montecarlo

EE

2024-10-07

```
library(quantmod)
```

```
## Warning: package 'quantmod' was built under R version 4.2.3
```

```
## Loading required package: xts
```

```
## Warning: package 'xts' was built under R version 4.2.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.2.3
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: TTR
```

```
## Warning: package 'TTR' was built under R version 4.2.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 4.2.2
```

```r
#from Yahoo take an example stock
getSymbols("AAPL", src = "yahoo", from = "2015-01-01", to = "2020-01-01")
```

```
## [1] "AAPL"
```

```r
prices <- Cl(AAPL)
head(prices)
```

```
##            AAPL.Close
## 2015-01-02    27.3325
## 2015-01-05    26.5625
## 2015-01-06    26.5650
## 2015-01-07    26.9375
## 2015-01-08    27.9725
## 2015-01-09    28.0025
```

```r
#compared to previous day
returns <- dailyReturn(prices)

#threshold to determine significance
limit <- 0.01

#categorize by quality
states <- ifelse(returns > limit, "Up", ifelse(returns < -limit,
                                               "Down", "Stable"))

#add it to df

price_data  <- data.frame(Date = index(prices), Price = coredata(prices), Returns = returns, State = sta

#now all in one
colnames(price_data)[c(2,3,4)] <- c("Price", "Returns", "State")
head(price_data)
```

```
##                  Date   Price       Returns  State
## 2015-01-02 2015-01-02 27.3325  0.000000e+00 Stable
## 2015-01-05 2015-01-05 26.5625 -2.817161e-02   Down
## 2015-01-06 2015-01-06 26.5650  9.413775e-05 Stable
## 2015-01-07 2015-01-07 26.9375  1.402219e-02     Up
## 2015-01-08 2015-01-08 27.9725  3.842227e-02     Up
## 2015-01-09 2015-01-09 28.0025  1.072506e-03 Stable
```

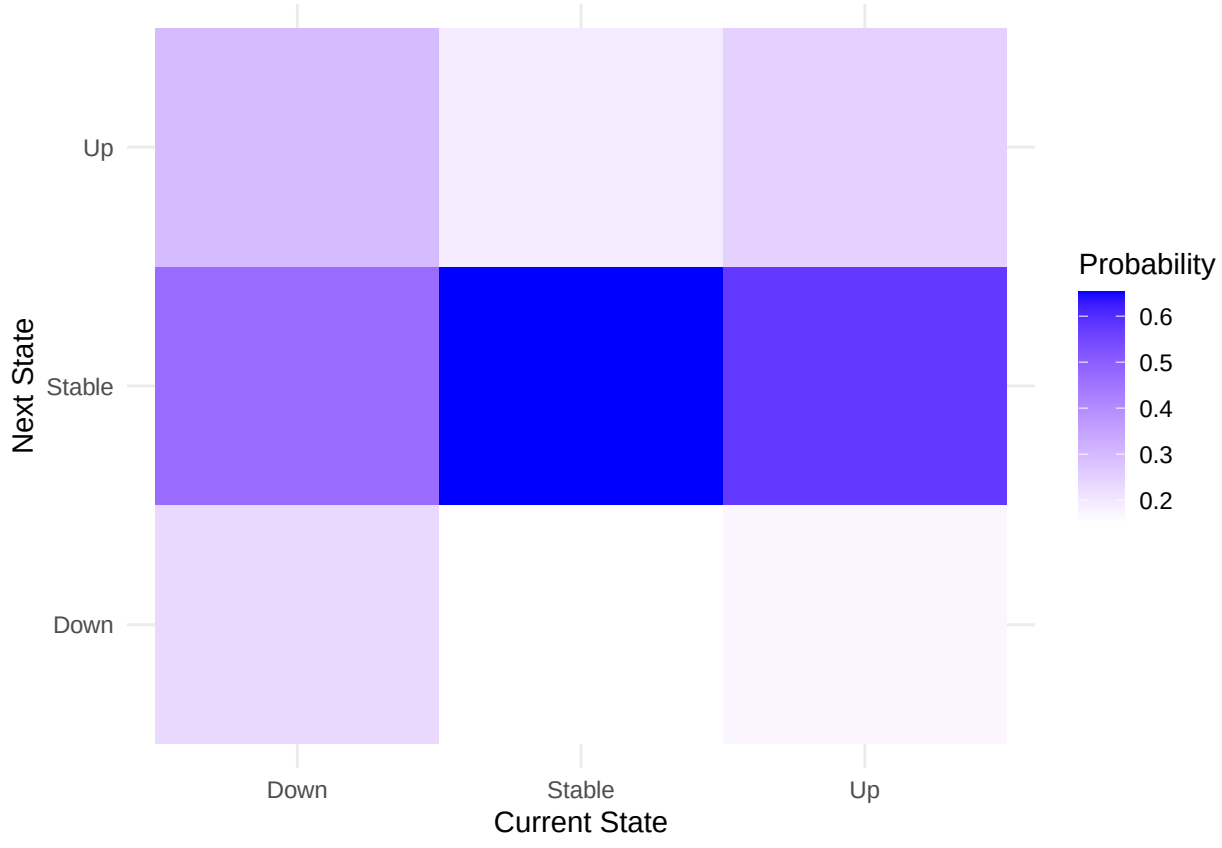
```r
#naturally we need the transition matrix now for Markov chain
price_data$NextState <- c(price_data$State[-1], NA)
#last row gone
price_data <- na.omit(price_data)


#freq  table
transition_count <- table(price_data$State, price_data$NextState)
#transition matrix rows adding up to one as per probability requirements
transition_matrix <- prop.table(transition_count, 1)
transition_matrix
```

```
##
##                Down    Stable        Up
##   Down   0.2325581 0.4697674 0.2976744
##   Stable 0.1528327 0.6521739 0.1949934
##   Up     0.1731449 0.5759717 0.2508834
```

```r
#verify
sums <- rowSums(transition_matrix)
all(sums == 1)
```

```
## [1] TRUE
```

```r
#quick visuals for the matrix
transition_df <- melt(transition_matrix)


#heatmap best fit
ggplot(transition_df, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "blue") +
  labs(x = "Current State", y = "Next State", fill = "Probability") +
  theme_minimal()
```



```r
#moving onto Monte Carlo simulation with basis of our transition matrix

#single price path
simulate_price_path <- function(initial_price, initial_state, transition_matrix, n_days, up_pct = 0.01,
  price <- initial_price
  state <- initial_state
  price_path<- numeric(n_days)
  price_path[1] <- price

  #state names
  states<- rownames(transition_matrix)

  for(day in 2:n_days){

    #next state based on current (respective probs)
    next_state <- sample(states, 1, prob = transition_matrix[state,])

    #adjust price
    if(next_state == "Up"){
      price <- price* (1 + up_pct)

    }
    else if(next_state == "Down") {
      price <- price * (1-down_pct)

    }
    #updated price
    price_path[day] <- price
    #update state
    state <- next_state
  }
  return(price_path)
}

#Example simulation


initial_price <- tail(price_data$Price,1)
initial_state <- tail(price_data$State, 1)
n_days <- 365

simulated_path <- simulate_price_path(initial_price, initial_state, transition_matrix, n_days)

simulated_path
```

```
##   [1] 72.88000 72.88000 72.88000 73.60880 72.87271 72.87271 72.87271 73.60144
##   [9] 74.33745 74.33745 74.33745 74.33745 74.33745 74.33745 75.08083 75.08083
##  [17] 74.33002 74.33002 73.58672 74.32258 74.32258 73.57936 74.31515 75.05830
##  [25] 75.05830 74.30772 75.05080 75.05080 75.05080 75.05080 75.05080 75.05080
##  [33] 74.30029 75.04329 75.04329 75.04329 74.29286 73.54993 73.54993 73.54993
##  [41] 74.28543 75.02828 75.02828 74.27800 75.02078 75.02078 75.02078 74.27057
##  [49] 75.01328 74.26315 75.00578 75.00578 75.75584 75.75584 75.75584 75.75584
##  [57] 75.75584 75.75584 76.51339 76.51339 76.51339 76.51339 76.51339 75.74826
```

```
##  [65] 75.74826 75.74826 76.50574 76.50574 75.74069 75.74069 75.74069 75.74069
##  [73] 74.98328 75.73311 75.73311 74.97578 75.72554 76.48279 77.24762 76.47515
##  [81] 77.23990 77.23990 78.01230 78.01230 77.23217 76.45985 76.45985 77.22445
##  [89] 77.22445 77.99669 77.99669 77.99669 78.77666 77.98889 78.76878 79.55647
##  [97] 78.76091 78.76091 78.76091 78.76091 78.76091 78.76091 78.76091 79.54852
## [105] 79.54852 78.75303 79.54056 79.54056 79.54056 79.54056 80.33597 80.33597
## [113] 81.13933 81.13933 80.32793 80.32793 81.13121 81.13121 81.13121 81.13121
## [121] 81.13121 81.13121 81.13121 80.31990 81.12310 81.12310 81.12310 81.12310
## [129] 81.12310 81.93433 82.75367 83.58121 84.41702 85.26119 85.26119 85.26119
## [137] 86.11380 86.97494 86.10519 86.96624 86.96624 86.96624 86.96624 86.09658
## [145] 86.09658 86.09658 86.09658 86.09658 85.23562 85.23562 84.38326 85.22709
## [153] 85.22709 84.37482 84.37482 84.37482 84.37482 83.53107 83.53107 83.53107
## [161] 83.53107 83.53107 82.69576 81.86881 82.68749 81.86062 81.86062 81.04201
## [169] 80.23159 80.23159 80.23159 79.42928 79.42928 79.42928 79.42928 79.42928
## [177] 79.42928 79.42928 79.42928 79.42928 79.42928 78.63498 79.42133 79.42133
## [185] 79.42133 79.42133 79.42133 80.21555 79.41339 80.20753 80.20753 80.20753
## [193] 80.20753 79.40545 80.19950 80.19950 81.00150 81.00150 81.81151 81.81151
## [201] 81.81151 81.81151 80.99340 80.99340 80.99340 80.99340 80.99340 81.80333
## [209] 80.98530 80.98530 80.98530 81.79515 81.79515 81.79515 81.79515 81.79515
## [217] 81.79515 81.79515 82.61310 82.61310 82.61310 82.61310 83.43924 83.43924
## [225] 83.43924 83.43924 84.27363 83.43089 82.59658 82.59658 82.59658 82.59658
## [233] 82.59658 83.42255 83.42255 82.58832 82.58832 83.41421 83.41421 83.41421
## [241] 83.41421 82.58006 82.58006 82.58006 82.58006 83.40586 83.40586 83.40586
## [249] 84.23992 84.23992 84.23992 85.08232 85.08232 85.08232 85.93315 85.93315
## [257] 86.79248 86.79248 86.79248 86.79248 86.79248 85.92455 85.92455 85.92455
## [265] 85.92455 85.92455 85.92455 85.92455 85.92455 85.92455 85.06531 85.91596
## [273] 85.05680 85.05680 85.05680 84.20623 83.36417 83.36417 83.36417 83.36417
## [281] 83.36417 83.36417 83.36417 83.36417 83.36417 82.53053 82.53053 82.53053
## [289] 82.53053 82.53053 82.53053 83.35583 84.18939 84.18939 83.34750 82.51402
## [297] 83.33916 83.33916 83.33916 84.17256 84.17256 84.17256 84.17256 84.17256
## [305] 83.33083 84.16414 84.16414 85.00578 85.00578 85.00578 85.00578 85.85584
## [313] 85.85584 85.85584 86.71440 86.71440 86.71440 86.71440 86.71440 86.71440
## [321] 86.71440 86.71440 86.71440 86.71440 87.58154 88.45735 89.34193 89.34193
## [329] 89.34193 89.34193 90.23535 89.33299 89.33299 89.33299 89.33299 89.33299
## [337] 90.22632 91.12859 92.03987 92.03987 92.03987 92.03987 92.03987 92.96027
## [345] 92.96027 92.96027 92.96027 92.03067 91.11036 91.11036 91.11036 90.19926
## [353] 89.29727 88.40429 89.28834 89.28834 88.39545 88.39545 88.39545 89.27941
## [361] 89.27941 89.27941 89.27941 89.27941 89.27941
```

```r
#now this will run for multiple simulations not only one
run_monte_carlo<- function(initial_price, initial_state, transition_matrix, n_days, n_simulations, up_p

  #stores all
  all_sim <- matrix(NA, nrow = n_days, ncol = n_simulations)


  #run
  for(i in 1:n_simulations){
    all_sim[, i]<- simulate_price_path(initial_price, initial_state, transition_matrix, n_days, up_pct,
  }

  return(all_sim)

}
```

```r
#example run (1000)

n_simulations <- 1000
all_sim <- run_monte_carlo(initial_price, initial_state, transition_matrix, n_days, n_simulations)

head(all_sim)
```

```
##          [,1]     [,2]  [,3]     [,4]     [,5]     [,6]     [,7]     [,8]
## [1,] 72.88000 72.88000 72.88 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.88000 72.88000 72.88 72.15120 72.88000 73.60880 73.60880 72.15120
## [3,] 73.60880 72.15120 72.88 71.42969 72.88000 72.87271 74.34489 71.42969
## [4,] 72.87271 72.87271 72.88 72.14398 72.15120 72.87271 74.34489 70.71539
## [5,] 72.14398 72.87271 72.88 71.42254 71.42969 72.87271 75.08833 70.71539
## [6,] 71.42254 73.60144 72.88 72.13677 70.71539 72.87271 75.08833 70.71539
##          [,9]   [,10]    [,11]    [,12]    [,13]    [,14] [,15]    [,16]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000 72.88 72.88000
## [2,] 73.60880 72.8800 72.88000 72.15120 72.88000 72.88000 72.88 72.88000
## [3,] 72.87271 72.8800 72.88000 72.87271 72.88000 72.88000 72.88 73.60880
## [4,] 72.87271 72.8800 72.88000 72.87271 72.88000 72.88000 72.88 73.60880
## [5,] 72.87271 72.8800 72.15120 73.60144 73.60880 72.15120 72.88 73.60880
## [6,] 72.87271 72.1512 72.87271 72.86542 74.34489 71.42969 72.88 74.34489
##         [,17] [,18]   [,19]   [,20] [,21] [,22]    [,23]   [,24]   [,25]
## [1,] 72.88000 72.88 72.8800 72.8800 72.88 72.88 72.88000 72.8800 72.8800
## [2,] 72.88000 72.88 72.8800 72.8800 72.88 72.88 73.60880 72.8800 72.1512
## [3,] 72.88000 72.88 72.8800 72.8800 72.88 72.88 73.60880 72.8800 72.1512
## [4,] 72.88000 72.88 72.8800 72.8800 72.88 72.88 73.60880 72.8800 72.1512
## [5,] 73.60880 72.88 72.8800 72.8800 72.88 72.88 72.87271 72.8800 72.1512
## [6,] 72.87271 72.88 73.6088 72.1512 72.88 72.88 72.87271 73.6088 72.1512
##        [,26]   [,27]    [,28]    [,29] [,30] [,31]    [,32]    [,33] [,34]
## [1,] 72.8800 72.8800 72.88000 72.88000 72.88 72.88 72.88000 72.88000 72.88
## [2,] 73.6088 72.8800 72.88000 72.15120 72.88 72.88 72.88000 72.15120 72.88
## [3,] 73.6088 72.8800 73.60880 72.15120 72.88 72.88 73.60880 72.15120 72.88
## [4,] 73.6088 73.6088 72.87271 72.87271 72.88 72.88 74.34489 72.15120 72.88
## [5,] 73.6088 73.6088 72.14398 72.87271 72.88 72.88 74.34489 72.15120 72.88
## [6,] 73.6088 73.6088 72.14398 73.60144 72.88 72.88 74.34489 72.87271 72.88
##         [,35] [,36] [,37]    [,38]    [,39]    [,40]   [,41]   [,42]    [,43]
## [1,] 72.88000 72.88 72.88 72.88000 72.88000 72.88000 72.8800 72.8800 72.88000
## [2,] 72.88000 72.88 72.88 72.88000 72.88000 72.88000 73.6088 72.8800 73.60880
## [3,] 73.60880 72.88 72.88 72.88000 72.15120 73.60880 73.6088 72.8800 74.34489
## [4,] 73.60880 72.88 72.88 73.60880 72.87271 73.60880 73.6088 72.1512 75.08833
## [5,] 73.60880 72.88 72.88 74.34489 72.87271 73.60880 73.6088 72.1512 74.33745
## [6,] 74.34489 72.88 72.88 74.34489 72.87271 74.34489 73.6088 72.1512 74.33745
##         [,44]    [,45] [,46]    [,47] [,48]   [,49]   [,50] [,51]    [,52]
## [1,] 72.88000 72.88000 72.88 72.88000 72.88 72.8800 72.8800 72.88 72.88000
## [2,] 73.60880 72.88000 72.88 73.60880 72.88 73.6088 72.8800 72.88 72.88000
## [3,] 74.34489 72.88000 72.88 73.60880 72.88 73.6088 73.6088 72.88 73.60880
## [4,] 74.34489 73.60880 72.88 74.34489 72.88 73.6088 73.6088 72.88 74.34489
## [5,] 74.34489 74.34489 72.88 74.34489 72.88 73.6088 73.6088 72.88 74.34489
## [6,] 75.08833 74.34489 72.88 74.34489 72.88 73.6088 73.6088 72.88 75.08833
##         [,53]   [,54]    [,55]    [,56]    [,57]    [,58]    [,59]    [,60]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.88000 72.8800 72.88000 72.88000 72.88000 73.60880 73.60880 72.88000
```

```
## [3,] 73.60880 72.8800 72.15120 72.15120 72.15120 73.60880 72.87271 73.60880
## [4,] 73.60880 72.8800 72.15120 72.87271 72.87271 73.60880 72.14398 73.60880
## [5,] 72.87271 73.6088 71.42969 73.60144 72.87271 74.34489 71.42254 73.60880
## [6,] 72.87271 73.6088 72.14398 73.60144 72.87271 74.34489 71.42254 72.87271
##        [,61]   [,62] [,63] [,64]    [,65]    [,66]    [,67]    [,68]    [,69]
## [1,] 72.88000 72.8800 72.88 72.88 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 73.60880 72.8800 72.88 72.88 72.88000 72.88000 72.88000 73.60880 72.88000
## [3,] 73.60880 73.6088 72.88 72.88 72.88000 73.60880 72.88000 74.34489 72.15120
## [4,] 73.60880 73.6088 72.88 72.88 72.88000 72.87271 72.15120 74.34489 72.15120
## [5,] 72.87271 73.6088 72.88 72.88 72.15120 72.87271 72.15120 75.08833 72.15120
## [6,] 72.87271 73.6088 72.88 72.88 72.87271 72.87271 72.87271 75.08833 72.87271
##        [,70]    [,71]    [,72]    [,73]   [,74]   [,75]    [,76]    [,77]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800 72.88000 72.88000
## [2,] 72.15120 72.15120 72.88000 72.88000 72.8800 72.8800 73.60880 72.15120
## [3,] 72.15120 72.15120 72.15120 72.15120 72.8800 73.6088 73.60880 71.42969
## [4,] 71.42969 72.15120 71.42969 72.15120 72.1512 73.6088 72.87271 71.42969
## [5,] 70.71539 72.87271 72.14398 72.87271 72.1512 73.6088 73.60144 72.14398
## [6,] 70.71539 73.60144 72.86542 72.87271 72.1512 73.6088 73.60144 72.14398
##        [,78]    [,79]    [,80]    [,81]    [,82]    [,83]   [,84]    [,85]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000
## [2,] 73.60880 72.88000 72.15120 72.88000 72.88000 72.15120 72.8800 72.88000
## [3,] 73.60880 72.88000 71.42969 72.88000 73.60880 72.15120 72.8800 72.88000
## [4,] 74.34489 73.60880 71.42969 72.88000 73.60880 71.42969 73.6088 73.60880
## [5,] 74.34489 72.87271 70.71539 72.15120 73.60880 71.42969 73.6088 73.60880
## [6,] 74.34489 72.87271 70.00823 71.42969 72.87271 71.42969 73.6088 72.87271
##        [,86]   [,87]    [,88]   [,89]    [,90]   [,91] [,92]    [,93]    [,94]
## [1,] 72.88000 72.8800 72.88000 72.8800 72.88000 72.8800 72.88 72.88000 72.88000
## [2,] 72.88000 72.8800 72.88000 72.8800 73.60880 72.8800 72.88 72.15120 72.15120
## [3,] 72.88000 73.6088 72.88000 73.6088 74.34489 72.8800 72.88 72.15120 72.15120
## [4,] 72.88000 73.6088 72.15120 73.6088 74.34489 72.8800 72.88 72.15120 71.42969
## [5,] 72.15120 73.6088 72.87271 73.6088 74.34489 72.8800 72.88 72.15120 72.14398
## [6,] 72.87271 73.6088 72.87271 73.6088 74.34489 72.1512 72.88 71.42969 72.14398
##        [,95]    [,96] [,97]    [,98]    [,99]   [,100]  [,101] [,102]   [,103]
## [1,] 72.88000 72.88000 72.88 72.88000 72.88000 72.88000 72.8800  72.88 72.88000
## [2,] 72.88000 72.88000 72.88 73.60880 73.60880 72.88000 72.8800  72.88 72.88000
## [3,] 72.88000 72.88000 72.88 72.87271 73.60880 73.60880 72.8800  72.88 72.88000
## [4,] 72.15120 73.60880 72.88 72.87271 72.87271 72.87271 72.8800  72.88 72.15120
## [5,] 72.87271 73.60880 72.88 72.87271 73.60144 72.87271 73.6088  72.88 72.15120
## [6,] 73.60144 74.34489 72.88 73.60144 73.60144 72.87271 73.6088  72.88 72.87271
##       [,104]   [,105]   [,106]   [,107]   [,108]   [,109]   [,110]  [,111]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800
## [2,] 72.15120 72.15120 73.60880 72.88000 72.15120 72.88000 73.60880 72.8800
## [3,] 71.42969 71.42969 74.34489 72.88000 71.42969 73.60880 73.60880 72.8800
## [4,] 70.71539 71.42969 74.34489 73.60880 71.42969 74.34489 74.34489 72.1512
## [5,] 70.71539 71.42969 73.60144 73.60880 72.14398 75.08833 74.34489 72.1512
## [6,] 70.71539 70.71539 73.60144 74.34489 72.14398 75.08833 74.34489 72.1512
##      [,112]   [,113]  [,114]  [,115] [,116]   [,117]   [,118] [,119]   [,120]
## [1,]  72.88 72.88000 72.8800 72.8800  72.88 72.88000 72.88000  72.88 72.88000
## [2,]  72.88 72.15120 72.8800 73.6088  72.88 72.88000 72.88000  72.88 72.88000
## [3,]  72.88 72.87271 72.8800 73.6088  72.88 72.15120 73.60880  72.88 72.15120
## [4,]  72.88 72.87271 73.6088 73.6088  72.88 72.87271 72.87271  72.88 72.87271
## [5,]  72.88 72.87271 73.6088 73.6088  72.88 73.60144 72.14398  72.88 72.87271
## [6,]  72.88 72.87271 73.6088 73.6088  72.88 73.60144 72.14398  72.88 72.87271
##       [,121]   [,122]   [,123]  [,124]   [,125] [,126]  [,127]   [,128] [,129]
```

```
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000  72.88 72.8800 72.88000  72.88
## [2,] 73.60880 72.88000 73.60880 72.8800 72.15120  72.88 72.8800 73.60880  72.88
## [3,] 73.60880 72.88000 73.60880 72.1512 72.87271  72.88 72.8800 74.34489  72.88
## [4,] 73.60880 72.15120 72.87271 72.1512 72.87271  72.88 72.8800 75.08833  72.88
## [5,] 72.87271 72.15120 72.87271 72.1512 72.14398  72.88 72.8800 75.08833  72.88
## [6,] 73.60144 71.42969 72.87271 72.1512 72.86542  72.88 73.6088 74.33745  72.88
##       [,130]   [,131]   [,132]  [,133]   [,134]   [,135]  [,136]   [,137]
## [1,] 72.8800 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.88000
## [2,] 73.6088 72.88000 73.60880 72.8800 72.88000 72.88000 72.8800 73.60880
## [3,] 73.6088 72.88000 73.60880 72.8800 72.88000 73.60880 72.1512 74.34489
## [4,] 73.6088 73.60880 74.34489 72.8800 72.88000 74.34489 72.1512 74.34489
## [5,] 73.6088 74.34489 74.34489 72.8800 73.60880 74.34489 72.1512 74.34489
## [6,] 73.6088 73.60144 74.34489 73.6088 72.87271 74.34489 72.1512 74.34489
##      [,138]   [,139] [,140]   [,141]  [,142]   [,143]  [,144]   [,145]   [,146]
## [1,]  72.88 72.88000  72.88 72.88000 72.8800 72.88000 72.8800 72.88000 72.88000
## [2,]  72.88 72.88000  72.88 72.88000 72.8800 72.88000 73.6088 72.88000 72.88000
## [3,]  72.88 72.88000  72.88 72.88000 72.8800 73.60880 73.6088 72.15120 72.15120
## [4,]  72.88 73.60880  72.88 73.60880 72.8800 73.60880 73.6088 72.87271 72.87271
## [5,]  72.88 72.87271  72.88 72.87271 73.6088 73.60880 73.6088 72.87271 72.87271
## [6,]  72.88 72.87271  72.88 72.87271 73.6088 74.34489 73.6088 72.87271 72.87271
##       [,147]  [,148]   [,149] [,150]   [,151]   [,152]   [,153]  [,154]
## [1,] 72.88000 72.8800 72.88000  72.88 72.88000 72.88000 72.88000 72.8800
## [2,] 72.88000 72.8800 73.60880  72.88 73.60880 72.15120 72.88000 72.8800
## [3,] 72.88000 72.8800 72.87271  72.88 73.60880 72.15120 73.60880 72.8800
## [4,] 72.88000 72.8800 72.14398  72.88 74.34489 72.15120 73.60880 72.8800
## [5,] 73.60880 73.6088 72.86542  72.88 74.34489 71.42969 73.60880 72.8800
## [6,] 72.87271 73.6088 72.86542  72.88 73.60144 70.71539 72.87271 73.6088
##       [,155]   [,156] [,157]   [,158] [,159]   [,160] [,161]   [,162]  [,163]
## [1,] 72.88000 72.88000  72.88 72.88000  72.88 72.88000  72.88 72.88000 72.8800
## [2,] 72.88000 72.88000  72.88 73.60880  72.88 72.88000  72.88 72.88000 72.8800
## [3,] 72.88000 72.15120  72.88 73.60880  72.88 72.88000  72.88 72.15120 72.8800
## [4,] 72.15120 72.15120  72.88 73.60880  72.88 72.15120  72.88 72.15120 73.6088
## [5,] 71.42969 72.15120  72.88 74.34489  72.88 72.87271  72.88 72.15120 73.6088
## [6,] 70.71539 72.87271  72.88 74.34489  72.88 73.60144  72.88 72.87271 73.6088
##       [,164]   [,165]   [,166]   [,167]   [,168]   [,169]   [,170]   [,171]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800 72.88000 73.60880 73.60880 72.88000 72.88000 72.15120 72.88000
## [3,] 72.8800 72.15120 72.87271 73.60880 72.15120 73.60880 71.42969 72.15120
## [4,] 72.8800 71.42969 72.87271 73.60880 72.15120 73.60880 72.14398 72.15120
## [5,] 72.8800 70.71539 72.14398 73.60880 72.15120 73.60880 71.42254 72.15120
## [6,] 72.1512 71.42254 72.14398 74.34489 72.87271 72.87271 71.42254 72.87271
##       [,172]   [,173] [,174] [,175]   [,176]  [,177]  [,178]  [,179]   [,180]
## [1,] 72.88000 72.88000  72.88  72.88 72.88000 72.8800 72.8800 72.8800 72.88000
## [2,] 73.60880 73.60880  72.88  72.88 73.60880 72.8800 73.6088 72.8800 72.15120
## [3,] 72.87271 73.60880  72.88  72.88 73.60880 72.8800 73.6088 72.1512 71.42969
## [4,] 73.60144 74.34489  72.88  72.88 73.60880 72.8800 73.6088 72.1512 72.14398
## [5,] 73.60144 74.34489  72.88  72.88 72.87271 72.8800 73.6088 72.1512 72.14398
## [6,] 73.60144 75.08833  72.88  72.88 72.14398 73.6088 73.6088 72.1512 72.86542
##       [,181]   [,182]   [,183]   [,184]   [,185]   [,186]  [,187]  [,188]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800
## [2,] 72.8800 72.15120 72.88000 72.88000 72.88000 72.15120 72.1512 72.8800
## [3,] 72.8800 72.15120 72.15120 73.60880 73.60880 72.15120 72.1512 72.8800
## [4,] 73.6088 72.15120 71.42969 72.87271 73.60880 72.87271 72.1512 73.6088
## [5,] 73.6088 72.87271 72.14398 73.60144 72.87271 72.87271 72.1512 73.6088
```

```
## [6,] 73.6088 73.60144 72.86542 73.60144 72.87271 72.14398 72.1512 73.6088
##        [,189]  [,190]   [,191] [,192]  [,193] [,194]   [,195]  [,196]   [,197]
## [1,] 72.88000 72.8800 72.88000  72.88 72.8800  72.88 72.88000 72.8800 72.88000
## [2,] 72.88000 72.8800 73.60880  72.88 72.8800  72.88 72.88000 72.1512 72.15120
## [3,] 73.60880 73.6088 74.34489  72.88 72.8800  72.88 72.88000 72.1512 72.15120
## [4,] 73.60880 73.6088 74.34489  72.88 73.6088  72.88 72.88000 72.1512 72.87271
## [5,] 73.60880 73.6088 75.08833  72.88 73.6088  72.88 72.15120 72.1512 73.60144
## [6,] 74.34489 73.6088 75.08833  72.88 73.6088  72.88 71.42969 72.1512 74.33745
##        [,198]   [,199]   [,200]   [,201] [,202]  [,203]   [,204] [,205]
## [1,] 72.88000 72.88000 72.88000 72.88000  72.88 72.8800 72.88000  72.88
## [2,] 73.60880 72.15120 73.60880 73.60880  72.88 72.8800 72.88000  72.88
## [3,] 74.34489 72.15120 72.87271 72.87271  72.88 72.8800 72.88000  72.88
## [4,] 74.34489 72.87271 73.60144 72.87271  72.88 72.8800 72.88000  72.88
## [5,] 73.60144 72.87271 73.60144 72.87271  72.88 72.1512 73.60880  72.88
## [6,] 73.60144 72.14398 73.60144 72.87271  72.88 72.1512 74.34489  72.88
##        [,206]   [,207]   [,208] [,209]   [,210] [,211]   [,212]   [,213] [,214]
## [1,] 72.88000 72.88000 72.88000  72.88 72.88000  72.88 72.88000 72.88000  72.88
## [2,] 72.15120 72.15120 72.15120  72.88 72.88000  72.88 72.88000 72.88000  72.88
## [3,] 72.15120 72.87271 72.15120  72.88 72.15120  72.88 72.88000 72.88000  72.88
## [4,] 72.15120 72.87271 72.15120  72.88 72.15120  72.88 73.60880 72.88000  72.88
## [5,] 72.15120 72.87271 72.87271  72.88 72.87271  72.88 72.87271 73.60880  72.88
## [6,] 72.87271 73.60144 72.87271  72.88 72.87271  72.88 73.60144 74.34489  72.88
##        [,215] [,216]   [,217]  [,218]   [,219]   [,220]   [,221]   [,222]
## [1,] 72.88000  72.88 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 72.15120  72.88 72.15120 72.8800 72.88000 72.88000 73.60880 72.88000
## [3,] 72.15120  72.88 72.15120 72.8800 72.88000 72.15120 72.87271 72.15120
## [4,] 72.15120  72.88 72.15120 72.8800 73.60880 72.15120 73.60144 71.42969
## [5,] 72.15120  72.88 72.15120 73.6088 74.34489 72.87271 74.33745 71.42969
## [6,] 72.87271  72.88 72.87271 73.6088 74.34489 72.87271 74.33745 72.14398
##       [,223]   [,224] [,225] [,226]  [,227]   [,228]   [,229]   [,230]   [,231]
## [1,] 72.8800 72.88000  72.88  72.88 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800 72.15120  72.88  72.88 72.8800 72.88000 72.88000 72.15120 73.60880
## [3,] 72.8800 72.15120  72.88  72.88 72.8800 73.60880 73.60880 71.42969 73.60880
## [4,] 72.8800 72.87271  72.88  72.88 73.6088 73.60880 74.34489 72.14398 74.34489
## [5,] 72.8800 73.60144  72.88  72.88 73.6088 74.34489 75.08833 71.42254 75.08833
## [6,] 72.1512 73.60144  72.88  72.88 73.6088 74.34489 75.08833 70.70832 75.08833
##       [,232]  [,233]   [,234]  [,235]   [,236]   [,237]   [,238] [,239]
## [1,] 72.8800 72.8800 72.88000 72.8800 72.88000 72.88000 72.88000  72.88
## [2,] 72.8800 72.8800 72.88000 72.8800 72.15120 72.15120 73.60880  72.88
## [3,] 72.8800 72.1512 72.15120 72.8800 71.42969 72.15120 73.60880  72.88
## [4,] 72.8800 72.1512 71.42969 72.8800 71.42969 72.15120 73.60880  72.88
## [5,] 73.6088 72.1512 70.71539 72.8800 71.42969 72.87271 74.34489  72.88
## [6,] 73.6088 72.1512 71.42254 73.6088 71.42969 73.60144 73.60144  72.88
##        [,240]  [,241]   [,242]   [,243]  [,244]  [,245]  [,246]   [,247]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.8800 72.8800 72.8800 72.88000
## [2,] 72.88000 72.1512 72.88000 72.88000 72.8800 72.8800 72.8800 72.88000
## [3,] 72.88000 72.1512 72.88000 72.15120 72.8800 72.8800 72.8800 72.88000
## [4,] 73.60880 72.1512 73.60880 72.87271 72.8800 72.1512 72.8800 72.88000
## [5,] 74.34489 72.1512 72.87271 72.87271 72.8800 72.1512 72.8800 73.60880
## [6,] 73.60144 72.1512 72.87271 72.87271 73.6088 72.1512 73.6088 72.87271
##        [,248]   [,249]   [,250]   [,251]   [,252]  [,253]   [,254]  [,255]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800
## [2,] 72.88000 72.88000 72.88000 72.15120 73.60880 72.8800 72.88000 73.6088
## [3,] 72.88000 73.60880 73.60880 72.15120 73.60880 72.8800 72.15120 73.6088
```

```
## [4,] 73.60880 72.87271 74.34489 72.15120 73.60880 72.8800 72.15120 73.6088
## [5,] 73.60880 72.87271 75.08833 72.15120 73.60880 72.8800 72.87271 73.6088
## [6,] 72.87271 72.87271 75.83922 71.42969 74.34489 72.1512 72.87271 73.6088
##       [,256]   [,257]   [,258]   [,259]   [,260]  [,261]  [,262]   [,263]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800 72.88000
## [2,] 72.8800 72.15120 73.60880 72.88000 72.15120 73.6088 72.8800 72.88000
## [3,] 72.8800 72.15120 74.34489 72.88000 72.87271 73.6088 72.8800 72.15120
## [4,] 72.8800 72.15120 74.34489 73.60880 72.87271 73.6088 72.8800 72.15120
## [5,] 72.8800 72.87271 74.34489 72.87271 72.87271 73.6088 72.8800 72.15120
## [6,] 72.1512 73.60144 74.34489 72.87271 72.87271 73.6088 72.1512 71.42969
##       [,264]  [,265]  [,266]   [,267]   [,268]   [,269]   [,270]  [,271]
## [1,] 72.88000 72.8800 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800
## [2,] 72.88000 72.8800 72.8800 72.88000 72.88000 72.88000 73.60880 72.8800
## [3,] 72.88000 72.8800 72.8800 72.15120 72.88000 73.60880 73.60880 72.8800
## [4,] 73.60880 72.8800 72.8800 71.42969 73.60880 73.60880 73.60880 73.6088
## [5,] 73.60880 73.6088 73.6088 72.14398 73.60880 74.34489 73.60880 73.6088
## [6,] 72.87271 73.6088 73.6088 72.14398 74.34489 74.34489 72.87271 73.6088
##       [,272]   [,273]  [,274]   [,275]   [,276]  [,277]   [,278]  [,279]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.88000 72.8800
## [2,] 73.60880 72.15120 72.8800 72.88000 72.88000 73.6088 73.60880 73.6088
## [3,] 72.87271 72.87271 72.8800 72.88000 72.15120 73.6088 72.87271 73.6088
## [4,] 72.87271 72.87271 72.8800 72.15120 72.15120 73.6088 72.87271 73.6088
## [5,] 72.87271 72.87271 72.8800 71.42969 72.87271 73.6088 72.87271 73.6088
## [6,] 72.87271 73.60144 72.1512 70.71539 72.14398 73.6088 72.87271 73.6088
##       [,280]  [,281]  [,282]   [,283]   [,284]   [,285]   [,286]  [,287]
## [1,] 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800
## [2,] 72.8800 73.6088 72.8800 72.15120 72.15120 72.88000 72.15120 72.8800
## [3,] 72.8800 73.6088 72.8800 72.15120 72.15120 72.88000 72.87271 72.8800
## [4,] 72.8800 73.6088 72.8800 71.42969 71.42969 72.15120 72.14398 72.8800
## [5,] 72.8800 73.6088 72.1512 71.42969 72.14398 71.42969 72.14398 73.6088
## [6,] 73.6088 73.6088 72.1512 72.14398 72.14398 71.42969 72.14398 73.6088
##       [,288] [,289]  [,290]   [,291]   [,292]   [,293]   [,294]  [,295]
## [1,] 72.88000  72.88 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800
## [2,] 72.88000  72.88 72.8800 72.88000 72.88000 72.88000 72.15120 73.6088
## [3,] 72.15120  72.88 72.8800 72.88000 72.15120 72.88000 71.42969 73.6088
## [4,] 72.15120  72.88 72.8800 73.60880 71.42969 72.88000 71.42969 73.6088
## [5,] 71.42969  72.88 72.8800 73.60880 71.42969 73.60880 70.71539 73.6088
## [6,] 72.14398  72.88 72.1512 74.34489 71.42969 74.34489 70.71539 73.6088
##       [,296]   [,297]   [,298]   [,299] [,300]  [,301]   [,302]   [,303]
## [1,] 72.88000 72.88000 72.88000 72.88000  72.88 72.8800 72.88000 72.88000
## [2,] 72.88000 72.88000 73.60880 72.88000  72.88 72.8800 73.60880 73.60880
## [3,] 73.60880 72.15120 73.60880 72.88000  72.88 72.8800 73.60880 74.34489
## [4,] 73.60880 72.87271 73.60880 72.15120  72.88 72.8800 72.87271 75.08833
## [5,] 74.34489 72.14398 72.87271 72.87271  72.88 73.6088 72.87271 75.08833
## [6,] 74.34489 72.86542 73.60144 72.87271  72.88 73.6088 72.87271 75.83922
##       [,304]   [,305] [,306]   [,307]  [,308]  [,309]  [,310]   [,311] [,312]
## [1,] 72.8800 72.88000  72.88 72.88000 72.8800 72.8800 72.8800 72.88000  72.88
## [2,] 72.8800 72.88000  72.88 72.88000 72.1512 72.8800 72.1512 73.60880  72.88
## [3,] 73.6088 73.60880  72.88 72.88000 72.1512 72.8800 72.1512 73.60880  72.88
## [4,] 73.6088 74.34489  72.88 73.60880 72.1512 73.6088 72.1512 73.60880  72.88
## [5,] 73.6088 75.08833  72.88 73.60880 72.1512 73.6088 72.1512 73.60880  72.88
## [6,] 73.6088 75.83922  72.88 74.34489 72.1512 73.6088 72.1512 72.87271  72.88
##       [,313]  [,314]  [,315]  [,316]  [,317]  [,318]   [,319]  [,320]   [,321]
## [1,] 72.88000 72.8800 72.8800 72.8800 72.8800 72.8800 72.88000 72.8800 72.88000
```

```
## [2,] 72.88000 72.8800 72.8800 72.8800 72.8800 72.8800 73.60880 72.8800 72.15120
## [3,] 73.60880 73.6088 72.8800 72.8800 72.8800 72.1512 72.87271 72.8800 72.87271
## [4,] 72.87271 73.6088 72.1512 72.8800 72.8800 72.1512 72.87271 72.1512 73.60144
## [5,] 73.60144 73.6088 72.1512 73.6088 73.6088 72.1512 72.14398 72.1512 73.60144
## [6,] 73.60144 73.6088 72.1512 73.6088 73.6088 72.1512 72.86542 72.1512 72.86542
##       [,322]   [,323]  [,324]  [,325]  [,326]   [,327]   [,328]   [,329]
## [1,] 72.8800 72.88000 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000
## [2,] 72.8800 72.88000 72.8800 72.8800 72.8800 72.15120 72.88000 72.88000
## [3,] 72.8800 73.60880 73.6088 72.8800 72.8800 72.15120 72.88000 72.88000
## [4,] 72.8800 74.34489 73.6088 72.8800 72.8800 72.87271 72.15120 72.88000
## [5,] 72.8800 75.08833 73.6088 72.8800 72.8800 73.60144 72.87271 72.15120
## [6,] 73.6088 75.83922 73.6088 72.1512 73.6088 72.86542 72.14398 72.87271
##       [,330]   [,331]   [,332]  [,333]   [,334]  [,335]   [,336]   [,337]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800 72.88000 72.88000
## [2,] 72.15120 72.15120 72.88000 72.8800 72.15120 72.8800 72.15120 72.88000
## [3,] 71.42969 72.87271 73.60880 72.8800 72.15120 72.8800 72.15120 72.88000
## [4,] 71.42969 72.87271 72.87271 72.8800 72.87271 72.8800 71.42969 72.88000
## [5,] 71.42969 73.60144 72.14398 72.1512 72.87271 72.8800 70.71539 72.15120
## [6,] 71.42969 73.60144 72.86542 72.1512 73.60144 72.1512 70.71539 71.42969
##       [,338]   [,339]   [,340]   [,341] [,342]   [,343]   [,344] [,345]
## [1,] 72.88000 72.88000 72.88000 72.88000  72.88 72.88000 72.88000  72.88
## [2,] 72.88000 72.15120 72.15120 72.88000  72.88 72.88000 73.60880  72.88
## [3,] 72.88000 72.87271 71.42969 72.88000  72.88 72.88000 72.87271  72.88
## [4,] 73.60880 72.87271 71.42969 72.15120  72.88 72.15120 72.87271  72.88
## [5,] 73.60880 72.87271 72.14398 72.15120  72.88 72.87271 73.60144  72.88
## [6,] 72.87271 72.87271 72.14398 72.87271  72.88 72.87271 73.60144  72.88
##       [,346]   [,347]  [,348]  [,349]   [,350]  [,351]   [,352]  [,353] [,354]
## [1,] 72.88000 72.88000 72.8800 72.8800 72.88000 72.8800 72.88000 72.8800  72.88
## [2,] 72.15120 72.15120 72.8800 72.8800 72.88000 72.8800 72.88000 72.8800  72.88
## [3,] 72.15120 72.15120 72.8800 73.6088 72.88000 72.8800 72.88000 72.8800  72.88
## [4,] 71.42969 72.15120 72.8800 73.6088 72.15120 72.8800 72.88000 72.8800  72.88
## [5,] 70.71539 72.15120 72.8800 73.6088 72.87271 72.8800 73.60880 72.8800  72.88
## [6,] 70.71539 72.87271 73.6088 73.6088 72.87271 73.6088 74.34489 73.6088  72.88
##       [,355]   [,356]   [,357]   [,358]   [,359] [,360]   [,361] [,362]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000  72.88 72.88000  72.88
## [2,] 73.60880 72.15120 72.88000 72.15120 72.88000  72.88 73.60880  72.88
## [3,] 73.60880 72.15120 72.88000 72.87271 72.88000  72.88 72.87271  72.88
## [4,] 72.87271 71.42969 72.88000 72.87271 73.60880  72.88 72.14398  72.88
## [5,] 72.87271 72.14398 72.15120 72.87271 73.60880  72.88 72.86542  72.88
## [6,] 72.87271 72.14398 72.87271 73.60144 74.34489  72.88 72.13677  72.88
##       [,363] [,364]  [,365]  [,366]   [,367]   [,368]   [,369]   [,370]
## [1,] 72.88000  72.88 72.8800 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 72.88000  72.88 72.8800 72.8800 72.15120 72.88000 72.88000 72.88000
## [3,] 72.88000  72.88 72.1512 72.8800 72.87271 72.88000 72.88000 73.60880
## [4,] 72.15120  72.88 72.1512 72.8800 72.87271 72.88000 72.88000 74.34489
## [5,] 72.15120  72.88 72.1512 72.8800 72.87271 72.15120 72.15120 75.08833
## [6,] 72.87271  72.88 72.1512 73.6088 72.87271 72.87271 71.42969 75.08833
##       [,371]  [,372]   [,373] [,374]  [,375]  [,376]   [,377]   [,378]
## [1,] 72.88000 72.8800 72.88000  72.88 72.8800 72.8800 72.88000 72.88000
## [2,] 73.60880 72.8800 73.60880  72.88 73.6088 73.6088 72.15120 73.60880
## [3,] 73.60880 72.8800 74.34489  72.88 73.6088 73.6088 72.15120 73.60880
## [4,] 74.34489 72.1512 74.34489  72.88 73.6088 73.6088 71.42969 73.60880
## [5,] 73.60144 72.1512 73.60144  72.88 73.6088 73.6088 70.71539 73.60880
## [6,] 74.33745 72.1512 72.86542  72.88 73.6088 73.6088 70.00823 74.34489
```

```
##        [,379]   [,380]   [,381]  [,382]   [,383]   [,384] [,385] [,386]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000  72.88  72.88
## [2,] 72.88000 72.88000 72.88000 73.6088 72.88000 72.15120  72.88  72.88
## [3,] 73.60880 73.60880 72.88000 73.6088 73.60880 72.15120  72.88  72.88
## [4,] 72.87271 73.60880 73.60880 73.6088 72.87271 72.15120  72.88  72.88
## [5,] 72.87271 72.87271 73.60880 73.6088 72.87271 72.87271  72.88  72.88
## [6,] 72.14398 73.60144 72.87271 73.6088 72.87271 73.60144  72.88  72.88
##        [,387]  [,388]   [,389] [,390]  [,391]   [,392]   [,393] [,394]  [,395]
## [1,] 72.88000 72.8800 72.88000  72.88 72.8800 72.88000 72.88000  72.88 72.8800
## [2,] 72.15120 72.8800 72.15120  72.88 72.8800 72.88000 72.88000  72.88 72.8800
## [3,] 71.42969 72.8800 71.42969  72.88 72.8800 73.60880 72.15120  72.88 72.8800
## [4,] 71.42969 72.8800 72.14398  72.88 72.8800 72.87271 72.87271  72.88 73.6088
## [5,] 72.14398 72.8800 72.86542  72.88 73.6088 73.60144 73.60144  72.88 73.6088
## [6,] 72.14398 73.6088 72.13677  72.88 73.6088 74.33745 73.60144  72.88 73.6088
##       [,396]  [,397]   [,398]   [,399]   [,400] [,401]   [,402]   [,403]
## [1,] 72.8800 72.8800 72.88000 72.88000 72.88000  72.88 72.88000 72.88000
## [2,] 72.8800 72.8800 72.15120 72.88000 72.88000  72.88 73.60880 72.88000
## [3,] 73.6088 73.6088 72.87271 73.60880 72.88000  72.88 74.34489 73.60880
## [4,] 73.6088 73.6088 72.87271 74.34489 72.88000  72.88 75.08833 74.34489
## [5,] 73.6088 73.6088 72.14398 75.08833 72.15120  72.88 75.08833 74.34489
## [6,] 73.6088 73.6088 71.42254 74.33745 71.42969  72.88 75.08833 74.34489
##       [,404]   [,405]   [,406]  [,407]   [,408]   [,409]   [,410] [,411]
## [1,] 72.8800 72.88000 72.88000 72.8800 72.88000 72.88000 72.88000  72.88
## [2,] 72.8800 72.88000 72.15120 72.8800 72.88000 72.88000 72.88000  72.88
## [3,] 72.1512 73.60880 72.15120 72.8800 72.88000 72.15120 72.88000  72.88
## [4,] 72.1512 73.60880 72.87271 72.8800 73.60880 71.42969 73.60880  72.88
## [5,] 72.1512 72.87271 73.60144 72.1512 73.60880 70.71539 74.34489  72.88
## [6,] 72.1512 72.87271 72.86542 72.1512 74.34489 70.71539 74.34489  72.88
##        [,412] [,413]   [,414]   [,415]  [,416]   [,417]  [,418]  [,419]
## [1,] 72.88000  72.88 72.88000 72.88000 72.8800 72.88000 72.8800 72.8800
## [2,] 72.15120  72.88 72.88000 73.60880 73.6088 72.88000 72.8800 72.8800
## [3,] 71.42969  72.88 72.15120 73.60880 73.6088 73.60880 72.8800 72.8800
## [4,] 72.14398  72.88 72.87271 73.60880 73.6088 73.60880 73.6088 72.8800
## [5,] 72.86542  72.88 72.87271 73.60880 73.6088 74.34489 73.6088 72.1512
## [6,] 73.59408  72.88 72.87271 72.87271 73.6088 74.34489 73.6088 72.1512
##        [,420]  [,421]   [,422]   [,423]   [,424]  [,425] [,426]   [,427] [,428]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.8800  72.88 72.88000  72.88
## [2,] 72.15120 72.8800 73.60880 72.15120 72.88000 72.8800  72.88 73.60880  72.88
## [3,] 71.42969 73.6088 74.34489 72.87271 73.60880 72.8800  72.88 73.60880  72.88
## [4,] 71.42969 73.6088 73.60144 72.87271 73.60880 72.8800  72.88 72.87271  72.88
## [5,] 72.14398 73.6088 74.33745 72.87271 72.87271 72.8800  72.88 72.14398  72.88
## [6,] 72.86542 73.6088 74.33745 72.87271 73.60144 72.1512  72.88 72.86542  72.88
##        [,429]   [,430]   [,431]   [,432]   [,433]  [,434]  [,435]   [,436]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800 72.88000
## [2,] 72.88000 73.60880 73.60880 72.88000 73.60880 72.8800 72.8800 72.88000
## [3,] 72.88000 73.60880 72.87271 73.60880 74.34489 72.8800 72.8800 72.88000
## [4,] 73.60880 73.60880 73.60144 73.60880 74.34489 72.1512 73.6088 73.60880
## [5,] 74.34489 74.34489 74.33745 74.34489 74.34489 72.1512 73.6088 72.87271
## [6,] 74.34489 74.34489 73.59408 73.60144 74.34489 72.1512 73.6088 72.87271
##        [,437]   [,438] [,439]  [,440]   [,441]   [,442]   [,443]   [,444]
## [1,] 72.88000 72.88000  72.88 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 73.60880 72.88000  72.88 72.8800 72.88000 72.88000 72.88000 72.88000
## [3,] 73.60880 72.15120  72.88 72.8800 72.88000 73.60880 73.60880 73.60880
## [4,] 72.87271 71.42969  72.88 72.1512 72.15120 72.87271 74.34489 72.87271
```

```
## [5,] 72.87271 71.42969  72.88 72.1512 71.42969 72.87271 74.34489 73.60144
## [6,] 73.60144 71.42969  72.88 72.1512 72.14398 72.87271 74.34489 73.60144
##       [,445]   [,446]  [,447]   [,448]  [,449]  [,450]  [,451] [,452]  [,453]
## [1,] 72.8800 72.88000 72.8800 72.88000 72.8800 72.8800 72.8800  72.88 72.8800
## [2,] 72.8800 72.15120 72.8800 72.88000 72.8800 72.1512 72.8800  72.88 72.8800
## [3,] 72.8800 72.15120 72.8800 72.88000 73.6088 72.1512 72.8800  72.88 72.8800
## [4,] 72.8800 72.87271 73.6088 73.60880 73.6088 72.1512 72.8800  72.88 72.8800
## [5,] 72.1512 72.87271 73.6088 73.60880 73.6088 72.1512 72.8800  72.88 73.6088
## [6,] 72.1512 72.14398 73.6088 74.34489 73.6088 72.1512 72.1512  72.88 73.6088
##       [,454] [,455] [,456]   [,457]  [,458]   [,459] [,460]   [,461] [,462]
## [1,] 72.88000  72.88  72.88 72.88000 72.8800 72.88000  72.88 72.88000  72.88
## [2,] 72.88000  72.88  72.88 72.15120 72.8800 72.15120  72.88 72.88000  72.88
## [3,] 73.60880  72.88  72.88 72.15120 73.6088 72.87271  72.88 72.88000  72.88
## [4,] 73.60880  72.88  72.88 71.42969 73.6088 72.87271  72.88 72.88000  72.88
## [5,] 73.60880  72.88  72.88 70.71539 73.6088 73.60144  72.88 72.15120  72.88
## [6,] 74.34489  72.88  72.88 70.71539 73.6088 73.60144  72.88 71.42969  72.88
##       [,463]   [,464]   [,465]  [,466]   [,467]   [,468] [,469]   [,470]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000  72.88 72.88000
## [2,] 72.88000 72.88000 72.88000 72.8800 73.60880 72.88000  72.88 73.60880
## [3,] 73.60880 73.60880 72.88000 72.8800 73.60880 72.15120  72.88 72.87271
## [4,] 74.34489 73.60880 72.15120 72.8800 73.60880 71.42969  72.88 73.60144
## [5,] 75.08833 74.34489 72.87271 72.8800 74.34489 70.71539  72.88 73.60144
## [6,] 75.08833 75.08833 73.60144 73.6088 75.08833 71.42254  72.88 73.60144
##      [,471]  [,472]   [,473]   [,474]  [,475]   [,476]  [,477]   [,478]
## [1,]  72.88 72.8800 72.88000 72.88000 72.8800 72.88000 72.8800 72.88000
## [2,]  72.88 72.8800 72.88000 73.60880 72.8800 73.60880 72.8800 72.15120
## [3,]  72.88 72.8800 73.60880 73.60880 72.8800 73.60880 72.1512 71.42969
## [4,]  72.88 72.8800 74.34489 74.34489 73.6088 72.87271 72.1512 70.71539
## [5,]  72.88 72.1512 74.34489 75.08833 73.6088 72.87271 72.1512 70.00823
## [6,]  72.88 72.1512 74.34489 75.08833 73.6088 73.60144 72.1512 70.00823
##       [,479]   [,480]  [,481]  [,482]   [,483]   [,484]  [,485]   [,486]
## [1,] 72.88000 72.88000 72.8800 72.8800 72.88000 72.88000 72.8800 72.88000
## [2,] 72.88000 73.60880 73.6088 72.8800 73.60880 72.88000 72.1512 72.15120
## [3,] 73.60880 73.60880 73.6088 72.8800 73.60880 72.88000 72.1512 72.15120
## [4,] 73.60880 73.60880 73.6088 72.8800 72.87271 72.15120 72.1512 71.42969
## [5,] 74.34489 74.34489 73.6088 72.8800 72.14398 71.42969 72.1512 72.14398
## [6,] 75.08833 74.34489 73.6088 72.1512 72.86542 70.71539 72.1512 72.86542
##      [,487]   [,488]   [,489]   [,490] [,491]   [,492]   [,493] [,494]   [,495]
## [1,]  72.88 72.88000 72.88000 72.88000  72.88 72.88000 72.88000  72.88 72.88000
## [2,]  72.88 72.88000 73.60880 72.88000  72.88 72.88000 72.88000  72.88 72.15120
## [3,]  72.88 72.15120 73.60880 73.60880  72.88 73.60880 72.15120  72.88 72.87271
## [4,]  72.88 72.87271 72.87271 74.34489  72.88 74.34489 72.87271  72.88 73.60144
## [5,]  72.88 72.87271 72.14398 74.34489  72.88 74.34489 73.60144  72.88 73.60144
## [6,]  72.88 72.87271 72.86542 74.34489  72.88 74.34489 72.86542  72.88 73.60144
##      [,496]  [,497]  [,498]  [,499]  [,500]   [,501]  [,502]   [,503] [,504]
## [1,]  72.88 72.8800 72.8800 72.8800 72.8800 72.88000 72.8800 72.88000  72.88
## [2,]  72.88 72.8800 73.6088 72.8800 72.8800 72.88000 72.8800 73.60880  72.88
## [3,]  72.88 72.8800 73.6088 72.8800 72.8800 73.60880 72.1512 72.87271  72.88
## [4,]  72.88 72.8800 73.6088 72.8800 73.6088 72.87271 72.1512 72.87271  72.88
## [5,]  72.88 72.8800 73.6088 72.1512 73.6088 72.87271 72.1512 73.60144  72.88
## [6,]  72.88 73.6088 73.6088 72.1512 73.6088 72.87271 72.1512 72.86542  72.88
##       [,505]  [,506] [,507]   [,508]   [,509]  [,510]   [,511]   [,512]
## [1,] 72.88000 72.8800  72.88 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,] 72.88000 72.8800  72.88 72.15120 72.88000 73.6088 72.15120 72.15120
```

```
## [3,] 72.88000 72.8800  72.88 71.42969 72.88000 73.6088 71.42969 72.15120
## [4,] 73.60880 72.8800  72.88 71.42969 72.15120 73.6088 71.42969 72.15120
## [5,] 74.34489 72.8800  72.88 71.42969 72.87271 73.6088 72.14398 71.42969
## [6,] 73.60144 73.6088  72.88 71.42969 72.87271 73.6088 72.86542 71.42969
##        [,513]   [,514]  [,515]   [,516]  [,517]   [,518]   [,519]   [,520]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.8800 72.88000 72.88000 72.88000
## [2,] 72.88000 73.60880 72.8800 72.15120 72.8800 72.15120 73.60880 72.88000
## [3,] 72.88000 74.34489 72.8800 72.87271 72.8800 72.15120 73.60880 72.15120
## [4,] 72.15120 74.34489 72.8800 72.87271 72.8800 72.87271 73.60880 71.42969
## [5,] 72.87271 74.34489 72.8800 72.87271 72.8800 73.60144 73.60880 70.71539
## [6,] 72.87271 75.08833 73.6088 72.87271 72.1512 73.60144 74.34489 70.71539
##        [,521]   [,522]  [,523]   [,524]   [,525]  [,526]  [,527]   [,528]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.8800 72.88000
## [2,] 72.88000 72.88000 72.1512 73.60880 72.15120 72.8800 72.8800 72.15120
## [3,] 72.15120 73.60880 72.1512 73.60880 72.87271 72.8800 73.6088 72.87271
## [4,] 72.15120 74.34489 72.1512 73.60880 72.14398 72.1512 73.6088 72.87271
## [5,] 71.42969 74.34489 72.1512 74.34489 72.14398 72.1512 73.6088 72.87271
## [6,] 71.42969 74.34489 72.1512 74.34489 71.42254 72.1512 73.6088 72.87271
##        [,529] [,530]   [,531]  [,532]   [,533]   [,534]   [,535]   [,536]
## [1,] 72.88000  72.88 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 72.15120  72.88 72.15120 72.8800 72.88000 73.60880 72.88000 73.60880
## [3,] 72.87271  72.88 72.15120 72.8800 72.88000 74.34489 73.60880 73.60880
## [4,] 72.14398  72.88 71.42969 73.6088 72.15120 74.34489 74.34489 73.60880
## [5,] 71.42254  72.88 70.71539 73.6088 71.42969 73.60144 73.60144 73.60880
## [6,] 70.70832  72.88 70.71539 73.6088 72.14398 74.33745 74.33745 72.87271
##       [,537] [,538]  [,539]  [,540]  [,541]  [,542]   [,543]   [,544]   [,545]
## [1,] 72.8800  72.88 72.8800 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000
## [2,] 72.8800  72.88 73.6088 73.6088 72.8800 72.8800 73.60880 72.15120 73.60880
## [3,] 72.8800  72.88 73.6088 73.6088 72.1512 72.8800 73.60880 72.87271 74.34489
## [4,] 72.8800  72.88 73.6088 73.6088 72.1512 72.8800 74.34489 72.87271 74.34489
## [5,] 72.1512  72.88 73.6088 73.6088 72.1512 72.8800 75.08833 72.87271 75.08833
## [6,] 72.1512  72.88 73.6088 73.6088 72.1512 73.6088 75.83922 73.60144 74.33745
##       [,546]   [,547]   [,548]   [,549]   [,550]  [,551]   [,552]   [,553]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,] 73.6088 72.88000 72.88000 72.15120 72.88000 72.8800 72.15120 72.88000
## [3,] 73.6088 72.88000 72.88000 71.42969 72.88000 72.8800 72.87271 72.88000
## [4,] 73.6088 72.15120 72.15120 70.71539 72.88000 73.6088 72.87271 73.60880
## [5,] 73.6088 72.87271 72.87271 71.42254 73.60880 73.6088 72.14398 74.34489
## [6,] 73.6088 72.87271 72.87271 71.42254 74.34489 73.6088 72.14398 73.60144
##      [,554]   [,555] [,556]  [,557] [,558]  [,559]   [,560]   [,561] [,562]
## [1,]  72.88 72.88000  72.88 72.8800  72.88 72.8800 72.88000 72.88000  72.88
## [2,]  72.88 72.15120  72.88 73.6088  72.88 72.8800 73.60880 72.88000  72.88
## [3,]  72.88 71.42969  72.88 73.6088  72.88 72.8800 72.87271 72.15120  72.88
## [4,]  72.88 72.14398  72.88 73.6088  72.88 72.8800 72.87271 72.87271  72.88
## [5,]  72.88 72.14398  72.88 73.6088  72.88 72.8800 72.87271 72.87271  72.88
## [6,]  72.88 71.42254  72.88 73.6088  72.88 73.6088 72.87271 73.60144  72.88
##        [,563]   [,564]   [,565]   [,566]   [,567]   [,568] [,569]  [,570]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000  72.88 72.8800
## [2,] 72.15120 72.88000 72.88000 72.88000 72.88000 72.88000  72.88 72.8800
## [3,] 71.42969 72.15120 72.15120 73.60880 72.15120 72.15120  72.88 72.8800
## [4,] 70.71539 72.15120 72.15120 74.34489 72.87271 72.87271  72.88 72.1512
## [5,] 70.00823 72.15120 72.15120 74.34489 72.87271 72.87271  72.88 72.1512
## [6,] 69.30815 71.42969 72.87271 74.34489 72.87271 72.87271  72.88 72.1512
##        [,571]   [,572]   [,573]   [,574]   [,575]   [,576]   [,577] [,578]
```

```
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000  72.88
## [2,] 73.60880 73.60880 73.60880 72.88000 72.88000 72.88000 72.88000  72.88
## [3,] 73.60880 74.34489 74.34489 73.60880 72.15120 72.88000 72.88000  72.88
## [4,] 73.60880 75.08833 74.34489 73.60880 72.15120 72.88000 73.60880  72.88
## [5,] 72.87271 74.33745 74.34489 73.60880 71.42969 73.60880 73.60880  72.88
## [6,] 72.87271 74.33745 74.34489 74.34489 71.42969 72.87271 72.87271  72.88
##      [,579]   [,580] [,581] [,582]   [,583]   [,584]   [,585]   [,586]  [,587]
## [1,]  72.88 72.88000  72.88  72.88 72.88000 72.88000 72.88000 72.88000 72.8800
## [2,]  72.88 72.88000  72.88  72.88 72.88000 72.88000 72.88000 73.60880 72.8800
## [3,]  72.88 73.60880  72.88  72.88 73.60880 72.15120 72.88000 72.87271 72.8800
## [4,]  72.88 73.60880  72.88  72.88 74.34489 72.87271 72.15120 73.60144 72.8800
## [5,]  72.88 73.60880  72.88  72.88 74.34489 72.87271 71.42969 74.33745 73.6088
## [6,]  72.88 72.87271  72.88  72.88 73.60144 72.87271 71.42969 75.08083 73.6088
##       [,588]   [,589]   [,590] [,591] [,592]   [,593]   [,594] [,595]  [,596]
## [1,] 72.88000 72.88000 72.88000  72.88  72.88 72.88000 72.88000  72.88 72.8800
## [2,] 72.88000 73.60880 72.15120  72.88  72.88 72.15120 72.15120  72.88 72.8800
## [3,] 72.15120 73.60880 72.15120  72.88  72.88 72.87271 72.87271  72.88 72.8800
## [4,] 71.42969 72.87271 72.15120  72.88  72.88 72.87271 72.87271  72.88 72.8800
## [5,] 70.71539 73.60144 72.15120  72.88  72.88 72.87271 72.87271  72.88 72.1512
## [6,] 71.42254 74.33745 71.42969  72.88  72.88 73.60144 72.87271  72.88 72.1512
##       [,597]  [,598] [,599]   [,600]   [,601]  [,602]  [,603]   [,604]
## [1,] 72.88000 72.8800  72.88 72.88000 72.88000 72.8800 72.8800 72.88000
## [2,] 72.88000 72.8800  72.88 72.15120 73.60880 72.8800 72.1512 73.60880
## [3,] 72.88000 72.8800  72.88 72.87271 73.60880 73.6088 72.1512 74.34489
## [4,] 73.60880 72.1512  72.88 72.87271 73.60880 73.6088 72.1512 73.60144
## [5,] 72.87271 72.1512  72.88 73.60144 73.60880 73.6088 72.1512 72.86542
## [6,] 72.14398 72.1512  72.88 73.60144 72.87271 73.6088 72.1512 72.86542
##       [,605]   [,606]  [,607]  [,608] [,609]   [,610]  [,611]   [,612]
## [1,] 72.88000 72.88000 72.8800 72.8800  72.88 72.88000 72.8800 72.88000
## [2,] 72.88000 73.60880 72.8800 72.8800  72.88 72.15120 72.8800 73.60880
## [3,] 72.88000 74.34489 72.8800 72.8800  72.88 72.15120 72.8800 72.87271
## [4,] 73.60880 74.34489 72.8800 73.6088  72.88 72.87271 73.6088 73.60144
## [5,] 73.60880 73.60144 73.6088 73.6088  72.88 72.87271 73.6088 73.60144
## [6,] 72.87271 73.60144 73.6088 73.6088  72.88 72.87271 73.6088 72.86542
##       [,613]  [,614] [,615]   [,616]  [,617] [,618]   [,619]   [,620] [,621]
## [1,] 72.88000 72.8800  72.88 72.88000 72.8800  72.88 72.88000 72.88000  72.88
## [2,] 72.88000 73.6088  72.88 72.88000 72.8800  72.88 73.60880 72.88000  72.88
## [3,] 73.60880 73.6088  72.88 72.88000 72.8800  72.88 74.34489 72.88000  72.88
## [4,] 74.34489 73.6088  72.88 73.60880 72.8800  72.88 75.08833 72.88000  72.88
## [5,] 75.08833 73.6088  72.88 74.34489 72.8800  72.88 75.08833 72.15120  72.88
## [6,] 75.83922 73.6088  72.88 75.08833 72.1512  72.88 75.83922 71.42969  72.88
##       [,622]  [,623]  [,624] [,625] [,626]   [,627]   [,628] [,629]   [,630]
## [1,] 72.88000 72.8800 72.8800  72.88  72.88 72.88000 72.88000  72.88 72.88000
## [2,] 72.88000 72.8800 72.8800  72.88  72.88 72.88000 73.60880  72.88 72.88000
## [3,] 72.88000 72.8800 72.8800  72.88  72.88 72.88000 74.34489  72.88 72.88000
## [4,] 73.60880 72.1512 72.8800  72.88  72.88 72.88000 74.34489  72.88 73.60880
## [5,] 72.87271 72.1512 72.8800  72.88  72.88 73.60880 75.08833  72.88 74.34489
## [6,] 72.87271 72.1512 73.6088  72.88  72.88 74.34489 75.08833  72.88 73.60144
##       [,631]  [,632]   [,633]   [,634]  [,635]  [,636]  [,637]  [,638]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.8800 72.8800 72.8800 72.8800
## [2,] 73.60880 73.6088 73.60880 73.60880 72.1512 73.6088 72.8800 72.8800
## [3,] 73.60880 73.6088 74.34489 73.60880 72.1512 73.6088 72.8800 72.8800
## [4,] 74.34489 73.6088 75.08833 74.34489 72.1512 73.6088 73.6088 72.8800
## [5,] 74.34489 73.6088 75.08833 74.34489 72.1512 73.6088 73.6088 72.8800
```

```
## [6,] 74.34489 73.6088 75.08833 75.08833 72.1512 73.6088 73.6088 73.6088
##        [,639] [,640]  [,641]  [,642]  [,643]   [,644]   [,645]   [,646] [,647]
## [1,] 72.88000  72.88 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000  72.88
## [2,] 72.88000  72.88 72.8800 72.8800 72.8800 72.15120 72.88000 72.88000  72.88
## [3,] 72.88000  72.88 72.8800 73.6088 72.8800 72.15120 72.15120 72.15120  72.88
## [4,] 72.88000  72.88 72.8800 73.6088 72.8800 71.42969 71.42969 72.87271  72.88
## [5,] 73.60880  72.88 73.6088 73.6088 72.1512 71.42969 72.14398 72.14398  72.88
## [6,] 74.34489  72.88 73.6088 73.6088 72.1512 71.42969 72.14398 71.42254  72.88
##       [,648]   [,649] [,650]  [,651]   [,652] [,653]   [,654]  [,655]  [,656]
## [1,] 72.8800 72.88000  72.88 72.8800 72.88000  72.88 72.88000 72.8800 72.8800
## [2,] 72.8800 72.88000  72.88 72.8800 72.88000  72.88 72.15120 72.8800 72.8800
## [3,] 73.6088 72.15120  72.88 72.8800 72.88000  72.88 72.87271 72.8800 73.6088
## [4,] 73.6088 72.15120  72.88 72.8800 72.15120  72.88 73.60144 72.1512 73.6088
## [5,] 73.6088 71.42969  72.88 72.8800 71.42969  72.88 73.60144 72.1512 73.6088
## [6,] 73.6088 72.14398  72.88 73.6088 72.14398  72.88 73.60144 72.1512 73.6088
##        [,657]   [,658]   [,659]  [,660]   [,661]  [,662]  [,663] [,664]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800 72.8800  72.88
## [2,] 72.88000 72.88000 72.15120 73.6088 72.88000 72.8800 72.8800  72.88
## [3,] 72.15120 72.88000 71.42969 73.6088 73.60880 72.8800 72.8800  72.88
## [4,] 71.42969 72.88000 72.14398 73.6088 73.60880 72.8800 72.1512  72.88
## [5,] 72.14398 72.15120 72.14398 73.6088 73.60880 72.8800 72.1512  72.88
## [6,] 72.14398 72.87271 72.14398 73.6088 74.34489 73.6088 72.1512  72.88
##        [,665]   [,666]   [,667]   [,668]   [,669]  [,670] [,671]  [,672]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800  72.88 72.8800
## [2,] 72.15120 72.15120 72.88000 73.60880 72.88000 72.8800  72.88 72.8800
## [3,] 71.42969 72.15120 73.60880 73.60880 73.60880 72.8800  72.88 73.6088
## [4,] 71.42969 72.15120 74.34489 73.60880 74.34489 72.8800  72.88 73.6088
## [5,] 71.42969 72.87271 74.34489 73.60880 74.34489 72.8800  72.88 73.6088
## [6,] 71.42969 72.87271 73.60144 74.34489 75.08833 73.6088  72.88 73.6088
##       [,673]  [,674]  [,675]   [,676]   [,677]   [,678]   [,679]   [,680]
## [1,] 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.1512 72.8800 72.8800 72.88000 72.88000 73.60880 72.88000 72.88000
## [3,] 72.1512 72.8800 73.6088 72.88000 72.88000 74.34489 72.15120 72.88000
## [4,] 72.1512 72.8800 73.6088 72.88000 72.15120 73.60144 72.15120 72.88000
## [5,] 72.1512 73.6088 73.6088 72.15120 72.87271 74.33745 71.42969 73.60880
## [6,] 72.1512 73.6088 73.6088 72.87271 73.60144 73.59408 71.42969 74.34489
##      [,681]   [,682]   [,683]  [,684]  [,685]   [,686]   [,687]  [,688]
## [1,]  72.88 72.88000 72.88000 72.8800 72.8800 72.88000 72.88000 72.8800
## [2,]  72.88 72.15120 72.15120 72.1512 72.8800 72.15120 72.88000 72.1512
## [3,]  72.88 72.15120 72.15120 72.1512 72.8800 71.42969 73.60880 72.1512
## [4,]  72.88 72.15120 72.15120 72.1512 72.8800 72.14398 74.34489 72.1512
## [5,]  72.88 72.15120 72.15120 72.1512 73.6088 72.86542 75.08833 72.1512
## [6,]  72.88 72.87271 72.87271 72.1512 73.6088 72.13677 75.08833 72.1512
##        [,689]  [,690]   [,691]   [,692]   [,693]   [,694]  [,695] [,696]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000 72.8800  72.88
## [2,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.15120 72.8800  72.88
## [3,] 72.15120 72.1512 73.60880 72.88000 73.60880 72.87271 73.6088  72.88
## [4,] 71.42969 72.1512 73.60880 72.88000 73.60880 73.60144 73.6088  72.88
## [5,] 72.14398 72.1512 74.34489 72.15120 74.34489 72.86542 73.6088  72.88
## [6,] 72.86542 72.1512 75.08833 71.42969 74.34489 72.13677 73.6088  72.88
##        [,697]   [,698] [,699]   [,700]   [,701]   [,702]   [,703]   [,704]
## [1,] 72.88000 72.88000  72.88 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.88000 72.15120  72.88 72.88000 73.60880 72.88000 73.60880 72.88000
## [3,] 72.88000 72.87271  72.88 72.15120 73.60880 72.88000 74.34489 72.88000
```

```
## [4,] 72.88000 72.14398  72.88 71.42969 73.60880 73.60880 74.34489 72.88000
## [5,] 73.60880 71.42254  72.88 71.42969 72.87271 73.60880 74.34489 72.15120
## [6,] 72.87271 71.42254  72.88 72.14398 72.87271 74.34489 75.08833 72.87271
##        [,705]  [,706]   [,707]  [,708]  [,709]   [,710]   [,711]   [,712]
## [1,] 72.88000 72.8800 72.88000 72.8800 72.8800 72.88000 72.88000 72.88000
## [2,] 73.60880 72.8800 72.88000 72.8800 73.6088 72.88000 72.88000 72.88000
## [3,] 73.60880 72.8800 72.88000 72.8800 73.6088 72.88000 73.60880 72.15120
## [4,] 73.60880 72.8800 73.60880 72.8800 73.6088 72.15120 74.34489 71.42969
## [5,] 74.34489 72.1512 74.34489 72.1512 73.6088 72.87271 74.34489 72.14398
## [6,] 75.08833 72.1512 74.34489 72.1512 73.6088 72.87271 75.08833 72.14398
##       [,713]   [,714]  [,715]   [,716]  [,717]  [,718]   [,719]   [,720]
## [1,] 72.8800 72.88000 72.8800 72.88000 72.8800 72.8800 72.88000 72.88000
## [2,] 73.6088 73.60880 72.8800 72.15120 72.8800 72.8800 73.60880 72.15120
## [3,] 73.6088 74.34489 72.8800 71.42969 72.8800 73.6088 74.34489 72.87271
## [4,] 73.6088 73.60144 72.8800 72.14398 72.8800 73.6088 73.60144 72.87271
## [5,] 73.6088 74.33745 72.8800 72.14398 72.8800 73.6088 73.60144 72.87271
## [6,] 73.6088 75.08083 72.1512 72.14398 72.1512 73.6088 73.60144 72.87271
##        [,721]  [,722]  [,723]   [,724]  [,725] [,726]  [,727]   [,728] [,729]
## [1,] 72.88000 72.8800 72.8800 72.88000 72.8800  72.88 72.8800 72.88000  72.88
## [2,] 72.15120 72.1512 72.8800 72.88000 72.8800  72.88 72.8800 73.60880  72.88
## [3,] 71.42969 72.1512 72.8800 72.15120 72.1512  72.88 72.8800 73.60880  72.88
## [4,] 71.42969 72.1512 72.8800 72.15120 72.1512  72.88 72.8800 74.34489  72.88
## [5,] 71.42969 72.1512 73.6088 71.42969 72.1512  72.88 73.6088 74.34489  72.88
## [6,] 70.71539 72.1512 73.6088 70.71539 72.1512  72.88 73.6088 74.34489  72.88
##        [,730]   [,731]   [,732]   [,733]  [,734]   [,735]  [,736]  [,737]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800 72.8800
## [2,] 72.88000 72.88000 73.60880 72.88000 72.8800 72.88000 72.8800 72.8800
## [3,] 73.60880 72.88000 73.60880 72.88000 72.8800 73.60880 72.8800 72.8800
## [4,] 73.60880 73.60880 72.87271 72.15120 72.8800 73.60880 72.8800 72.8800
## [5,] 73.60880 74.34489 73.60144 72.87271 72.1512 72.87271 73.6088 73.6088
## [6,] 74.34489 73.60144 72.86542 73.60144 72.1512 72.87271 73.6088 73.6088
##        [,738]   [,739]   [,740]   [,741]   [,742]  [,743]   [,744]   [,745]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,] 73.60880 72.88000 72.88000 72.88000 73.60880 72.8800 73.60880 73.60880
## [3,] 74.34489 72.15120 72.88000 73.60880 73.60880 73.6088 72.87271 72.87271
## [4,] 74.34489 72.87271 72.88000 72.87271 73.60880 73.6088 72.14398 72.14398
## [5,] 74.34489 72.87271 72.15120 72.14398 73.60880 73.6088 72.86542 71.42254
## [6,] 74.34489 72.87271 72.87271 72.86542 74.34489 73.6088 72.86542 72.13677
##        [,746]   [,747]  [,748]   [,749]   [,750]  [,751]   [,752]  [,753]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.88000 72.8800
## [2,] 73.60880 72.88000 72.8800 72.88000 72.15120 72.1512 72.15120 72.8800
## [3,] 74.34489 72.88000 72.8800 72.88000 71.42969 72.1512 72.87271 73.6088
## [4,] 73.60144 73.60880 72.8800 72.88000 71.42969 72.1512 73.60144 73.6088
## [5,] 73.60144 73.60880 72.8800 72.15120 70.71539 72.1512 73.60144 73.6088
## [6,] 74.33745 74.34489 72.1512 71.42969 70.71539 72.1512 73.60144 73.6088
##       [,754]  [,755] [,756]   [,757]   [,758]  [,759]   [,760]   [,761] [,762]
## [1,] 72.8800 72.8800  72.88 72.88000 72.88000 72.8800 72.88000 72.88000  72.88
## [2,] 72.8800 72.8800  72.88 72.88000 72.15120 72.8800 73.60880 72.88000  72.88
## [3,] 72.8800 72.8800  72.88 72.88000 72.15120 73.6088 73.60880 73.60880  72.88
## [4,] 72.8800 73.6088  72.88 72.15120 72.15120 73.6088 74.34489 72.87271  72.88
## [5,] 72.8800 73.6088  72.88 72.87271 72.87271 73.6088 75.08833 72.87271  72.88
## [6,] 73.6088 73.6088  72.88 73.60144 72.87271 73.6088 75.83922 72.14398  72.88
##        [,763]  [,764]   [,765]  [,766]  [,767]   [,768] [,769]   [,770]
## [1,] 72.88000 72.8800 72.88000 72.8800 72.8800 72.88000  72.88 72.88000
```

```
## [2,] 72.88000 72.8800 73.60880 73.6088 72.8800 72.88000  72.88 72.88000
## [3,] 72.88000 73.6088 74.34489 73.6088 72.1512 72.15120  72.88 72.15120
## [4,] 72.15120 73.6088 73.60144 73.6088 72.1512 72.15120  72.88 72.87271
## [5,] 72.87271 73.6088 74.33745 73.6088 72.1512 72.15120  72.88 72.87271
## [6,] 73.60144 73.6088 73.59408 73.6088 72.1512 71.42969  72.88 72.14398
##        [,771]   [,772]   [,773]  [,774]   [,775]   [,776]  [,777]  [,778]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.8800
## [2,] 72.88000 73.60880 73.60880 72.8800 73.60880 73.60880 72.1512 72.8800
## [3,] 72.15120 74.34489 74.34489 72.8800 73.60880 74.34489 72.1512 72.8800
## [4,] 72.15120 74.34489 74.34489 72.8800 74.34489 75.08833 72.1512 72.8800
## [5,] 71.42969 74.34489 73.60144 72.1512 74.34489 75.08833 72.1512 72.1512
## [6,] 70.71539 74.34489 73.60144 72.1512 74.34489 75.08833 72.1512 72.1512
##        [,779]  [,780]   [,781]  [,782]   [,783]   [,784]   [,785]   [,786]
## [1,] 72.88000 72.8800 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000
## [2,] 72.88000 72.8800 72.88000 72.8800 72.88000 73.60880 73.60880 72.88000
## [3,] 73.60880 73.6088 72.88000 72.8800 72.88000 73.60880 72.87271 72.88000
## [4,] 74.34489 73.6088 73.60880 72.8800 73.60880 74.34489 72.87271 73.60880
## [5,] 74.34489 73.6088 74.34489 72.8800 74.34489 75.08833 72.87271 74.34489
## [6,] 74.34489 73.6088 74.34489 73.6088 74.34489 75.83922 73.60144 73.60144
##        [,787]  [,788]   [,789]  [,790]  [,791]   [,792]   [,793]  [,794]
## [1,] 72.88000 72.8800 72.88000 72.8800 72.8800 72.88000 72.88000 72.8800
## [2,] 72.88000 72.8800 72.15120 72.8800 72.8800 72.15120 72.88000 72.8800
## [3,] 73.60880 73.6088 72.87271 72.8800 72.8800 71.42969 73.60880 72.1512
## [4,] 72.87271 73.6088 72.87271 72.8800 72.8800 72.14398 73.60880 72.1512
## [5,] 72.87271 73.6088 72.14398 73.6088 72.8800 72.14398 73.60880 72.1512
## [6,] 72.14398 73.6088 72.14398 73.6088 73.6088 72.14398 74.34489 72.1512
##       [,795]   [,796]   [,797]   [,798]  [,799]   [,800]  [,801]   [,802]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800 72.88000
## [2,] 73.6088 72.15120 72.88000 72.15120 72.8800 73.60880 72.8800 72.88000
## [3,] 73.6088 72.87271 72.15120 72.87271 72.1512 73.60880 73.6088 72.15120
## [4,] 73.6088 72.87271 71.42969 73.60144 72.1512 72.87271 73.6088 72.87271
## [5,] 73.6088 72.87271 70.71539 73.60144 72.1512 72.87271 73.6088 72.87271
## [6,] 73.6088 72.87271 70.71539 73.60144 72.1512 73.60144 73.6088 73.60144
##       [,803]  [,804]  [,805]   [,806]   [,807]   [,808]   [,809]   [,810]
## [1,] 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800 72.8800 72.8800 73.60880 73.60880 72.88000 72.88000 73.60880
## [3,] 72.8800 72.8800 72.8800 72.87271 74.34489 72.88000 73.60880 74.34489
## [4,] 72.8800 72.8800 72.8800 72.87271 73.60144 73.60880 72.87271 74.34489
## [5,] 72.8800 73.6088 72.1512 72.14398 72.86542 72.87271 73.60144 74.34489
## [6,] 73.6088 73.6088 72.1512 71.42254 72.86542 72.87271 73.60144 74.34489
##        [,811]   [,812]  [,813]   [,814]   [,815]  [,816]   [,817]   [,818]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,] 72.88000 72.15120 72.8800 72.15120 72.88000 72.8800 73.60880 72.88000
## [3,] 72.15120 72.15120 72.8800 72.87271 73.60880 72.8800 73.60880 72.15120
## [4,] 72.87271 71.42969 72.8800 72.14398 72.87271 72.8800 73.60880 71.42969
## [5,] 73.60144 71.42969 72.8800 72.14398 72.87271 72.8800 73.60880 72.14398
## [6,] 73.60144 71.42969 72.1512 72.14398 72.87271 73.6088 74.34489 72.14398
##       [,819] [,820]   [,821]  [,822] [,823]   [,824]   [,825]   [,826]   [,827]
## [1,] 72.8800  72.88 72.88000 72.8800  72.88 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800  72.88 72.15120 72.8800  72.88 72.88000 72.88000 72.15120 72.88000
## [3,] 72.8800  72.88 72.87271 72.8800  72.88 72.15120 72.88000 72.87271 73.60880
## [4,] 72.8800  72.88 73.60144 72.8800  72.88 72.87271 72.88000 72.87271 72.87271
## [5,] 73.6088  72.88 72.86542 72.8800  72.88 73.60144 72.15120 73.60144 72.87271
## [6,] 73.6088  72.88 72.86542 73.6088  72.88 74.33745 72.87271 73.60144 72.87271
```

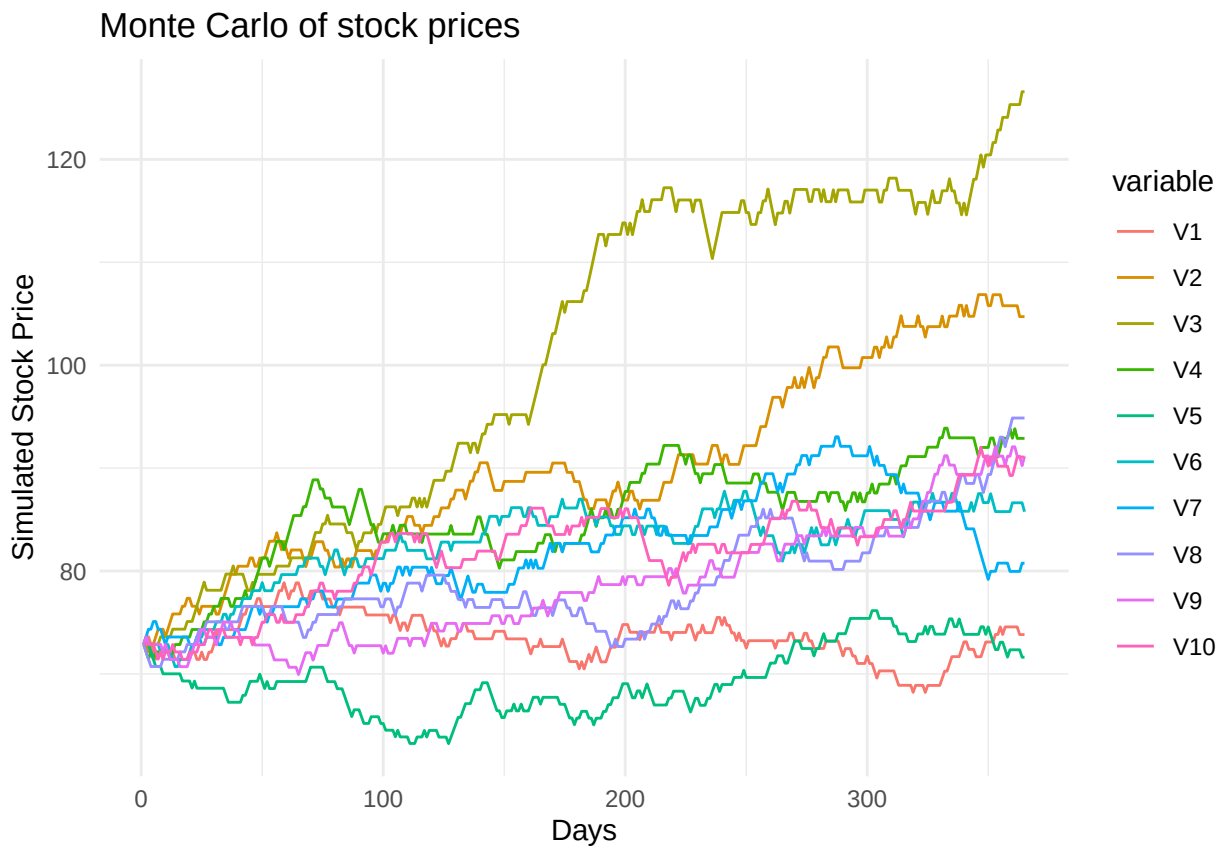
```
##       [,828]   [,829]   [,830]   [,831]   [,832]   [,833]   [,834]   [,835]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.15120 72.88000 72.88000
## [3,] 73.6088 72.88000 72.88000 73.60880 72.15120 72.15120 72.88000 72.15120
## [4,] 73.6088 73.60880 72.88000 74.34489 72.15120 71.42969 72.15120 72.15120
## [5,] 73.6088 72.87271 73.60880 74.34489 72.15120 71.42969 72.87271 71.42969
## [6,] 73.6088 72.87271 72.87271 74.34489 72.87271 70.71539 73.60144 70.71539
##       [,836]   [,837]  [,838]   [,839]   [,840]   [,841]  [,842]   [,843]
## [1,] 72.8800 72.88000 72.8800 72.88000 72.88000 72.88000 72.8800 72.88000
## [2,] 72.8800 72.15120 72.8800 72.88000 72.88000 72.88000 72.8800 73.60880
## [3,] 72.8800 72.87271 72.8800 72.15120 72.88000 72.15120 72.8800 74.34489
## [4,] 73.6088 73.60144 72.8800 71.42969 72.88000 71.42969 72.8800 74.34489
## [5,] 73.6088 73.60144 72.8800 71.42969 73.60880 71.42969 72.8800 74.34489
## [6,] 73.6088 74.33745 73.6088 71.42969 72.87271 71.42969 73.6088 74.34489
##      [,844]   [,845]  [,846]  [,847]  [,848]   [,849]   [,850]   [,851]
## [1,]  72.88 72.88000 72.8800 72.8800 72.8800 72.88000 72.88000 72.88000
## [2,]  72.88 73.60880 72.8800 72.8800 72.8800 72.15120 73.60880 73.60880
## [3,]  72.88 73.60880 72.8800 73.6088 72.8800 72.15120 74.34489 73.60880
## [4,]  72.88 73.60880 72.8800 73.6088 72.8800 71.42969 74.34489 74.34489
## [5,]  72.88 73.60880 72.8800 73.6088 73.6088 71.42969 73.60144 74.34489
## [6,]  72.88 74.34489 73.6088 73.6088 73.6088 71.42969 74.33745 74.34489
##        [,852]  [,853]   [,854]   [,855]  [,856]   [,857]   [,858]  [,859]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.8800 72.88000 72.88000 72.8800
## [2,] 72.15120 72.8800 72.15120 72.88000 72.8800 72.15120 72.88000 72.8800
## [3,] 72.15120 72.8800 72.15120 72.88000 72.8800 71.42969 72.15120 72.8800
## [4,] 72.15120 72.1512 72.15120 73.60880 73.6088 72.14398 72.87271 72.8800
## [5,] 72.15120 72.1512 72.87271 73.60880 73.6088 72.14398 72.14398 72.8800
## [6,] 72.87271 72.1512 73.60144 74.34489 73.6088 72.86542 72.14398 72.1512
##       [,860]   [,861] [,862] [,863]   [,864]  [,865]   [,866]   [,867]  [,868]
## [1,] 72.8800 72.88000  72.88  72.88 72.88000 72.8800 72.88000 72.88000 72.8800
## [2,] 73.6088 73.60880  72.88  72.88 72.88000 72.8800 73.60880 72.88000 72.8800
## [3,] 73.6088 73.60880  72.88  72.88 72.88000 72.8800 73.60880 72.15120 72.8800
## [4,] 73.6088 73.60880  72.88  72.88 72.15120 72.8800 73.60880 72.87271 73.6088
## [5,] 73.6088 73.60880  72.88  72.88 72.15120 72.8800 73.60880 73.60144 73.6088
## [6,] 73.6088 72.87271  72.88  72.88 71.42969 72.1512 74.34489 73.60144 73.6088
##       [,869]   [,870]  [,871]   [,872]  [,873]   [,874]   [,875]   [,876]
## [1,] 72.8800 72.88000 72.8800 72.88000 72.8800 72.88000 72.88000 72.88000
## [2,] 72.8800 72.88000 72.8800 73.60880 72.8800 72.15120 72.88000 72.88000
## [3,] 73.6088 72.88000 72.8800 74.34489 72.8800 71.42969 72.15120 73.60880
## [4,] 73.6088 72.88000 72.1512 75.08833 72.8800 72.14398 71.42969 73.60880
## [5,] 73.6088 73.60880 72.1512 75.83922 72.1512 72.14398 70.71539 74.34489
## [6,] 73.6088 74.34489 72.1512 75.08083 72.1512 72.86542 71.42254 74.34489
##        [,877]   [,878]   [,879]   [,880]   [,881]  [,882]   [,883]  [,884]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800
## [2,] 72.15120 72.88000 72.88000 73.60880 72.88000 73.6088 72.15120 72.1512
## [3,] 72.15120 72.88000 72.15120 73.60880 72.88000 73.6088 72.15120 72.1512
## [4,] 72.15120 72.15120 71.42969 73.60880 73.60880 73.6088 72.15120 72.1512
## [5,] 72.87271 72.87271 71.42969 73.60880 72.87271 73.6088 72.15120 72.1512
## [6,] 72.87271 72.87271 71.42969 72.87271 72.87271 73.6088 71.42969 72.1512
##      [,885]  [,886]   [,887]  [,888] [,889]  [,890] [,891]  [,892]   [,893]
## [1,]  72.88 72.8800 72.88000 72.8800  72.88 72.8800  72.88 72.8800 72.88000
## [2,]  72.88 72.8800 73.60880 72.8800  72.88 73.6088  72.88 72.8800 73.60880
## [3,]  72.88 72.8800 74.34489 72.1512  72.88 73.6088  72.88 72.8800 72.87271
## [4,]  72.88 72.8800 74.34489 72.1512  72.88 73.6088  72.88 72.8800 72.14398
```

```
## [5,]  72.88 72.8800 74.34489 72.1512  72.88 73.6088  72.88 72.8800 72.14398
## [6,]  72.88 73.6088 74.34489 72.1512  72.88 73.6088  72.88 73.6088 72.14398
##        [,894]  [,895]   [,896]   [,897]   [,898]  [,899]   [,900]   [,901]
## [1,] 72.88000 72.8800 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,] 72.88000 72.8800 72.88000 72.88000 73.60880 72.8800 72.15120 72.88000
## [3,] 73.60880 72.1512 73.60880 72.88000 72.87271 72.8800 72.87271 72.88000
## [4,] 73.60880 72.1512 74.34489 72.88000 72.87271 73.6088 73.60144 72.15120
## [5,] 74.34489 72.1512 74.34489 72.15120 72.14398 73.6088 73.60144 72.87271
## [6,] 73.60144 72.1512 74.34489 71.42969 72.86542 73.6088 74.33745 73.60144
##       [,902]   [,903] [,904]   [,905]  [,906]  [,907]   [,908]  [,909]   [,910]
## [1,] 72.8800 72.88000  72.88 72.88000 72.8800 72.8800 72.88000 72.8800 72.88000
## [2,] 72.8800 73.60880  72.88 72.15120 72.8800 72.8800 73.60880 72.8800 72.88000
## [3,] 72.8800 73.60880  72.88 71.42969 73.6088 72.8800 73.60880 72.8800 73.60880
## [4,] 72.8800 73.60880  72.88 71.42969 73.6088 72.8800 73.60880 72.8800 73.60880
## [5,] 73.6088 73.60880  72.88 71.42969 73.6088 72.1512 74.34489 72.1512 73.60880
## [6,] 73.6088 74.34489  72.88 71.42969 73.6088 72.1512 73.60144 72.1512 74.34489
##      [,911]   [,912]   [,913]   [,914]   [,915]  [,916]   [,917]   [,918]
## [1,]  72.88 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000
## [2,]  72.88 72.15120 72.15120 72.88000 73.60880 72.1512 72.88000 73.60880
## [3,]  72.88 72.87271 71.42969 72.88000 74.34489 72.1512 73.60880 74.34489
## [4,]  72.88 72.87271 71.42969 72.15120 73.60144 72.1512 74.34489 74.34489
## [5,]  72.88 72.14398 71.42969 72.87271 74.33745 72.1512 74.34489 74.34489
## [6,]  72.88 72.86542 70.71539 72.87271 74.33745 72.1512 74.34489 75.08833
##       [,919]   [,920]   [,921]   [,922]   [,923]   [,924]   [,925]   [,926]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000
## [2,] 72.8800 73.60880 72.88000 72.15120 73.60880 72.88000 72.88000 72.15120
## [3,] 72.8800 72.87271 73.60880 71.42969 73.60880 72.88000 72.88000 72.15120
## [4,] 72.8800 72.87271 72.87271 70.71539 72.87271 72.88000 72.15120 72.87271
## [5,] 72.8800 72.87271 72.87271 70.00823 73.60144 72.15120 72.87271 72.87271
## [6,] 73.6088 72.87271 72.87271 70.00823 74.33745 71.42969 73.60144 72.87271
##        [,927]   [,928]   [,929]   [,930]   [,931]   [,932]  [,933]   [,934]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000
## [2,] 72.88000 72.88000 73.60880 73.60880 72.88000 72.88000 72.8800 72.88000
## [3,] 72.88000 72.88000 74.34489 73.60880 72.88000 72.88000 72.8800 72.15120
## [4,] 72.15120 72.15120 73.60144 74.34489 72.15120 72.15120 73.6088 72.87271
## [5,] 72.87271 71.42969 72.86542 74.34489 72.15120 72.87271 73.6088 72.87271
## [6,] 72.87271 71.42969 72.86542 74.34489 72.87271 73.60144 73.6088 72.87271
##        [,935]   [,936]   [,937]   [,938]  [,939]  [,940]  [,941]   [,942]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800 72.8800 72.88000
## [2,] 72.88000 72.88000 73.60880 72.88000 72.8800 72.8800 72.8800 72.15120
## [3,] 73.60880 72.88000 74.34489 72.15120 72.8800 73.6088 72.8800 71.42969
## [4,] 72.87271 72.15120 74.34489 71.42969 73.6088 73.6088 72.8800 71.42969
## [5,] 72.87271 72.15120 73.60144 70.71539 73.6088 73.6088 72.1512 72.14398
## [6,] 72.14398 72.87271 72.86542 70.00823 73.6088 73.6088 72.1512 72.14398
##        [,943]   [,944]   [,945]   [,946]   [,947]  [,948]   [,949]  [,950]
## [1,] 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.88000 72.8800
## [2,] 72.15120 72.88000 73.60880 73.60880 72.88000 72.8800 73.60880 73.6088
## [3,] 71.42969 73.60880 72.87271 74.34489 72.88000 72.8800 73.60880 73.6088
## [4,] 70.71539 73.60880 72.87271 75.08833 72.15120 72.8800 73.60880 73.6088
## [5,] 70.71539 74.34489 73.60144 75.83922 72.87271 72.8800 72.87271 73.6088
## [6,] 70.71539 74.34489 73.60144 75.83922 72.87271 73.6088 73.60144 73.6088
##        [,951]   [,952] [,953]   [,954]   [,955]   [,956] [,957]  [,958]
## [1,] 72.88000 72.88000  72.88 72.88000 72.88000 72.88000  72.88 72.8800
## [2,] 73.60880 72.88000  72.88 72.88000 72.88000 72.15120  72.88 72.8800
```

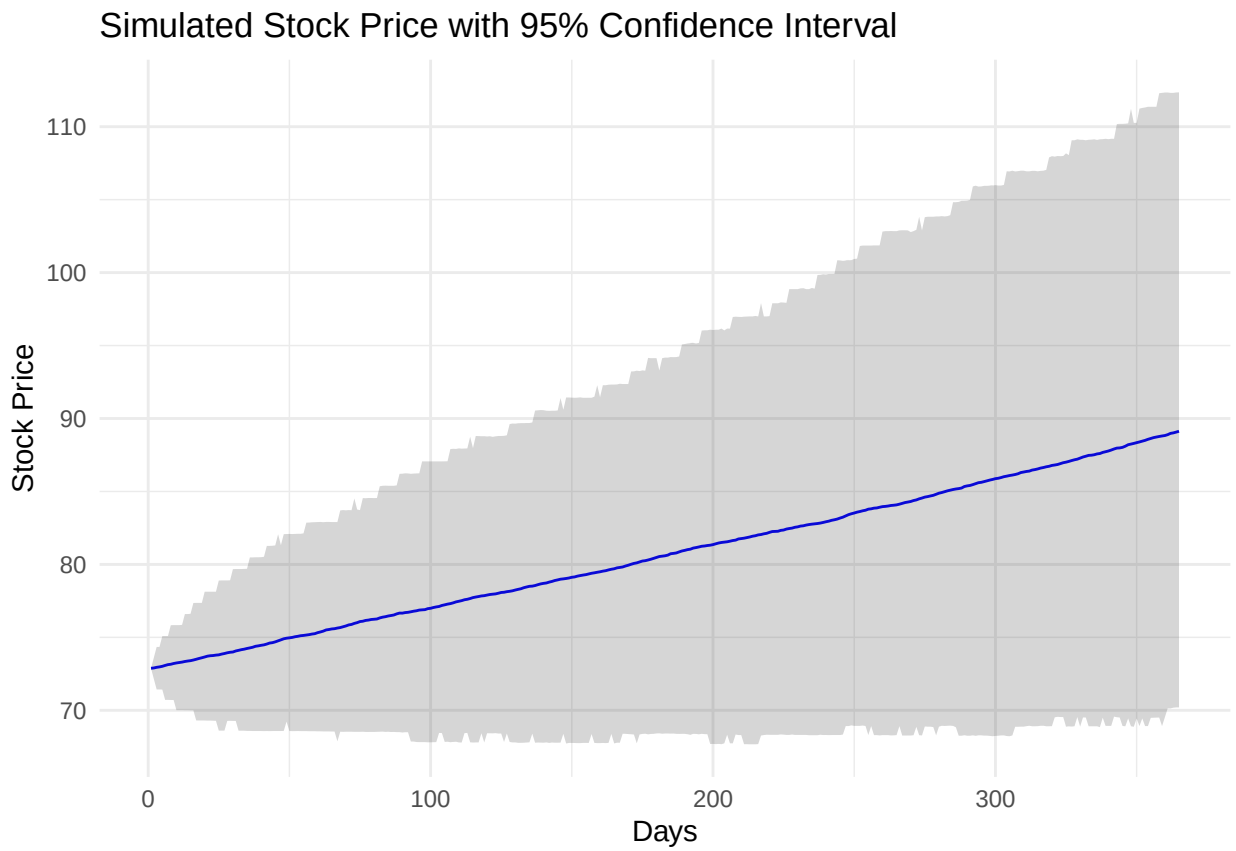
```
## [3,] 74.34489 72.88000  72.88 72.88000 72.15120 72.15120  72.88 72.8800
## [4,] 74.34489 72.15120  72.88 72.15120 72.87271 72.15120  72.88 72.8800
## [5,] 74.34489 71.42969  72.88 71.42969 72.87271 71.42969  72.88 72.1512
## [6,] 73.60144 71.42969  72.88 72.14398 72.87271 71.42969  72.88 72.1512
##        [,959] [,960]   [,961]   [,962] [,963]   [,964]   [,965]   [,966]
## [1,] 72.88000  72.88 72.88000 72.88000  72.88 72.88000 72.88000 72.88000
## [2,] 73.60880  72.88 72.88000 72.88000  72.88 72.15120 73.60880 72.88000
## [3,] 74.34489  72.88 73.60880 72.88000  72.88 72.87271 73.60880 73.60880
## [4,] 74.34489  72.88 74.34489 72.88000  72.88 72.87271 73.60880 72.87271
## [5,] 74.34489  72.88 74.34489 73.60880  72.88 72.87271 73.60880 72.87271
## [6,] 73.60144  72.88 74.34489 74.34489  72.88 72.87271 74.34489 72.87271
##       [,967]   [,968]   [,969]   [,970]   [,971]   [,972]  [,973]  [,974]
## [1,] 72.8800 72.88000 72.88000 72.88000 72.88000 72.88000 72.8800 72.8800
## [2,] 72.8800 72.88000 72.88000 73.60880 72.88000 72.88000 72.8800 72.8800
## [3,] 72.8800 72.88000 72.88000 73.60880 72.88000 72.88000 72.8800 72.8800
## [4,] 72.8800 72.15120 72.88000 73.60880 73.60880 72.88000 72.8800 73.6088
## [5,] 72.8800 72.87271 73.60880 74.34489 74.34489 72.15120 73.6088 73.6088
## [6,] 73.6088 72.87271 74.34489 74.34489 73.60144 71.42969 73.6088 73.6088
##        [,975]   [,976]  [,977]   [,978]   [,979]   [,980]   [,981] [,982]
## [1,] 72.88000 72.88000 72.8800 72.88000 72.88000 72.88000 72.88000  72.88
## [2,] 73.60880 72.88000 72.8800 72.88000 72.88000 72.88000 73.60880  72.88
## [3,] 74.34489 73.60880 72.8800 73.60880 72.15120 73.60880 73.60880  72.88
## [4,] 74.34489 73.60880 73.6088 72.87271 71.42969 74.34489 74.34489  72.88
## [5,] 74.34489 72.87271 73.6088 72.87271 71.42969 74.34489 74.34489  72.88
## [6,] 74.34489 72.87271 73.6088 72.87271 71.42969 74.34489 74.34489  72.88
##      [,983]   [,984] [,985]  [,986]   [,987] [,988]  [,989]   [,990]   [,991]
## [1,]  72.88 72.88000  72.88 72.8800 72.88000  72.88 72.8800 72.88000 72.88000
## [2,]  72.88 72.15120  72.88 72.8800 72.15120  72.88 72.8800 72.88000 72.15120
## [3,]  72.88 72.15120  72.88 72.8800 72.87271  72.88 73.6088 72.15120 72.87271
## [4,]  72.88 72.15120  72.88 72.8800 72.87271  72.88 73.6088 72.87271 73.60144
## [5,]  72.88 72.15120  72.88 73.6088 72.87271  72.88 73.6088 73.60144 73.60144
## [6,]  72.88 71.42969  72.88 73.6088 72.14398  72.88 73.6088 72.86542 72.86542
##        [,992]   [,993]   [,994]  [,995]   [,996]   [,997] [,998]   [,999]
## [1,] 72.88000 72.88000 72.88000 72.8800 72.88000 72.88000  72.88 72.88000
## [2,] 73.60880 72.88000 72.88000 72.8800 72.88000 73.60880  72.88 72.15120
## [3,] 73.60880 73.60880 72.88000 73.6088 73.60880 72.87271  72.88 71.42969
## [4,] 73.60880 74.34489 72.15120 73.6088 74.34489 73.60144  72.88 71.42969
## [5,] 73.60880 75.08833 72.15120 73.6088 75.08833 72.86542  72.88 71.42969
## [6,] 72.87271 75.08833 72.87271 73.6088 75.83922 73.59408  72.88 71.42969
##       [,1000]
## [1,] 72.88000
## [2,] 72.88000
## [3,] 72.88000
## [4,] 73.60880
## [5,] 72.87271
## [6,] 72.14398
```

```r
simulated_df <- as.data.frame(all_sim)


#for plotting
simulated_df$Day <- 1:n_days
```

```r
simulated_long <- melt(simulated_df, id = "Day")

#plot first 10 paths

ggplot(simulated_long[simulated_long$variable %in% paste0("V", 1:10),], aes(x = Day, y = value, color=va
  labs(title = "Monte Carlo of stock prices", x = "Days", y = "Simulated Stock Price") +
  theme_minimal()
```




```r
#As we can see, as Monte Carlo is Stochastic statistics this data itself doesn't help us make conclusio


summary_stats<- data.frame(Day = 1:n_days,Mean=apply(all_sim,1,mean),
                           Median=apply(all_sim,1,median),
                           Lower95=apply(all_sim, 1,quantile, probs=0.025),
                           Upper95=apply(all_sim,1,quantile,probs=0.975))


#now plot mean and confint

ggplot(summary_stats, aes(x = Day)) +
  geom_line(aes(y=Mean), color = "blue")+
  geom_ribbon(aes(ymin=Lower95, ymax=Upper95), alpha=0.2) +
  labs(title = "Simulated Stock Price with 95% Confidence Interval", x = "Days", y = "Stock Price") +
  theme_minimal()
```




```
#now that we have metrics well annotated and ready,let's move on to Value at Risk calculation (Of cours

#final day prices from all sims

final_prices<- all_sim[n_days, ]


#VaR
VaR_95 <- quantile(final_prices, probs = 0.05)
VaR_95
```

```
##      5%
## 71.7045
```

```
#Now more in depth, looking at conditioanal VaR

CVaR_prices <- final_prices[final_prices < VaR_95]

#calculate CVaR as avg

CVaR_95<- mean(CVaR_prices)
CVaR_95
```

```
## [1] 68.33302
```

```r
#Not done yet, now calculate Maximum Drawdown which is essentially worst case decline from peak
max_drawdown_fun <- function(price_path){
  peak <- max(price_path)
  trough<- min(price_path[which(price_path == peak): length(price_path)])

  max_drawdown<- (peak- trough) / peak
  return(max_drawdown)

}


#apply to all

max_dds<- apply(all_sim,2, max_drawdown_fun)
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 12 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 11 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 10 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 10 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 18 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 21 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 10 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 16 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 14 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 16 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 11 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 8 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 6 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 12 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 10 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 10 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 7 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 11 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used
```

```
## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 5 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 3 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 4 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 2 elements: only the first used

## Warning in which(price_path == peak):length(price_path): numerical expression
## has 9 elements: only the first used
```

```
#calculate avg and worst case dd
avg_max_dd <- mean(max_dds)
worst_case <- max(max_dds)


avg_max_dd
```

```
## [1] 0.04833077
```

```
worst_case
```

```
## [1] 0.2775528
```

```
#plot the dist

ggplot(data.frame(Price= final_prices), aes(x = Price))+
  geom_histogram(bins = 30, fill= "blue", alpha = 0.7)+
  geom_vline(xintercept = VaR_95, color = "red", linetype = "dashed", size = 1.5)+
  geom_vline(xintercept = CVaR_95, color = "orange", linetype = "dashed", size = 1.5) +
  labs(title = "Distribution of Simulated Final Prices", x = "Final Stock Price",
       y = "Frequency") +
  theme_minimal() +
  annotate("text", x=VaR_95, y = max(hist(final_prices, plot = FALSE)$counts),
           label = paste("VaR (95%):", round(VaR_95, 2)), hjust = 1.5, color = "red") +
  annotate("text", x = CVaR_95, y = max(hist(final_prices, plot = FALSE)$counts),
           label = paste("CVaR (95%):", round(CVaR_95, 2)), hjust = -0.5, color = "orange")
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```


```r
#let's check for sensitivity analysis for different percentages
sens_analysis <- function(up_pct_vals, down_pct_vals, n_simulations, n_days){
  results <- data.frame()

  for(up in up_pct_vals){
    for(down in down_pct_vals){

      #run sim
      sim <- run_monte_carlo(initial_price, initial_state, transition_matrix, n_days, n_simulations, up


      #calculate VaR
      final <-sim[n_days, ]
      VaR_95 <- quantile(final, probs = 0.05)
      CVaR_95<- mean(final[final < VaR_95])

      #store
      results <- rbind(results, data.frame(up, down, VaR_95, CVaR_95))
    }
  }
```

```
  return(results)
}



#analysis
up_pct_vals <- c(0.01,0.02,0.05)
down_pct_vals<- c(0.01,0.02,0.05)
sens_res<- sens_analysis(up_pct_vals, down_pct_vals, n_simulations,n_days)
sens_res
```

```
##       up down      VaR_95    CVaR_95
## 5%  0.01 0.01   73.080326  69.310895
## 5%1 0.01 0.02   34.147766  31.814456
## 5%2 0.01 0.05    3.417604   2.884429
## 5%3 0.02 0.01  142.965245 131.869886
## 5%4 0.02 0.02   70.834669  64.433999
## 5%5 0.02 0.05    6.812242   5.754451
## 5%6 0.05 0.01 1057.141540 897.016453
## 5%7 0.05 0.02  521.911461 448.612612
## 5%8 0.05 0.05   61.005679  49.042255
```

```
#This provides great info!
```

```
#Now let's do stress testing to understand prices react to extreme conditions

#We modify tr matrix to increase probability of "Down"

stress_trans_matrix <- transition_matrix
stress_trans_matrix["Up", "Down"] <- 0.4
stress_trans_matrix["Down", "Down"] <- 0.7
stress_trans_matrix<- prop.table(stress_trans_matrix,1) #normalize

stress_sims <- run_monte_carlo(initial_price, initial_state, stress_trans_matrix, n_days, n_simulations



#calculate value at risk and conditioned var

final_prices_stress<- stress_sims[n_days, ]
var_95_stress<- quantile(final_prices_stress, probs=0.05)

cvar_95_stress <- mean(final_prices_stress[final_prices_stress < var_95_stress])

var_95_stress
```

```
##           5%
## 0.0005976425
```

```
cvar_95_stress
```

```
## [1] 0.0003536521
```

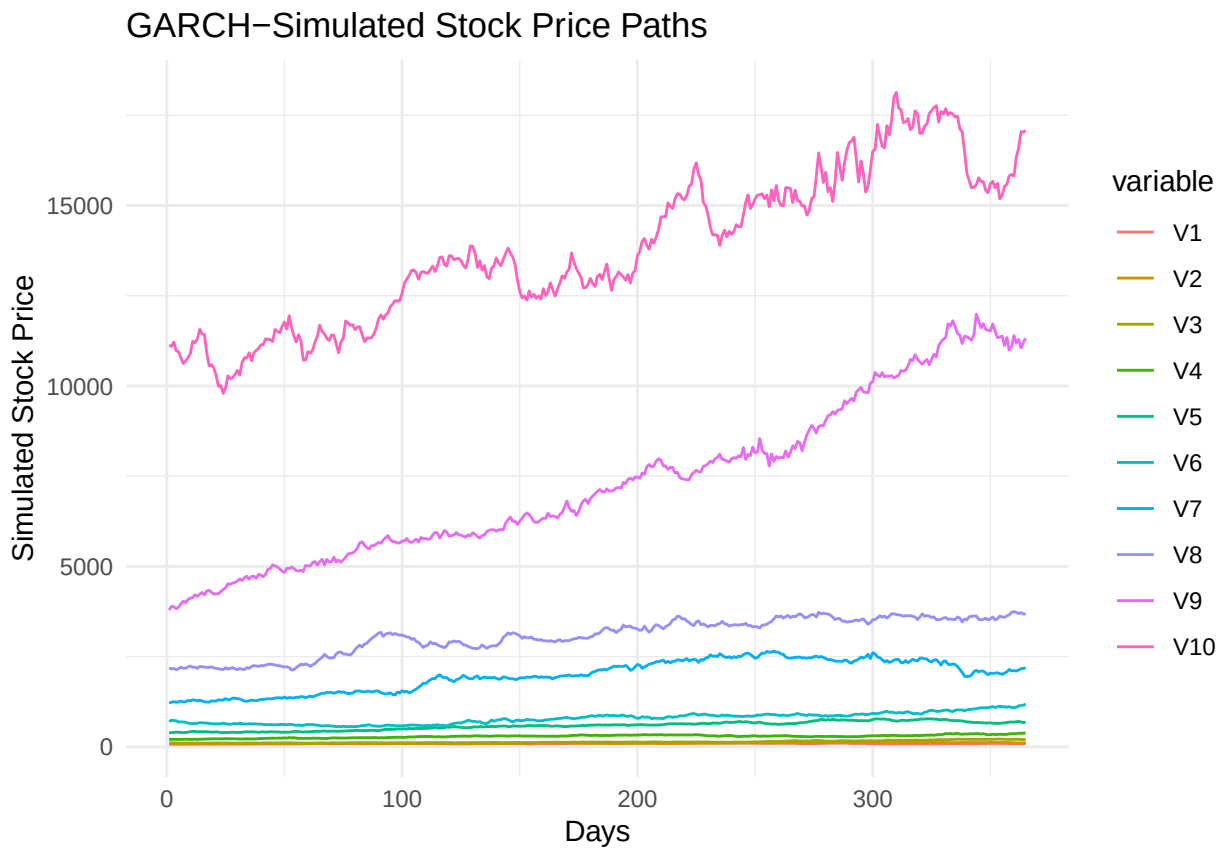
```r
#Finally let's try GARCH modeling to test for varying variances
library(rugarch)
```

```
## Loading required package: parallel
```

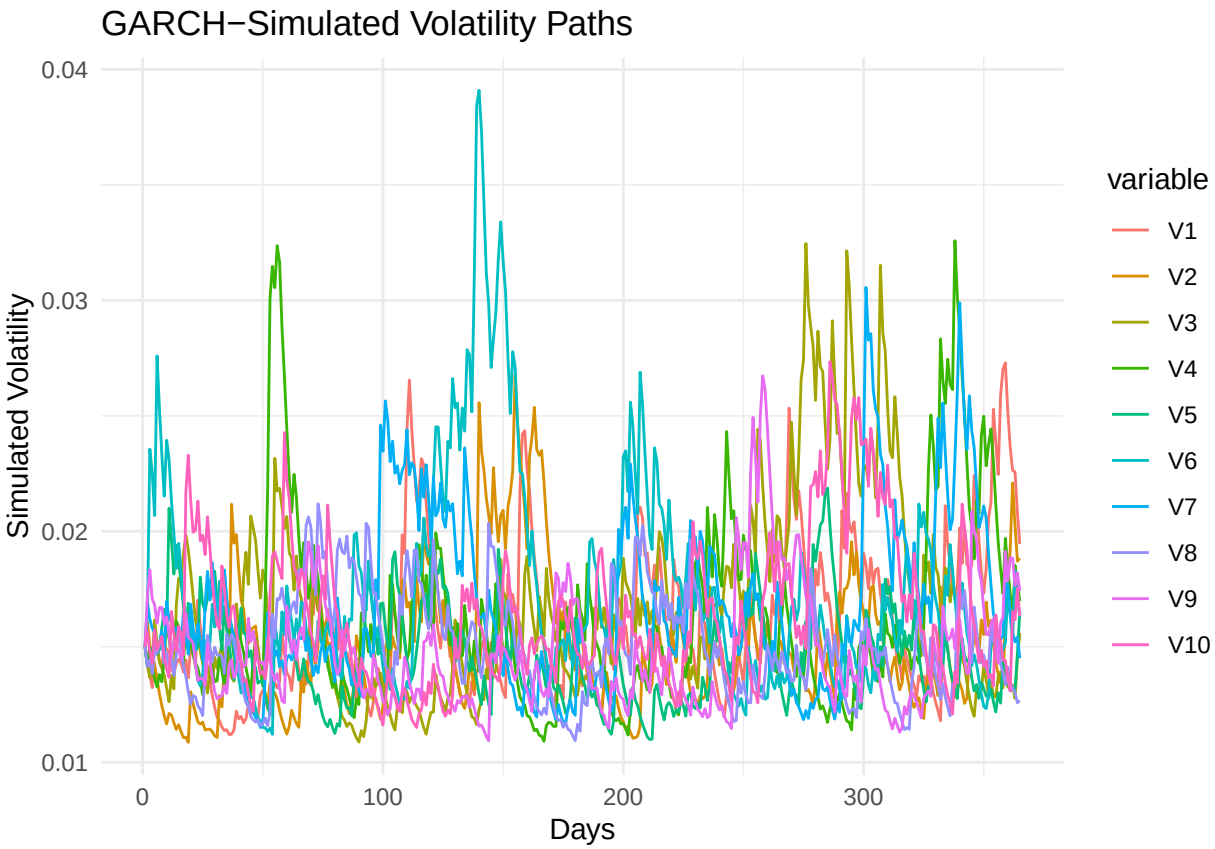
```
##
## Attaching package: 'rugarch'
```

```
## The following object is masked from 'package:stats':
##
##     sigma
```

```r
# Specify GARCH(1,1) model
garch_spec <- ugarchspec(variance.model = list(garchOrder = c(1, 1)),
                         mean.model = list(armaOrder = c(1, 0)))

# Fit GARCH model to stock returns
garch_fit <- ugarchfit(spec = garch_spec, data = returns)

# Simulate future returns with GARCH model
garch_sim <- ugarchsim(garch_fit, n.sim = n_days, m.sim = n_simulations)

# Extract simulated prices
simulated_returns <- fitted(garch_sim)
garch_price_paths <- initial_price * exp(cumsum(simulated_returns))

# Make sure garch_price_paths is a matrix with columns for each simulation
garch_price_paths <- matrix(garch_price_paths, nrow = n_days, ncol = n_simulations)

# Convert the price paths matrix into a data frame
garch_simulated_df <- as.data.frame(garch_price_paths)
garch_simulated_df$Day <- 1:n_days  # Add the Day column for plotting

# Reshape the data for ggplot, treating each column as a separate simulation (V1, V2, ..., Vn)
library(reshape2)
garch_long <- melt(garch_simulated_df, id = "Day")

# Check the reshaped data (this should now contain variables like V1, V2, etc.)

# Plot the first 10 GARCH-simulated paths
ggplot(garch_long[garch_long$variable %in% paste0("V", 1:10),], aes(x = Day, y = value, color = variable
  geom_line() +
  labs(title = "GARCH-Simulated Stock Price Paths", x = "Days", y = "Simulated Stock Price") +
  theme_minimal()
```




```r
#volatility check from GARCH(common)


sim_volatility<- sigma(garch_sim)

volatility_df <- as.data.frame(sim_volatility)

volatility_df$Day <- 1:n_days


#reshape

volatility_long <- melt(volatility_df, id = "Day")


#plot
ggplot(volatility_long[volatility_long$variable %in% paste0("V", 1:10),], aes(x = Day, y = value, color
  geom_line() +
  labs(title = "GARCH-Simulated Volatility Paths", x = "Days", y = "Simulated Volatility") +
  theme_minimal()
```


GARCH−Simulated Volatility Paths
Simulated Volatility
0.04
0.03
0.02
0.01
0
100
200
300
Days
variable
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10