'
```



?