Week 3: Computing the VaR with R. First steps.

Assume that our loss distribution follows a Normal or Gaussian with mean 0, and standard deviation 1.

This is not really a plausible distribution for credit losses, but nevertheless it is important to understand how to use R in this case.

What is the 95% VaR?

As we have seen, the command is simply qnorm(0.95,0,1), where 0.95 is the alpha level of the VaR, and 0 and 1 are respectively the mean and the standard deviation of our normal distribution.
qnorm stands for "quantile of a normal distribution".

In the case of a standard Gaussian, with mean 0 and standard deviation 1, the information about the mean and the standard deviation can be omitted.

For all the other normal distributions, the information about the mean and the standard deviation is essential and cannot be ignored!

So, here we can see some examples.

Assume now that we want to compute the VaR for an empirical historical distribution of losses.

In R Studio it is very simple to import a dataset.

Just click on "Import Dataset" and choose the location of your file, which can be local or online.

As soon as you have selected your file, you can import it in R Studio.

Let's use the dataset losses.txt, which you can also find here below.

When we import a dataset, we can select different options, such as the name we want to assign to the data in R Studio. Here we choose "data".

We can also tell R Studio that there is a header, which kind of separator is used in the data, and so on.

In our case the separator is the semicolon.

Once we have imported the data, we can have a look at them in the inspector and we can plot them.

Since we are considering a loss distribution, let's produce a histogram.

Notice that I am using the dollar syntax, in order to tell R Studio that I want to select the column named "losses". In this case, since we only have one column, this syntax is a little redundant, but it is always good to be precise in typing our commands.

What is the 95% VaR for this loss distribution?

Here we can use the quantile function, which computes empirical quantiles.

The syntax is rather simple. We type quantile and then we provide the name of the data we are using and the alpha level in decimals.

This is what we get: 6.6796.

In computing the quantiles, R uses some interpolation algorithms. This implies that the quantile we get could not be physically present in the data, but rather an approximation.

Just consider our data, we have a total of 1504 data points. Assume we have ordered them, as you did in the R Exercise number 3 last week. The 95% of 1504 is 1428.8...but observation 1428.8 cannot exist, since this number is not an integer, hence it cannot be the position of an observation. We can only have observation number 1428 or observation number 1429. This is why R interpolates.

If we want to select the nearest existing data point, in statistical terms the closest order statistics, which satisfies the alpha level we have chosen, we can use the option type and select the number 3. In that case the result is slightly different: 6.6817.

For more information about the options of the quantile function, you can always refer to the help. Just type ? quantile.

Is our VaR correct?

When we work with actual data, it is almost impossible to obtain exact values, we usually obtain approximations.

A simple command involving the "which" function (type ? which) tells us that only 76 observations lie above 6.6817, our 95% VaR.

76 observations correspond to roughly 5.05% of all the observations we have in our data sets, that is 1504. This is a rather satisfactory approximation for our 1 minus alpha, that must be equal to 5%.

If instead of R Studio, you use R, nothing really changes. In this case you can import your data by using many different commands. A simple one is represented by read.table.

Again, type ? read.table for more info.

Or check the videos in the extra materials of Week 2.