AMS-511 Foundations of Quantitative Finance

Fall 2020 — Solutions 04

Robert J. Frey, Research Professor
Stony Brook University, Applied Mathematics and Statistics

Robert.Frey@StonyBrook.edu
http://www.ams.sunysb.edu/~frey

Question 1

You are given the following annual data on four investments, $i \in \{1, 2, 3, 4\}$ in a market M:

- $r_f = 0.02$
- $\mu_M = 0.085$
- $\sigma_M = 0.105$
- $\boldsymbol{\beta} = \{0.9, 1.2, 0.6, 2.1\}$
- $\boldsymbol{\sigma}_\epsilon = \{0.05, 0.07, 0.04, 0.09\}$

Under the assumption that the CAPM applies:

- Compute the mean vector of the asset returns.
- Compute the correlation and covariance matrices of the asset returns.
- Compute the mean-variance efficient portfolio such that $\mathbb{1}^{\mathrm{T}} \mathbf{x} = 1$. Assume there are no further constraints, specifically, that short positions are permitted
- Compute the mean and standard deviation of that portfolio.
- The investor wishes to keep 10% of its assets in cash and place the remainder in the optimal portfolio. Assuming returns are Normally distributed what are the mean and standard deviation of return for this combined cash-risky portfolio?

Solution

In[◦]:=
```
nRiskFree = 0.02;
nMktMean = 0.085;
nMktSdev = 0.105;
vnBeta = {0.9, 1.2, 0.6, 2.1};
vnErrSdev = {0.05, 0.07, 0.04, 0.09};
```

- We can use the CAPM to compute the mean vector and covariance matrix, and then use these statistics to compute the correlation matrix.

$$\mu_i = r_f + \beta_i(\mu_M - r_f)$$

```
In[•]:= vnMean = nRiskFree + vnBeta (nMktMean - nRiskFree);
mnCovariance =
  nMktSdev KroneckerProduct[vnBeta, vnBeta] + DiagonalMatrix[vnErrSdev^2];
```

The correlation matrix first requires us to extract the standard deviations from the covariance matrix and then use them to normalize the covariances using $\rho_{i,j} = \sigma_{i,j} / \sigma_i \sigma_j$.

```
In[•]:= vnSdev = Sqrt[Diagonal[mnCovariance]];
mnCorrelation = mnCovariance / KroneckerProduct[vnSdev, vnSdev];
```

The Print[ ] function is useful for simple reporting.

```
In[•]:= Print["μ = ", MatrixForm[vnMean]]
Print["Σ = ", MatrixForm[mnCovariance]]
Print["C = ", MatrixForm[mnCorrelation]]
```

$$\mu = \begin{pmatrix} 0.0785 \\ 0.098 \\ 0.059 \\ 0.1565 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 0.08755 & 0.1134 & 0.0567 & 0.19845 \\ 0.1134 & 0.1561 & 0.0756 & 0.2646 \\ 0.0567 & 0.0756 & 0.0394 & 0.1323 \\ 0.19845 & 0.2646 & 0.1323 & 0.47115 \end{pmatrix}$$

$$C = \begin{pmatrix} 1. & 0.970026 & 0.965399 & 0.97711 \\ 0.970026 & 1. & 0.963989 & 0.975683 \\ 0.965399 & 0.963989 & 1. & 0.971029 \\ 0.97711 & 0.975683 & 0.971029 & 1. \end{pmatrix}$$

- The efficient mean-variance solution is $x_i \propto \beta_i / \sigma_{\epsilon_i}$ and portfolio mean and standard deviation are

```
In[•]:= vnX = #/Total[#] &[vnBeta / vnErrSdev^2];
Print["x = ", MatrixForm[vnX]]
```

$$x = \begin{pmatrix} 0.29052 \\ 0.197633 \\ 0.302625 \\ 0.209222 \end{pmatrix}$$

```
In[•]:= nPortMean = vnMean.vnX;
nPortSdev = Sqrt[vnX.mnCovariance.vnX];
Print["μ𝒫 = " , nPortMean]
Print["σ𝒫 = ", nPortSdev]
```

$\mu_{\mathcal{P}}$ = 0.092772

$\sigma_{\mathcal{P}}$ = 0.364025

- Given a 10% cash position, $\phi = 0.9$ below

$$\begin{pmatrix} \sigma_{\mathcal{P}(\phi)} \\ \mu_{\mathcal{P}(\phi)} \end{pmatrix} = (1-\phi)\begin{pmatrix} 0 \\ r_f \end{pmatrix} + \phi \begin{pmatrix} \sigma_{\mathcal{P}} \\ \mu_{\mathcal{P}} \end{pmatrix}$$

In[•]:=
```
ϕ = 0.9;
{nLeverageSdev, nLeverageMean} = (1 - ϕ) {0, nRiskFree} + ϕ {nPortSdev, nPortMean};
```

In[•]:=
```
Print[MatrixForm[{"σ𝒫 (0.9)", "μ𝒫 (0.9)"}],
 " = ", MatrixForm[{nLeverageSdev, nLeverageMean}]]
```

$$\begin{pmatrix} \sigma_{\mathcal{P}}\,(0.9) \\ \mu_{\mathcal{P}}\,(0.9) \end{pmatrix} = \begin{pmatrix} 0.327622 \\ 0.0854948 \end{pmatrix}$$

Question 2

You have a portfolio with estimated monthly mean return of 0.8% and monthly standard deviation of 3.5%.

- Assuming the portfolio returns follow a Normal distribution, what is the VaR and CVaR at a 99.9% confidence level?
- Assuming the portfolio returns follow a Student t distribution with 4 degrees of freedom, what is the VaR and CVaR at a 99.9% confidence level?

Note: Consider a random variable $R \approx$ StudentTDistribution[μ, σ, ν]. Generally, the parameter μ is known as a *location* parameter and σ a *scale* parameter. The mean of the Student t is μ, but it standard deviation is not σ, but

$$\text{StandardDeviation}[R] = \sigma \sqrt{\frac{\nu}{\nu - 2}}$$

Thus, the parameter σ is related to but not identical to the standard deviation. Given the standard deviation as in this problem, you must solve for the Student t scale parameter σ to specify the distribution correctly.

In[•]:=
```
StandardDeviation[StudentTDistribution[m, s, d]]
```

Out[•]=

$$\begin{cases} \sqrt{\frac{d}{-2+d}}\; s & d > 2 \\ \text{Indeterminate} & \text{True} \end{cases}$$

Solution

The scale parameter σ of the Student t distribution with ν degrees of freedom can be calculated by

$$\text{scale} = \sqrt{\frac{\nu - 2}{\nu}}\ \text{sdev}$$

The VaR and CVaR are calculated

$$\text{VaR}_{\delta,\chi} = \underset{r}{\text{argmin}}[1 - F_{\delta}\,(r)\ \leq\ 1 - \chi]$$

$$\text{CVaR}_{\delta,\chi} = \text{E}[\,r\ |\ r\ \leq\ \text{VaR}_{\delta,\chi}]\ = \frac{1}{1-\chi}\int_{-\infty}^{\text{VaR}_{\delta,\chi}} r\, dF_{\delta}[r]$$

Normal

The required parameters for the Normal distribution are

In[∘]:=
```
nPortMu = 0.008;
nPortSigma = 0.035;
nConfLevel = 0.999;
```

The VaR and CVaR are

In[∘]:=
```
nVaRN = InverseCDF[NormalDistribution[nPortMu, nPortSigma], 1 - nConfLevel]
```

Out[∘]= -0.100158

In[∘]:=
$$\text{nCVaRN} = \frac{1}{1 - \text{nConfLevel}}$$
```
  Integrate[r PDF[NormalDistribution[nPortMu, nPortSigma], r], {r, -∞, nVaRN}]
```

Out[∘]= -0.109848

Student t

We need the degrees of freedom and scale.

In[∘]:=
```
nDegOfFree = 4;
```
$$\text{nTScale} = \sqrt{\frac{\text{nDegOfFree} - 2}{\text{nDegOfFree}}}\ \text{nPortSigma};$$

And the VaR and CVaR are

In[∘]:=
```
nVaRT =
 InverseCDF[StudentTDistribution[nPortMu, nTScale, nDegOfFree], 1 - nConfLevel]
```

Out[∘]= -0.169527

In[∘]:=
$$\text{nCVaRT} = \frac{1}{1 - \text{nConfLevel}}\ \text{Integrate}[$$
```
   r PDF[StudentTDistribution[nPortMu, nTScale, nDegOfFree], r], {r, -∞, nVaRT}]
```

Out[∘]= -0.231722

Summary

We can produce a nice report by using the Grid[ ] function. The argument is a 2×1 column vector consisting of a title string and a TableForm[ ] object. Note the use of Style[ ] to format the title text.

In[∘]:=
```
? Grid
```

Out[∘]=

Symbol

Grid[{{$expr_{11}$, $expr_{12}$, …}, {$expr_{21}$, $expr_{22}$, …}, …}] is
an object that formats with the $expr_{ij}$ arranged in a two–dimensional grid.

Note that subscripts can be entered as, for example, "nVaRN[CTRL]_0.999”. If there is following text, then hitting the right-arrow-key (*not* the character →) will bring you back up a level to the baseline. Superscripts work the same way using “x[CTRL]^2” and a right-arrow will bring you down to the text baseline. You don’t have to hit the shift-key, so it’s actually “nVaRN[CTRL]-0.999” for $\text{nVaR}_{0.999}$ and “x[CTRL]62” for x^2.

In[∘]:=
```
Grid[
 {
  {Style["Results", FontSize → 18]}, {TableForm[{{nVaRN, nVaRT}, {nCVaRN, nCVaRT}},
    TableHeadings → {{"VaR0.999", "CVaR0.999"}, {"Normal", "StudentT(ν=4)"}}]}
 },
 Frame → All
]
```

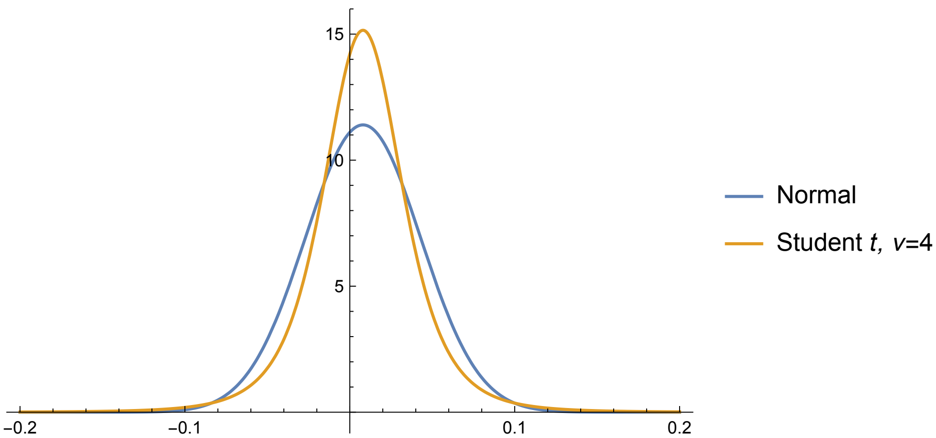
Out[∘]=

Results		
	Normal	StudentT(ν=4)
$VaR_{0.999}$	-0.100158	-0.169527
$CVaR_{0.999}$	-0.109848	-0.231722

The risk measures for a Normal *versus* Student t with $\nu = 4$ are quite different, even though the distributions appear somewhat similar. Also observe that, compared with the Normal, the Student t shows a higher concentration close about the mean but is punctuated by occasional large excursions to the tails.

In[∘]:=
```
Plot[{PDF[NormalDistribution[nPortMu, nPortSigma], r],
  PDF[StudentTDistribution[nPortMu, nTScale, nDegOfFree], r]},
 {r, -0.2, 0.2}, PlotLegends → {"Normal", "Student t, ν=4"}, PlotRange → All]
```

Out[∘]=



Note the tail.

In[∘]:=
```
Plot[{PDF[NormalDistribution[nPortMu, nPortSigma], r],
  PDF[StudentTDistribution[nPortMu, nTScale, nDegOfFree], r]},
 {r, -0.2, -0.1}, PlotLegends → {"Normal", "Student t, ν=4"}, PlotRange → All]
```

Out[∘]=