

Conditional Value-at-Risk Models

One of the most widely used risk measures is the Value-at-Risk, defined as the expected loss on a portfolio at a specified confidence level. In other words, VaR is a percentile of a loss distribution. But despite its popularity VaR suffers from well-known limitations: its tendency to underestimate the risk in the (left) tail of the loss distribution and its failure to capture the dynamics of correlation between portfolio components or nonlinearities in the risk characteristics of the underlying assets.

One method of seeking to address these shortcomings is discussed in a previous post *Copulas in Risk Management*. Another approach known as Conditional Value at Risk (CVaR), which seeks to focus on tail risk, is the subject of this post.

VaR is the lower α -percentile of X , a random variable representing the loss distribution:

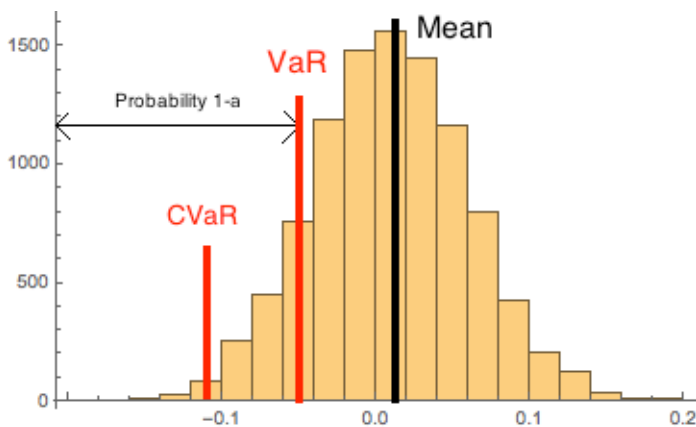
$$\text{VaR}_\alpha(X) = \min\{x \mid F_X(x) \geq \alpha\}$$

CVaR is an alternative percentile measure of risk with conditional interpretation – it is the conditional expectation of the loss that exceeds VaR:

$$\text{CVaR}_\alpha(X) = E[X \mid X \geq \text{VaR}_\alpha(X)].$$

In other words, it is the expected loss, given that the loss exceeds the VaR limit $\text{VaR}_\alpha(X)$.

Histogram[RandomVariate[NormalDistribution[0.01, 0.05], 10 000], ImageSize → Medium]



The Gaussian Model

In a Gaussian framework the VaR is defined by:

$$\text{VaR}_{\text{normal}} = \text{Refine}[\text{InverseCDF}[\text{NormalDistribution}[\mu, \sigma], \alpha], 0 < \alpha < 1] // \text{Simplify}$$
$$\mu - \sqrt{2} \sigma \text{InverseErfc}[2 \alpha]$$

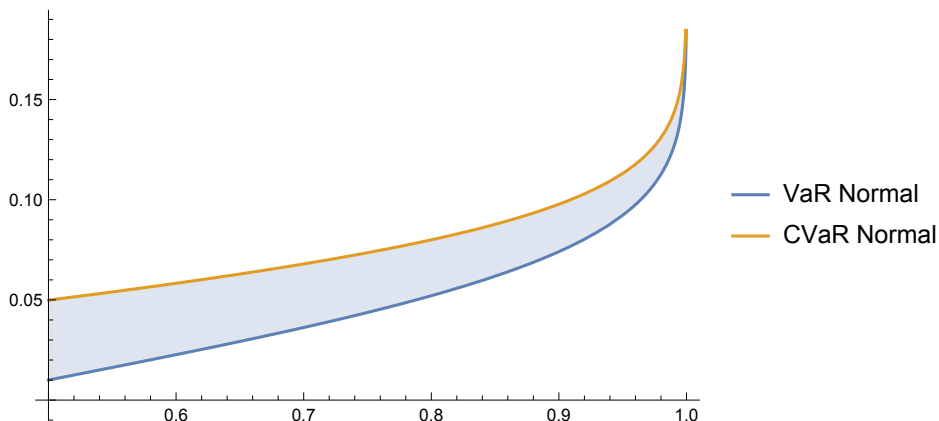
While the CVaR is given by :

```
CVaRnormal = Expectation[x | x > VaRnormal, x ≈ NormalDistribution[μ, σ]] // Simplify
```

$$\frac{2\mu - 2\alpha\mu + e^{-\text{InverseErfc}[2\alpha]^2} \sqrt{\frac{2}{\pi}} \sigma}{2 - 2\alpha}$$

Both VaR and CVaR are nonlinear functions of the confidence level α , rising to a maximum as α approaches 1. Since it measures risk conditionally, in the tail, CVaR always exceeds VaR, as illustrated in the chart below.

```
Plot[{VaRnormal /. {μ → 0.01, σ → 0.05}, CVaRnormal /. {μ → 0.01, σ → 0.05}},
  {α, 0.5, 1}, PlotLegends → {"VaR Normal", "CVaR Normal"}, Filling → {1 → {2}}]
```



Non-Gaussian Models

The normal VaR model, and hence also the CVaR model, tends to underestimate the risk of loss, due to the heaviness of the tails of the loss distribution, which are typically “fatter” than modeled in a Gaussian framework. One approach to addressing this shortcoming is to substitute an alternative distribution that allows for greater weight in the tails.

The **Student-T** distribution is an obvious choice, but while this yields a tractable formula for the VaR, the CVaR has to be evaluated numerically:

```
VaRstudent =
```

```
Refine[InverseCDF[StudentTDistribution[μ, σ, ν], α], 0 < α < 1/2] // Simplify
```

$$\mu - \sqrt{\nu} \sigma \sqrt{-1 + \frac{1}{\text{InverseBetaRegularized}\left[2\alpha, \frac{\nu}{2}, \frac{1}{2}\right]}}$$

```
CVaRstudent =
```

```
Expectation[x | x > VaRstudent, x ≈ StudentTDistribution[μ, σ, ν]] // Simplify
```

$$\text{Expectation}\left[x \mid x + \sqrt{\nu} \sigma \sqrt{-1 + \frac{1}{\text{InverseBetaRegularized}\left[2\alpha, \frac{\nu}{2}, \frac{1}{2}\right]}} > \mu,\right.$$

```
x ≈ StudentTDistribution[μ, σ, ν]
```

The **Logistic** distribution, another heavy-tailed distribution, yields tractable analytic results for both VaR and CVaR:

```
VaRlogistic = Refine[InverseCDF[LogisticDistribution[μ, σ], α], 0 < α < 1] // Simplify
```

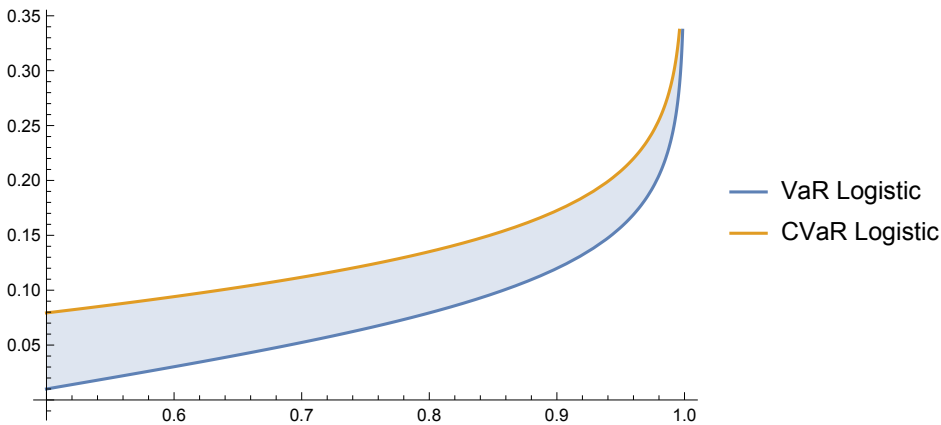
$$\mu - \sigma \operatorname{Log}\left[-1 + \frac{1}{\alpha}\right]$$

```
CVaRlogistic =
```

```
Expectation[x | x > VaRlogistic, x ≈ LogisticDistribution[μ, σ]] // Simplify
```

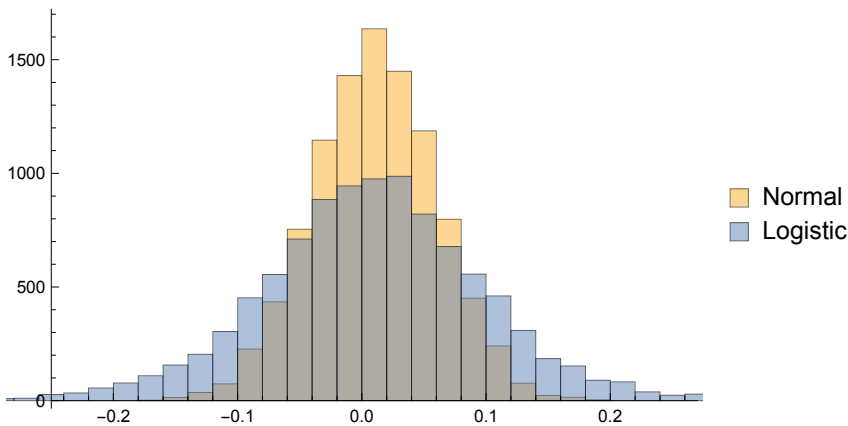
$$\frac{\mu - \alpha \mu + \sigma \operatorname{Log}\left[\frac{1}{1-\alpha}\right] + \alpha \sigma \operatorname{Log}[1 - \alpha] + \alpha \sigma \operatorname{Log}\left[\frac{1}{\alpha}\right]}{-1 + \alpha}$$

```
Plot[{VaRlogistic /. {μ → 0.01, σ → 0.05}, CVaRlogistic /. {μ → 0.01, σ → 0.05}},
     {α, 0.5, 1}, PlotLegends → {"VaR Logistic", "CVaR Logistic"}, Filling → {1 → {2}}]
```

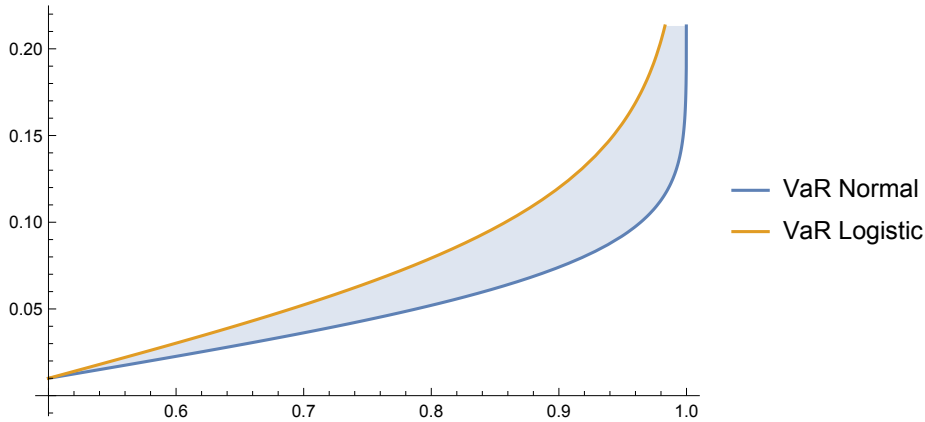


The VaR estimated for a Logistic loss distribution exceeds that for the Gaussian distribution due to its “fat tails”. The same is also true for the CVaR.

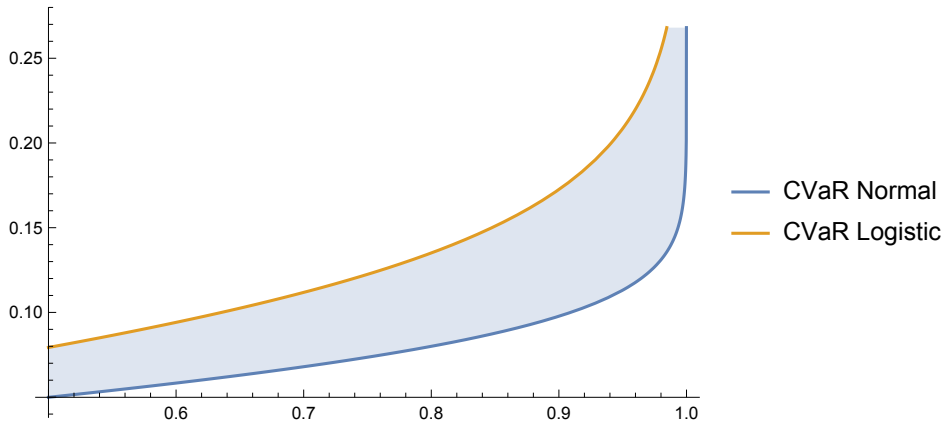
```
Histogram[{RandomVariate[NormalDistribution[0.01, 0.05], 10 000],
           RandomVariate[LogisticDistribution[0.01, 0.05], 10 000]},
          ChartLegends → {"Normal", "Logistic"}, ImageSize → Medium]
```



```
Plot[{VaRnormal /. {μ → 0.01, σ → 0.05}, VaRlogistic /. {μ → 0.01, σ → 0.05}},
  {α, 0.5, 1}, PlotLegends → {"VaR Normal", "VaR Logistic"}, Filling → {1 → {2}}]
```



```
Plot[{CVaRnormal /. {μ → 0.01, σ → 0.05}, CVaRlogistic /. {μ → 0.01, σ → 0.05}},
  {α, 0.5, 1}, PlotLegends → {"CVaR Normal", "CVaR Logistic"}, Filling → {1 → {2}}]
```



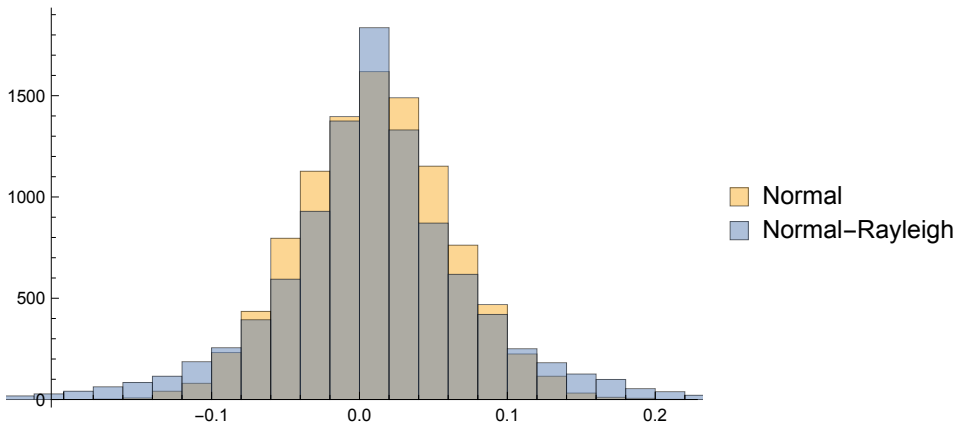
Mixed Density Models

An alternative approach to defining a heavy-tailed loss distribution is to use a mixture of densities. In this framework the loss function follows a normal density, but one in which the the second moment (the volatility) is stochastic. Specifically, we define a mixture distribution for the loss function along the lines of:

```
distnormalrayleigh = ParameterMixtureDistribution[
  NormalDistribution[μ, σ], σ ≈ RayleighDistribution[λ]
  LaplaceDistribution[μ, λ]
```

As with the Non-Gaussian loss function, the mixture distribution has heavier tails than the Gaussian distribution. The mixed density model arguably provides a more realistic fit to the observed empirical data, since it matches the normal density more closely around the mean. Empirically, we tend to find financial assets are more likely to produce close to the mean return, or returns of much larger magnitude, than predicted by the normal distribution.

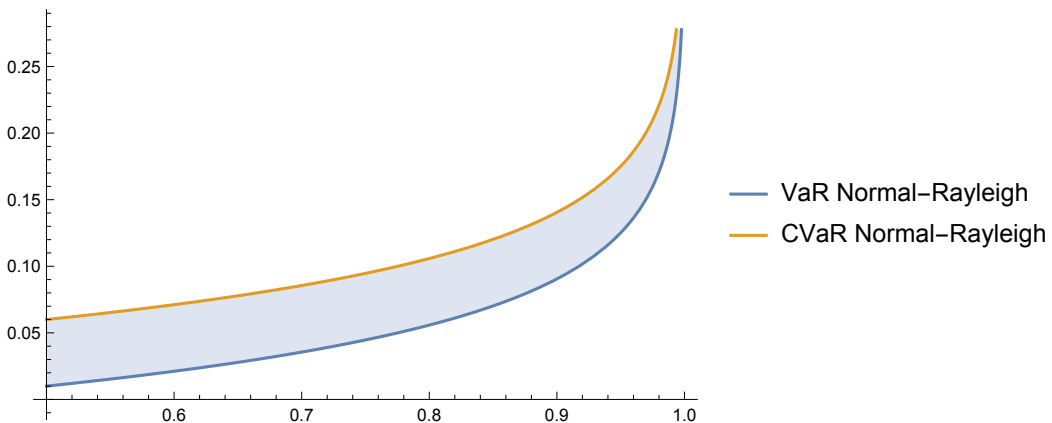
```
Histogram[{RandomVariate[NormalDistribution[0.01, 0.05], 10 000],
  RandomVariate[LaplaceDistribution[0.01, 0.05], 10 000]},
  ChartLegends -> {"Normal", "Normal-Rayleigh"}, ImageSize -> Medium]
```



```
VaRrayleigh =
  Refine[InverseCDF[LaplaceDistribution[μ, λ], α], 1/2 < α < 1] // Simplify
μ - λ Log[2 - 2 α]
```

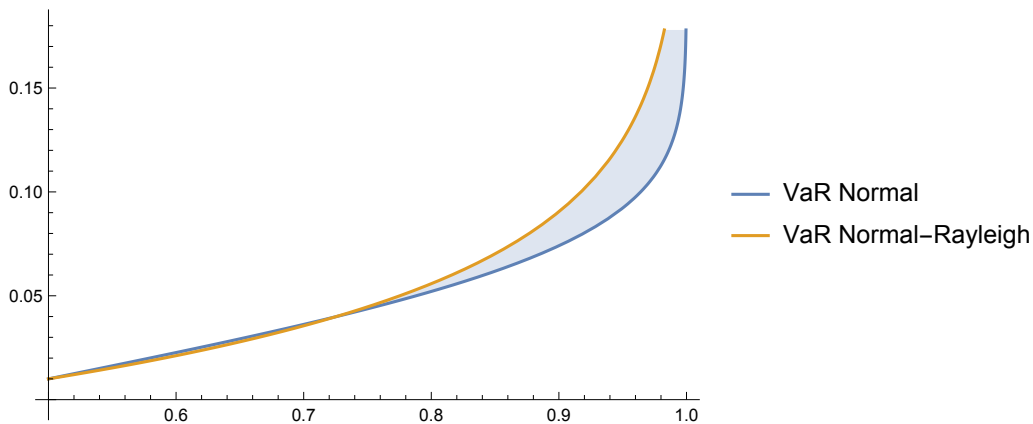
```
CVaRrayleigh =
  Refine[Expectation[x | x > VaRrayleigh, x ≈ LaplaceDistribution[μ, λ],
  λ > 0 && 1/2 < α < 1] // Simplify
λ + μ - λ Log[2 - 2 α]
```

```
Plot[{VaRrayleigh /. {μ -> 0.01, λ -> 0.05}, CVaRrayleigh /. {μ -> 0.01, λ -> 0.05}},
  {α, 0.5, 1}, PlotLegends -> {"VaR Normal-Rayleigh", "CVaR Normal-Rayleigh"},
  Filling -> {1 -> {2}}]
```

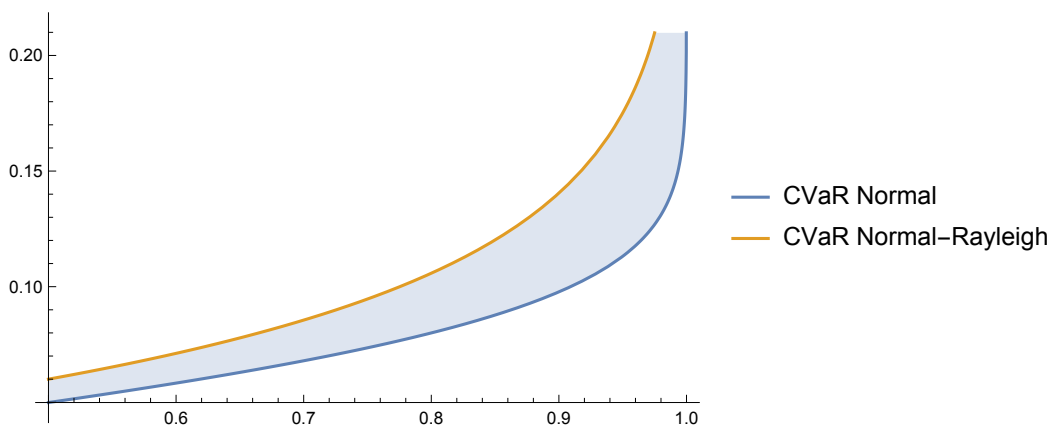


Again, both the VaR and CVaR are elevated under the mixed density framework, compared to the Gaussian model.

```
Plot[{VaRnormal /. {μ → 0.01, σ → 0.05}, VaRrayleigh /. {μ → 0.01, λ → 0.05}}, {α, 0.5, 1}, PlotLegends → {"VaR Normal", "VaR Normal-Rayleigh"}, Filling → {1 → {2}}]
```



```
Plot[{CVaRnormal /. {μ → 0.01, σ → 0.05}, CVaRrayleigh /. {μ → 0.01, λ → 0.05}}, {α, 0.5, 1}, PlotLegends → {"CVaR Normal", "CVaR Normal-Rayleigh"}, Filling → {1 → {2}}]
```





Application to Time Series Models

We can apply the concepts of VaR and CVaR to stochastic processes and thereby assess how the riskiness of an investment asset (or portfolio of assets) fluctuates over time.

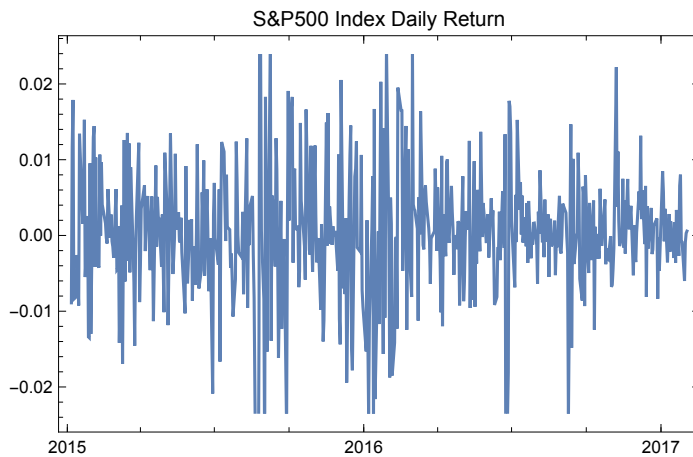
We will use the daily returns in the S&P 500 index as an illustration of the methodology.

```
SP500returns =
```

```
TimeSeriesResample@TimeSeries[FinancialData["^GSPC", "Return", {2015, 01, 05}]]
```


```
TimeSeries[   Time: 06 Jan 2015 to 02 Feb 2017  
Data points: 759 ]
```

```
DateListPlot[SP500returns, PlotLabel → "S&P500 Index Daily Return"]
```

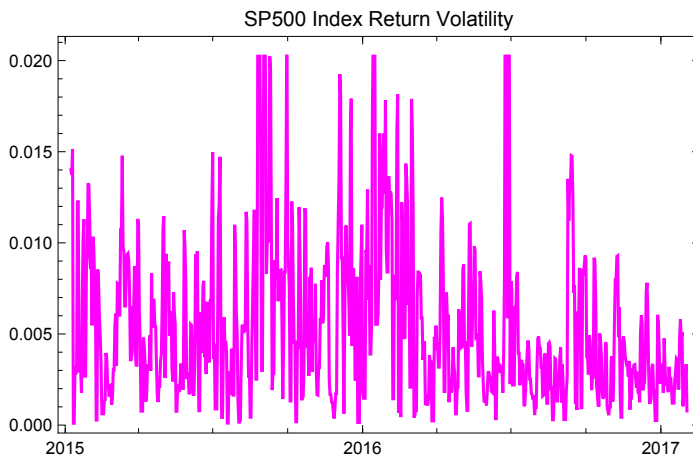


There is clear evidence of conditional heteroskedasticity in the process:

```
realizedvolatility = TimeSeriesResample[
  MovingMap[StandardDeviation, SP500returns, Quantity[2, "Days"]]]
```

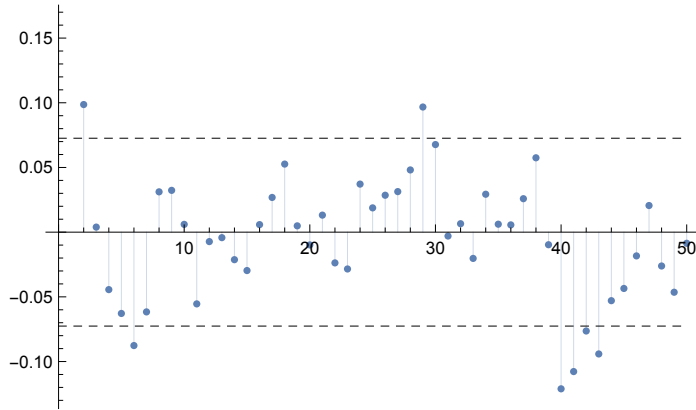
```
TimeSeries [  Time: 08 Jan 2015 to 02 Feb 2017  
Data points: 757 ]
```

```
DateListPlot[realizedvolatility, PlotStyle → Magenta,
  PlotLabel → "SP500 Index Return Volatility"]
```



Statistical portmanteau tests show significant autocorrelations in the returns process:

```
lag = 50;
acf = CorrelationFunction[SP500returns, {lag}];
level = 2 / (Sqrt[SP500returns["PathLength"]]);
ListPlot[acf, Filling -> Axis, Epilog ->
  {Dashed, Line[{{0, level}, {lag, level}}], Line[{{0, -level}, {lag, -level}}]}]
```



```
AutocorrelationTest[SP500returns, Automatic, {"TestDataTable", All}]
```

	Statistic	P-Value
Box-Pierce	107.349	3.25491×10^{-20}
Ljung-Box	107.868	2.54134×10^{-20}

GARCH Model

To address the conditional heteroskedasticity issue, we fit a GARCH model to the returns process:

```
tsm = TimeSeriesModelFit[SP500returns,
  "GARCH", ProcessEstimator -> "MaximumConditionalLikelihood"]
```

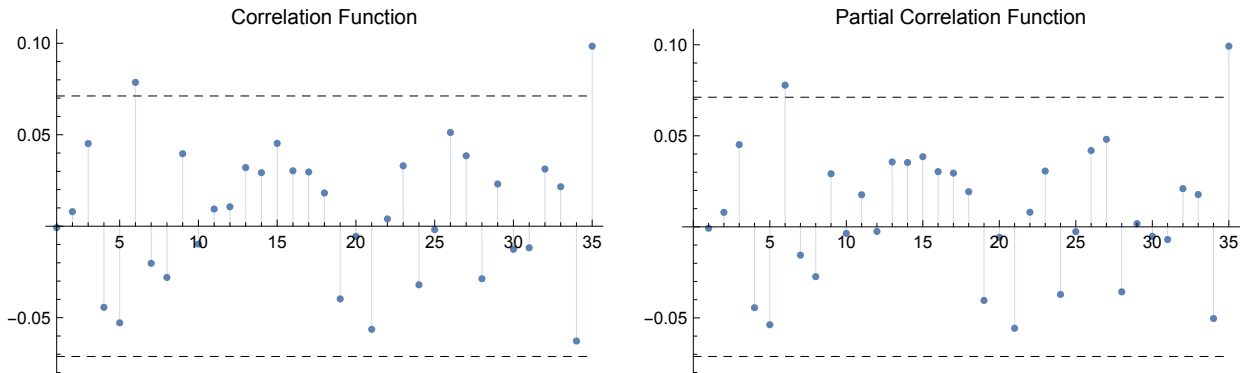
```
TimeSeriesModel[ Family: GARCH  
Order: {1, 1}]
```

```
tsm["ParameterTable"]
```

	Estimate	Standard Error	t-Statistic	P-Value
α_1	0.440072	0.0414096	10.6273	5.49461×10^{-25}
β_1	0.313722	0.0598361	5.24301	1.02535×10^{-7}

The residuals from the fitted GARCH(1,1) model appear well-behaved and there is no evidence of significant patterning in the autocorrelations:


```
GraphicsRow[{tsm["ACFPlot"], tsm["PACFPlot"]}, ImageSize -> Full]
```



```
AutocorrelationTest[tsm["FitResiduals"], Automatic, {"TestDataTable", All}]
```

	Statistic	P-Value
Box-Pierce	10.1936	0.177864
Ljung-Box	10.2892	0.172769

GARCH-VaR and GARCH-CVaR

In a GARCH(1,1) process with parameters $\{\kappa, \alpha, \beta\}$ the conditional variance is given by:

$$\sigma_t^2 = \kappa + \alpha x_{t-1}^2 + \beta \sigma_{t-1}^2$$

In this case the parameter estimates for the SP500 GARCH model are:

```
parameters = List@@ tsm["BestFit"]
{0.0000173108, {0.440139}, {0.313749}}
```

Using this result we can derive the expressions for the GARCH-VaR and GARCH-CVaR risk metrics in the Gaussian framework, as follows:

```
VaRnormalGARCH = VaRnormal /.

```

$$\{\mu \rightarrow 0, \sigma \rightarrow \text{Sqrt}[\text{parameters}[[1]] + \text{parameters}[[2, 1]] x^2 + \text{parameters}[[3, 1]] y^2]\}$$

$$-\sqrt{2} \sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2} \text{InverseErfc}[2 \alpha]$$

```
CVaRnormalGARCH = CVaRnormal /.


```

$$\{\mu \rightarrow 0, \sigma \rightarrow \text{Sqrt}[\text{parameters}[[1]] + \text{parameters}[[2, 1]] x^2 + \text{parameters}[[3, 1]] y^2]\}$$

$$\frac{1}{2 - 2 \alpha} e^{-\text{InverseErfc}[2 \alpha]^2} \sqrt{\frac{2}{\pi}} \sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2}$$

We now combine the returns and volatility series and derive the realized daily VaR and CVaR for the S&P500 series:

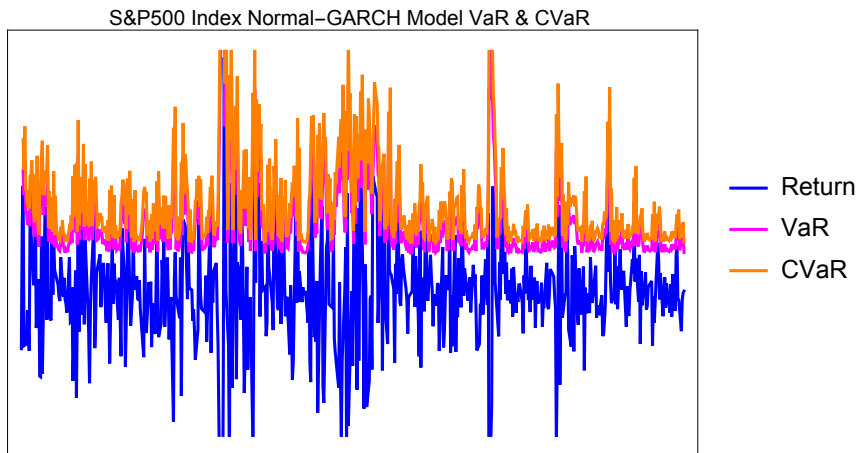
```
returnsvol = TimeSeriesThread[QuantityMagnitude, {SP500returns, realizedvolatility}]
```

```
TimeSeries [  Time: 06 Jan 2015 to 02 Feb 2017  
Data points: 759 ]
```

```

dailyVaRnormalGARCH =
  MovingMap[VaRnormalGARCH /. {α → 0.95, x → First[#], y → Last[#]} &, returnsvol, 2];
dailyCVaRnormalGARCH = MovingMap[
  CVaRnormalGARCH /. {x → First[#], y → Last[#], α → 0.95} &, returnsvol, 2];
DateListPlot[{SP500returns, dailyVaRnormalGARCH["PathComponent", 1],
  dailyCVaRnormalGARCH["PathComponent", 1]},
  PlotStyle → {Blue, Magenta, Orange}, PlotLegends → {"Return", "VaR", "CVaR"},
  PlotLabel → "S&P500 Index Normal-GARCH Model VaR & CVaR ", ImageSize → Medium]

```



Heavy-Tailed GARCH-VaR and GARCH CVaR

We can just as easily estimate VaR and CVaR in a non-Gaussian framework. For example, here are the comparable plots using the Logistic framework. Note that the VaR and CVaR estimates are almost double the size of the Gaussian model estimates.

```
VaRlogisticGARCH = VaRlogistic /.
```

$$\{\mu \rightarrow 0, \sigma \rightarrow \text{Sqrt}[\text{parameters}[[1]] + \text{parameters}[[2, 1]] x^2 + \text{parameters}[[3, 1]] y^2]\}$$

$$-\sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2} \text{Log}\left[-1 + \frac{1}{\alpha}\right]$$

```
CVaRlogisticGARCH = CVaRlogistic /.
```

$$\{\mu \rightarrow 0, \sigma \rightarrow \text{Sqrt}[\text{parameters}[[1]] + \text{parameters}[[2, 1]] x^2 + \text{parameters}[[3, 1]] y^2]\}$$

$$-\frac{1}{-1 + \alpha} \left(\sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2} \text{Log}\left[\frac{1}{1 - \alpha}\right] + \right.$$

$$\left. \sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2} \alpha \text{Log}[1 - \alpha] + \right.$$

$$\left. \sqrt{0.0000173108 + 0.440139 x^2 + 0.313749 y^2} \alpha \text{Log}\left[\frac{1}{\alpha}\right] \right)$$

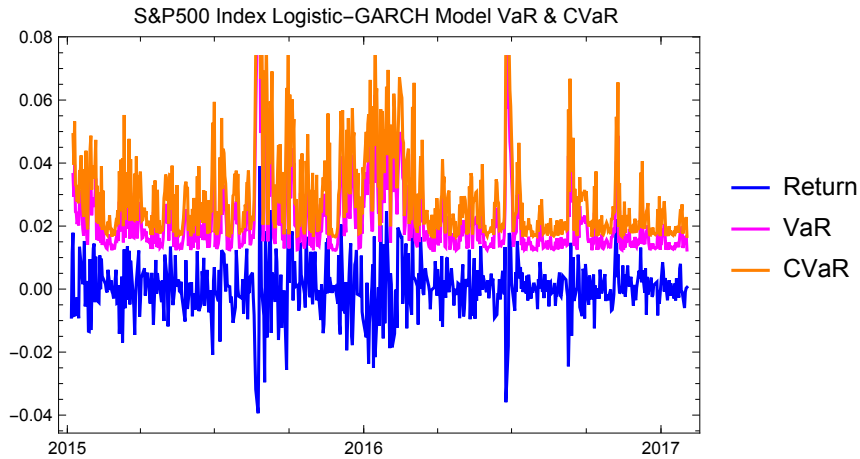
```
dailyVaRlogisticGARCH = MovingMap[
```

```
  VaRlogisticGARCH /. {α → 0.95, x → First[#], y → Last[#]} &, returnsvol, 2];
```

```
dailyCVaRlogisticGARCH = MovingMap[
```

```
  CVaRlogisticGARCH /. {x → First[#], y → Last[#], α → 0.95} &, returnsvol, 2];
```

```
DateListPlot[{SP500returns, dailyVaRlogisticGARCH["PathComponent", 1],
  dailyCVaRlogisticGARCH["PathComponent", 1]},
  PlotStyle -> {Blue, Magenta, Orange}, PlotLegends -> {"Return", "VaR", "CVaR"},
  PlotLabel -> "S&P500 Index Logistic-GARCH Model VaR & CVaR ", ImageSize -> Medium]
```



Conclusion

Rather than explicitly modeling the correlation between portfolio assets as a basis for assessing tail risk, as in the copula approach, Conditional Value at Risk focuses on the tail of the loss distribution. It is straightforward to extend the concept beyond the basic Gaussian framework to model the effect of heavy tails using either a suitable non-Gaussian loss distribution, or a mixture of distributions.

Finally, we have demonstrated how the concept of heavy-tailed Conditional Value at Risk can be combined with non-linear time series models such as GARCH to create a flexible and effective risk management tool.