MATH60633A – TP1

Victor Liu & Fang Si Tang

2024-02-29

PART 1 - Static VaR estimation

```
#1) Load data

load("indices.rda")
prices <- prices["2005-01-01/"]
rets <- CalculateReturns(prices, method = "log")
rets <- rets[-1,]

#Vectorize returns for code efficiency
sp500_rets <- as.numeric(as.character(unlist(rets$SP500)))
ftse100_rets <- as.numeric(as.character(unlist(rets$FTSE100)))

#2) Compute the next-step ahead static VaR forecast with a 95% level confidence

level=0.95
f_forecast_var(sp500_rets[1:1000], level)$VaR_Forecast #SP500 VaR
```

```
## [1] 7.246424e-05
```

```
f_forecast_var(ftse100_rets[1:1000], level)$VaR_Forecast #FTSE100 VaR
```

```
## [1] 6.186707e-05
```

Since the static VaR estimation for the SP500 is larger than that of FTSE100, it is the riskier index at the T+1 horizon.

Part 2.1 - Rolling VaR estimation

```
#1) Backtesting

VaR_SP500 <- rep(NA,1000)
VaR_FTSE100 <- rep(NA,1000)

for (i in 1 : 1000) {
```

```
  #Create rolling windows
  window_SP500 <- sp500_rets[i : (1000 + i)]
  window_FTSE100 <- ftse100_rets[i : (1000 + i)]

  #Get VaR for rolling windows and store in their lists
  VaR_SP500[i] <- f_forecast_var(window_SP500, level)$VaR_Forecast
  VaR_FTSE100[i] <- f_forecast_var(window_FTSE100, level)$VaR_Forecast
}
```

*For code with progress tracker, see main.r

Part 2.2 - Plotting the results

```
#1) Display results (using GGPlot)

#Set variables for x & y axis

realized_rets <- CalculateReturns(prices)
dates <- as.Date(index(realized_rets), "%Y-%m-%d")
x <- dates[1001:2000]
y1 <- realized_rets$SP500[1001:2000]
y2 <- -VaR_SP500
y3 <- -VaR_FTSE100

df_SP500 <- data.frame(x, y1, y2)
df_FTSE100 <- data.frame(x, y1, y3)

#2) Create subplot for SP500

plot1 <- ggplot(df_SP500, aes(x = x)) +
  geom_line(aes(y = y1, color = "Realized Returns")) +
  geom_line(aes(y = y2 * 1000, color = "VaR")) +
  scale_y_continuous(
    name = "Realized Returns",
    sec.axis = sec_axis(~./1000, name = "VaR")
  ) +
  labs(color = "Lines") +
  ggtitle("Realized returns and VaR estimates for SP500")+
  xlab("Time")+
  theme_minimal()

#3) Create subplot for FTSE100

plot2 <- ggplot(df_FTSE100, aes(x = x)) +
  geom_line(aes(y = y1, color = "Realized Returns")) +
  geom_line(aes(y = y3 * 1000, color = "VaR")) +
  scale_y_continuous(
    name = "Realized Returns",
    sec.axis = sec_axis(~./1000, name = "VaR")) +
  labs(color = "Lines") +
```

```
  ggtitle("Realized returns and VaR estimates for FTSE100")+
  xlab("Time")+
  theme_minimal()

grid.arrange(plot1, plot2, ncol = 1)
```