Untitled

Homework 2

April 6, 2017

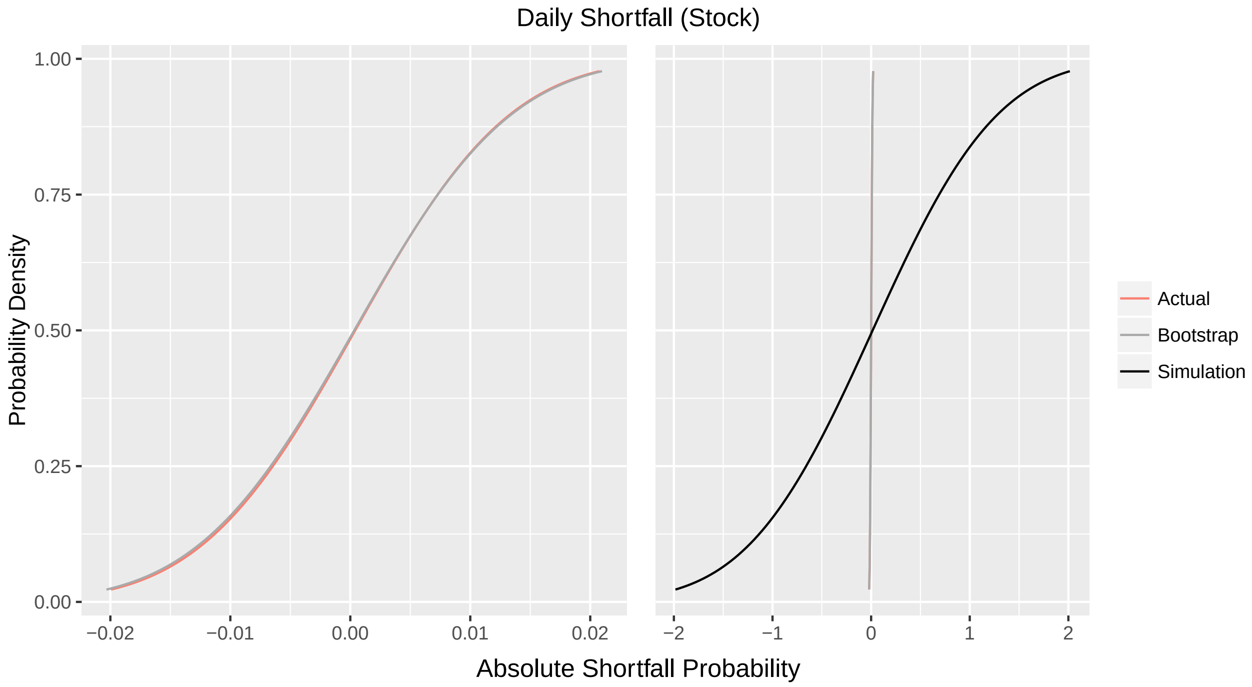
```
##############
# Homework 2 #
##############

########
# Load Config Files
########

options("width" = 250)
options(scipen  = 999)
options(digits  = 003)

library(xts); library(zoo); library(e1071);
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

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

```
library(ggplot2); library(knitr); library(gridExtra)
library(reshape2)

set.seed(666) # the devils seed

username    <- Sys.info()[["user"]]
dir         <- paste("/home/", username, "/Documents/Education/Chicago_Booth/Classes/35120_Portfolio_Ma
setwd(dir)

getReturns <- function(name){
                returns <- read.csv(name, skip = 4, sep = "\t", header = FALSE)

                if(name == 'returns_annual.txt')
                    returns <- as.xts(returns[ ,-1], order.by = as.Date(as.character(returns[,1]), forma
                else
                    returns <- as.xts(returns[ ,-1], order.by = as.Date(as.character(returns[,1]), forma

                colnames(returns) <- c("SP500_rtrn", "BOND_rtrn")
                return(returns)
}

daily           <- getReturns('returns_daily.txt')
monthly         <- getReturns('returns_monthly.txt')
```

```
annually        <- getReturns('returns_annual.txt')

# Notes
# This is the same thing
# 1 - pnorm(q = (0.05 - mean(annually$SP500_rtrn)) / sd(annually$SP500_rtrn), mean = 0, sd = 1, lower.t
# pnorm(q = (0.05 - mean(annually$SP500_rtrn)) / sd(annually$SP500_rtrn), mean = 0, sd = 1, lower.tail

# Part B
# Question 1 and 2

makeAbsShortfall <- function(returns, title, sim_obs = 10000){

    makeZScore <- function(x){

        rtrn_mean <- mean(x)
        rtrn_sd <- sd(x)
        z_scores <- seq(
                    from = rtrn_mean - 2 * rtrn_sd,
                    to = rtrn_mean + 2 * rtrn_sd,
                    by = 4 * rtrn_sd / 100)
        return(z_scores)
    }

    # Question 1; compute Prob(z < Z) via pnorm
    rtrn_zscore <- makeZScore(returns)
    pvalues <- pnorm(rtrn_zscore, mean = mean(returns), sd = sd(returns))

    # Draw from standard normal; compute Prob(z < Z) via pnorm
    simulation <- rnorm(sim_obs) # draw from standard normal
    sim_zscore <- makeZScore(simulation)
    pvalues_sim <- pnorm(sim_zscore, mean = mean(simulation), sd = sd(simulation))

    # Bootstrap; compute Prob(z < Z) via pnorm
    bootstrap <- sample(x = matrix(returns), size = sim_obs, replace = TRUE)
    bootstrap_zscore <- makeZScore(bootstrap)
    pvalues_bootstrap <- pnorm(bootstrap_zscore, mean = mean(bootstrap), sd = sd(bootstrap))

    frame <- data.frame(pValue = pvalues, Actual = rtrn_zscore, Bootstrap = bootstrap_zscore, Simulatio

    p1 <- ggplot(melt(frame, id = "pValue")) +
            geom_line(aes(x = value, y = pValue, colour = variable)) +
            xlab(NULL) +
            ylab(NULL) +
            theme(axis.title.y = element_blank(), axis.text.y = element_blank(), axis.ticks.y = element_
            scale_colour_manual(values = c("salmon", "darkgrey", "black"), guide = guide_legend(title =

    p2 <- ggplot(melt(frame[,-4], id = "pValue")) +
            geom_line(aes(x = value, y = pValue, colour = variable)) +
            xlab(NULL) +
            ylab("Probability Density") +
            scale_colour_manual(values = c("salmon", "darkgrey"), guide = guide_legend(title = NULL)) +
            theme(legend.position = "none")
```

```r
    p_out <- grid.arrange(p2, p1, ncol = 2, top = title, bottom = "Absolute Shortfall Probability")
    detail <- rbind(head(frame, 3), tail(frame, 3))
    return(list(plot = p_out, detail = detail))
}

# Stocks

stock_daily_answer <- makeAbsShortfall(daily$SP500_rtrn, title = "Daily Shortfall (Stock)")
```

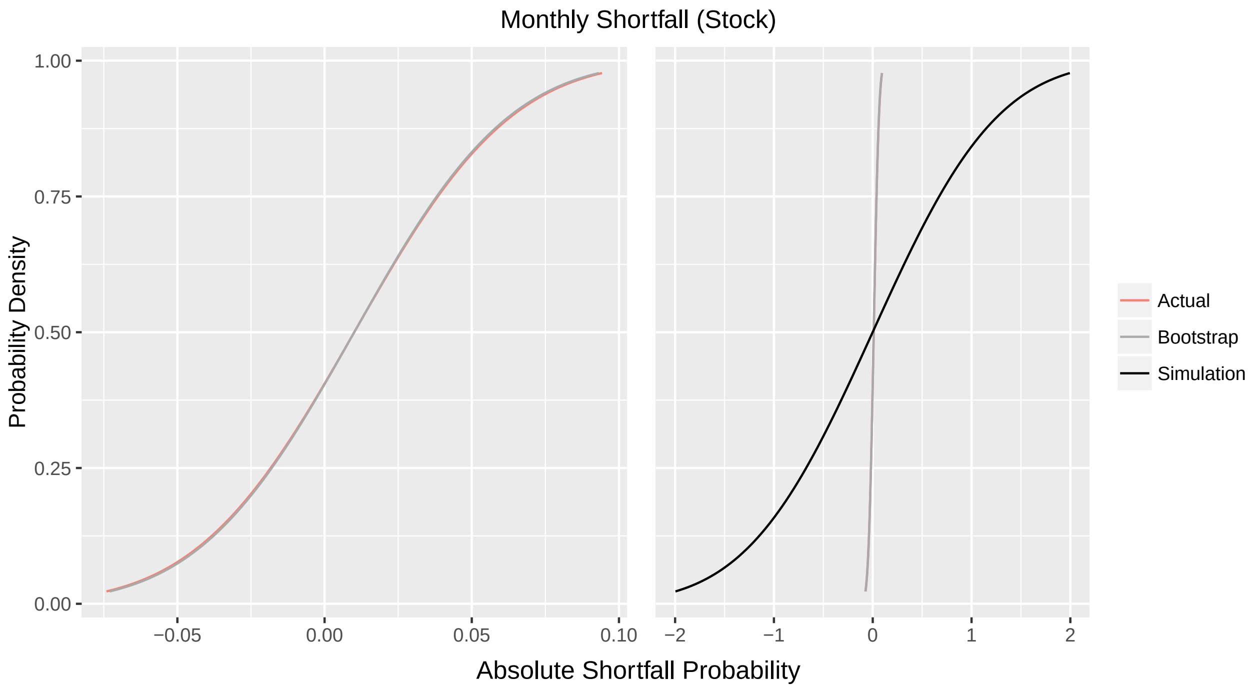


```r
kable(stock_daily_answer$detail, digits = 6, caption = "Daily")
```

Table 1: Daily

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.0199	-0.0203	-1.98
2	0.0250	-0.0195	-0.0199	-1.94
3	0.0274	-0.0191	-0.0195	-1.90
99	0.9726	0.0200	0.0202	1.93
100	0.9750	0.0204	0.0206	1.97
101	0.9772	0.0208	0.0210	2.01

```r
stock_monthly_answer <- makeAbsShortfall(monthly$SP500_rtrn, title = "Monthly Shortfall (Stock)")
```

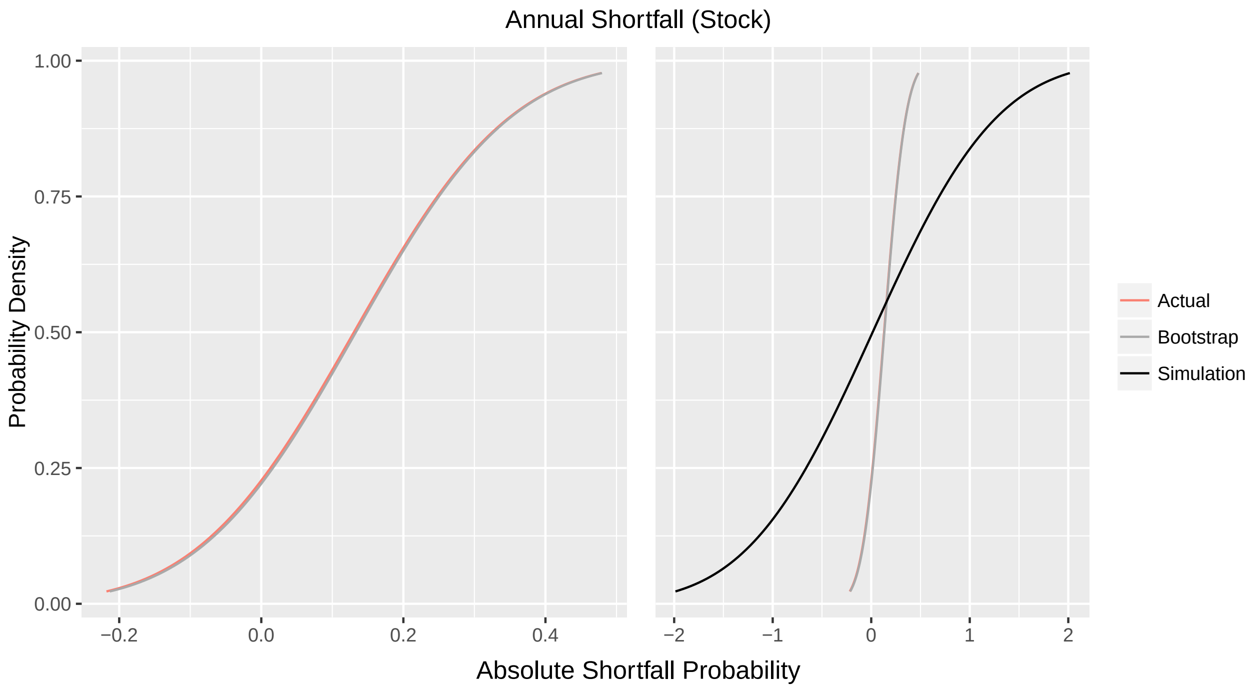


```r
kable(stock_monthly_answer$detail, digits = 6, caption = "Monthly")
```

Table 2: Monthly

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.0741	-0.0730	-2.00
2	0.0250	-0.0724	-0.0714	-1.96
3	0.0274	-0.0707	-0.0697	-1.92
99	0.9726	0.0909	0.0899	1.91
100	0.9750	0.0926	0.0915	1.95
101	0.9772	0.0942	0.0932	1.99

```r
stock_annually_answer <- makeAbsShortfall(annually$SP500_rtrn, title = "Annual Shortfall (Stock)")
```

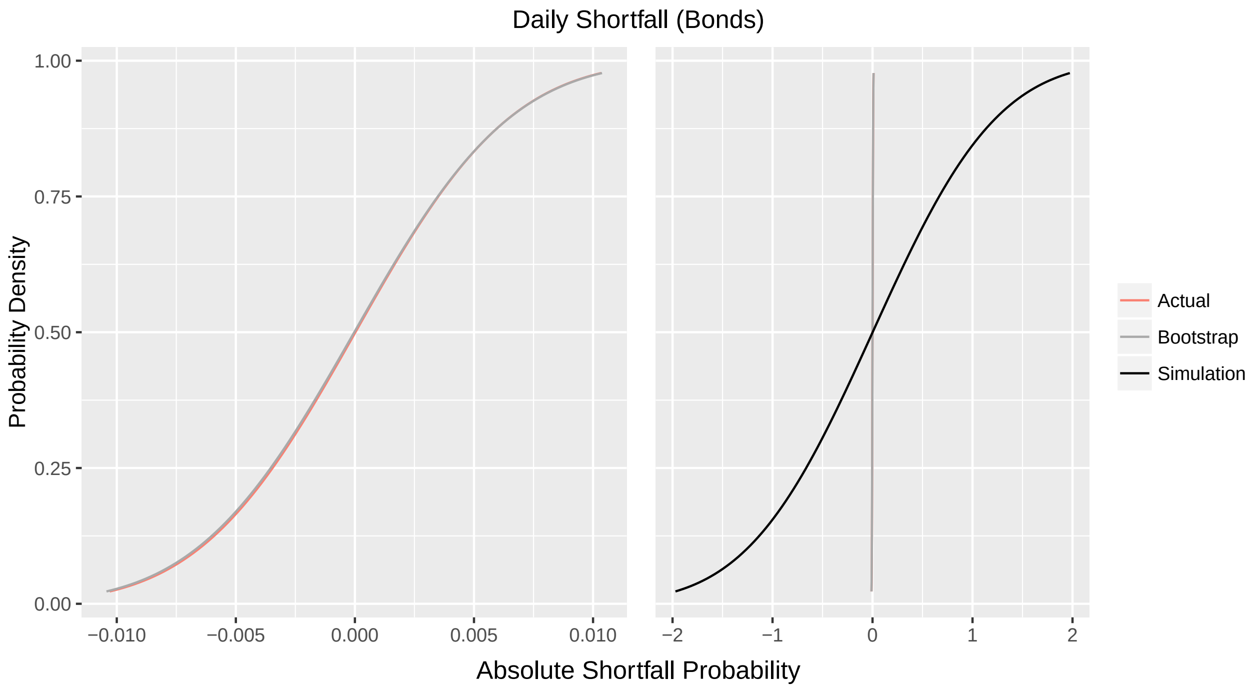


```
kable(stock_annually_answer$detail, digits = 6, caption = "Annually")
```

Table 3: Annually

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.218	-0.213	-1.99
2	0.0250	-0.211	-0.206	-1.95
3	0.0274	-0.204	-0.199	-1.91
99	0.9726	0.465	0.466	1.94
100	0.9750	0.472	0.473	1.98
101	0.9772	0.479	0.480	2.02

```
# Bonds

bond_daily_answer <- makeAbsShortfall(daily$BOND_rtrn, title = "Daily Shortfall (Bonds)")
```

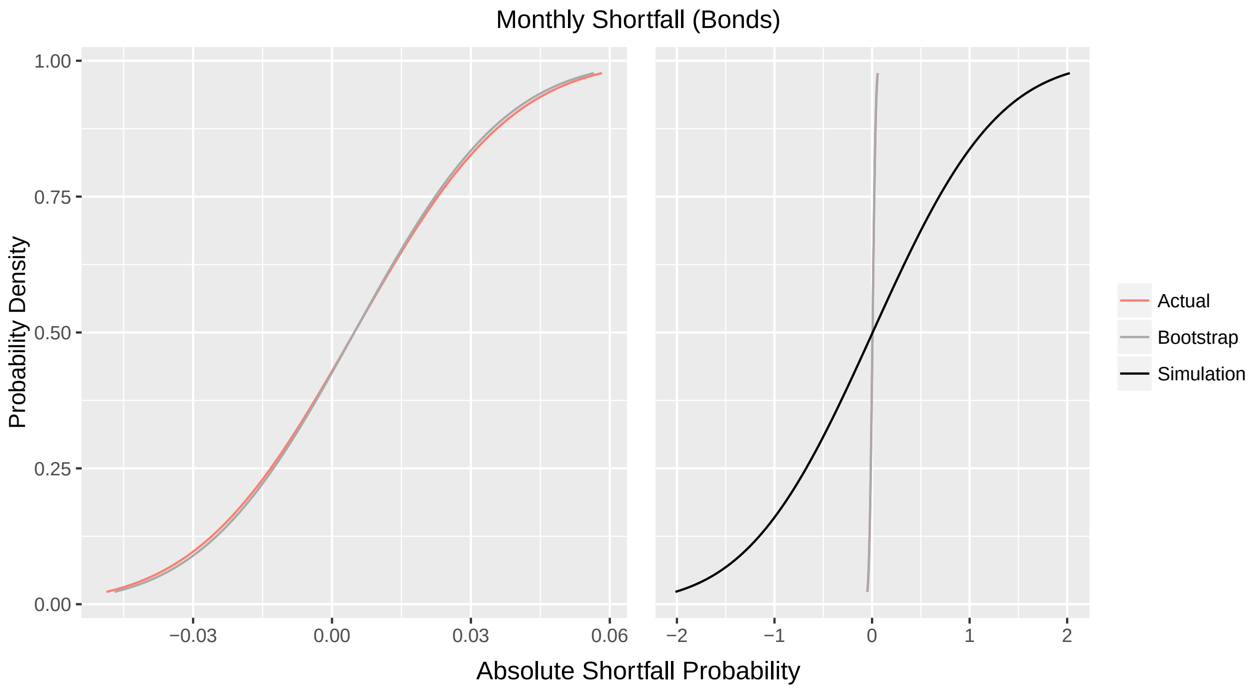


```
kable(bond_daily_answer$detail, digits = 6, caption = "Daily")
```

Table 4: Daily

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.01030	-0.01043	-1.97
2	0.0250	-0.01009	-0.01022	-1.93
3	0.0274	-0.00988	-0.01002	-1.89
99	0.9726	0.00993	0.00996	1.89
100	0.9750	0.01013	0.01017	1.93
101	0.9772	0.01034	0.01038	1.97

```
bond_monthly_answer <- makeAbsShortfall(monthly$BOND_rtrn, title = "Monthly Shortfall (Bonds)")
```

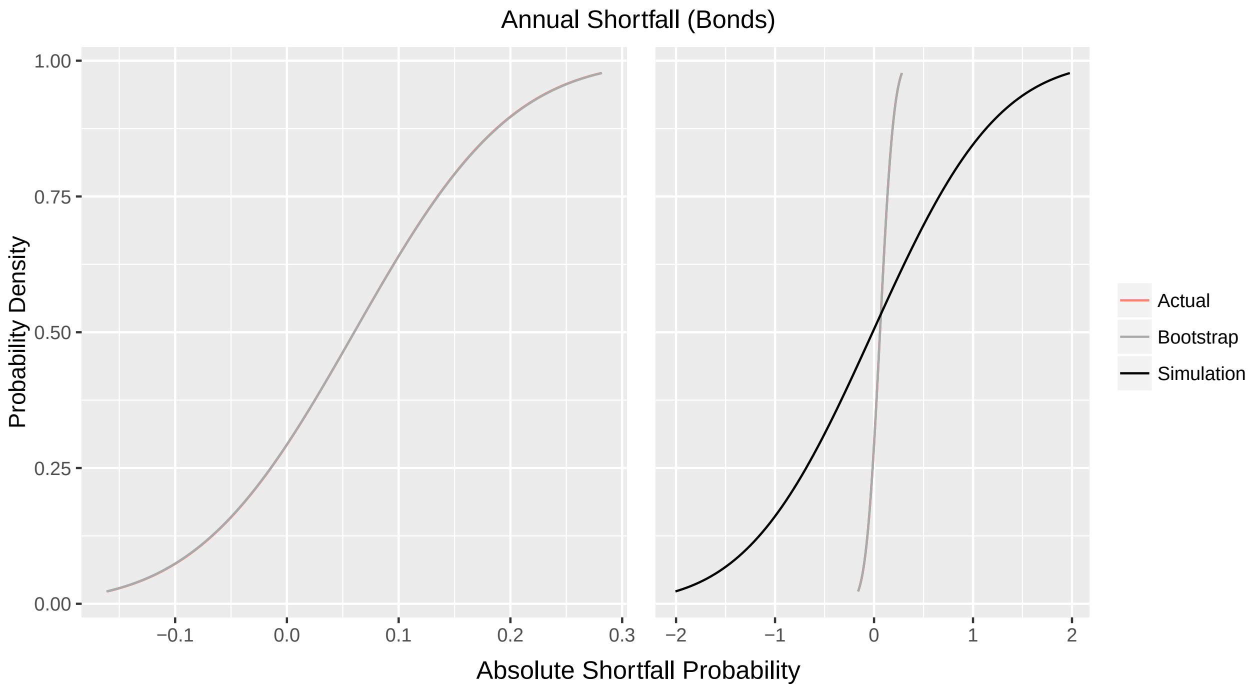


```r
kable(bond_monthly_answer$detail, digits = 6, caption = "Monthly")
```

Table 5: Monthly

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.0487	-0.0469	-2.02
2	0.0250	-0.0476	-0.0459	-1.97
3	0.0274	-0.0466	-0.0449	-1.93
99	0.9726	0.0562	0.0545	1.95
100	0.9750	0.0573	0.0555	1.99
101	0.9772	0.0583	0.0566	2.03

```r
bond_annually_answer <- makeAbsShortfall(annually$BOND_rtrn, title = "Annual Shortfall (Bonds)")
```



```
kable(stock_annually_answer$detail, digits = 6, caption = "Annually")
```

Table 6: Annually

	pValue	Actual	Bootstrap	Simulation
1	0.0227	-0.218	-0.213	-1.99
2	0.0250	-0.211	-0.206	-1.95
3	0.0274	-0.204	-0.199	-1.91
99	0.9726	0.465	0.466	1.94
100	0.9750	0.472	0.473	1.98
101	0.9772	0.479	0.480	2.02

```
# Question 3
probReturn <- function(R, K = 1.20, T = 5){ # remeber to add 1 to K because of cum return
    R <- matrix(R)
    r <- log(1 + R) # turn return series into cum return
    mu <- mean(r)
    sigma2 <- var(r)

    # Prob Vt < K
    1 - pnorm(log(K), mean = mu * T, sd = sqrt(sigma2) * sqrt(T))

    # formulaic approach; Prob(z < ln(K) - Tu / sqrt(T) *(sigma))
    Z <- (log(K) - T * mu) / (sqrt(T) * sqrt(sigma2))
    out <- as.numeric(1 - pnorm(Z))
    return(out)
}

makeTable <- function(parent_function){
    out <- cbind.data.frame(
            stocks = rbind(
                eval(parent_function(R = annually$SP500_rtrn)),
                eval(parent_function(R = monthly$SP500_rtrn)),
                eval(parent_function(R = daily$SP500_rtrn))
                ),

            bonds = rbind(
                eval(parent_function(R = annually$BOND_rtrn)),
                eval(parent_function(R = monthly$BOND_rtrn)),
                eval(parent_function(R = daily$BOND_rtrn))),
            row.names = c("Annually", "Monthly", "Daily")
    )

    out <- kable(out, digits = 6)
    return(out)
}

makeTable(probReturn)
```

	stocks	bonds
Annually	0.8405	0.64526
Monthly	0.0738	0.00344
Daily	0.0000	0.00000

```
# Question 4
simKnownDist <- function(R, K = 1.20, T = 5, sim_obs = 10000){

    R_sims <- replicate(sim_obs, rnorm(T, mean = mean(R), sd = sd(R))) # sim_obs (n) draws from standar

    Vt <- apply(R_sims, 2, function(x) prod(x + 1)) # prod return series; each simulation addes 1 (beca
    Vt_log <- apply(R_sims, 2, function(x) exp(sum(log(x + 1)))) # from log'ed return series; now logge
    stopifnot(all.equal(Vt, Vt_log))

    prob_Vt_greater_than_K <- length(which(Vt_log > K)) / sim_obs # objective; since question asks for
    return(prob_Vt_greater_than_K)
}

makeTable(simKnownDist)
```

	stocks	bonds
Annually	0.8549	0.6380
Monthly	0.0684	0.0036
Daily	0.0000	0.0000

```
# Question 5
simBootstrap <- function(R, K = 1.20, T = 5, sim_obs = 10000){

    R_sims <- replicate(sim_obs, sample(R, T), simplify = FALSE) # sim_obs (n) draws from data(bootstra
    Vt <- lapply(R_sims, function(x) prod(x + 1)) # prod return series; each simulation addes 1 (becaus
    Vt_log <- lapply(R_sims, function(x) exp(sum(log(x + 1)))) # from log'ed return series; now logged 
    stopifnot(all.equal(Vt, Vt_log))

    prob_Vt_greater_than_K <- length(which(Vt_log > K)) / sim_obs # objective; since question asks for
    return(prob_Vt_greater_than_K)
}

makeTable(simBootstrap)
```

	stocks	bonds
Annually	0.8478	0.6367
Monthly	0.0629	0.0062
Daily	0.0000	0.0000

```
# Question 6
stock.VS.bonds.Analytical <- function(Ra, Rb, T = 30, sim_obs = 10000){

    Ra <- matrix(Ra)
        ra <- log(1 + Ra) # turn return series into cum return
        mu_a <- mean(ra)
        sigma2_a <- var(ra)

    Rb <- matrix(Rb)
        rb <- log(1 + Rb)
        mu_b <- mean(rb)
        sigma2_b <- var(rb)

    rho <- cor(Ra, Rb) # rho; not used
    E_delta <- T * (mu_a - mu_b)
    sigma2_delta <- T * (sigma2_a - sigma2_b)

    # Prob(z < Z)
    Z <- -E_delta / sqrt(sigma2_delta)
    out <- pnorm(Z)
    return(out)
}

makeTable2 <- function(parent_function, ...){

    out <- rbind.data.frame(
                eval(parent_function(
                    Ra = daily$SP500_rtrn,
                    Rb = daily$BOND_rtrn,
                    ... = ...
                )),
                eval(parent_function(
                    Ra = monthly$SP500_rtrn,
                    Rb = monthly$BOND_rtrn,
                    ... = ...
                )),
                eval(parent_function(
                    Ra = annually$SP500_rtrn,
                    Rb = annually$BOND_rtrn,
                    ... = ...
                ))
            )
    colnames(out) <- paste("T = ", ..., sep = "")
    rownames(out) <- c("Daily", "Monthly", "Annually")
    out <- kable(out, digits = 6)
    return(out)
}

makeTable2(stock.VS.bonds.Analytical, T = 5)
```

	T = 5
Daily	0.465
Monthly	0.374
Annually	0.165

```
makeTable2(stock.VS.bonds.Analytical, T = 30)
```

	T = 30
Daily	0.41387
Monthly	0.21615
Annually	0.00855

```
makeTable2(stock.VS.bonds.Analytical, T = 100)
```

	T = 100
Daily	0.345576
Monthly	0.075834
Annually	0.000007

```
# Question 7
stock.VS.bonds.Bootstrap <- function(Ra, Rb, T = 30, sim_obs = 10000){

    sample_indices <- replicate(sim_obs, sample(index(Ra), T), simplify = FALSE) # sim_obs (n) draws fr

    sim_samples <- lapply(sample_indices, function(x) cbind(Ra[x], Rb[x])) # get the difference in retu
    sim_cumreturns <- lapply(sim_samples, function(x) c(prod(x$SP500_rtrn + 1), prod(x$BOND_rtrn + 1)))

    prob_Vs_greater_than_Vb <- length(which(unlist(lapply(sim_cumreturns, function(x) x[1] < x[2])))) /
    return(prob_Vs_greater_than_Vb)
}

out <- cbind(
    stock.VS.bonds.Bootstrap(Ra = daily$SP500_rtrn, Rb = daily$BOND_rtrn),
    stock.VS.bonds.Bootstrap(Ra = monthly$SP500_rtrn, Rb = monthly$BOND_rtrn),
    stock.VS.bonds.Bootstrap(Ra = annually$SP500_rtrn, Rb = annually$BOND_rtrn))

rownames(out) <- "T = 30"
kable(t(rbind(c("Daily", "Monthly", "Annually"), out)),  digits = 6)
```

	T = 30
Daily	0.4246
Monthly	0.2779
Annually	0.0184

```
# Part C
# Question 1
mu <- 0.10
sigma2 <- 0.2
T <- 50 - 35

car <- 1e5
target <- 1e6
savings <- 5e5 - car

Z <- (log(target / savings) - T * mu) / (sqrt(T) * sqrt(sigma2))
pnorm(Z) # probability of less than target amount: we cannot afford any of the cars
```

```
## [1] 0.368
```

```
# Question 2
b_mu <- 0.003
b_sigma2 <- 0.015
b_sigma2 <- 0.3

# Part A
T <- 10 * 12 # in months
rf_mu <- 0.003
rf_sigma2 <- 0 # risk free

Z <- sqrt(T) * (rf_mu - b_mu) / sqrt(b_sigma2)
1 - pnorm(Z)
```

```
## [1] 0.5
```

```
# Part B
# Generally, as the length of the time horizon grows(T),
# the probability that the risk free asset will outperform the risky asset decreases; that is, the prob
# However, here rf_mu = b_mu, thus the result is indepedent of the time horizon.
# Again, as for volatility, rf_mu = b_mu, and is equal to 0, so the numeritor becomes 0, making the res

# Question 3
# TO DO
# Question 4
# TO DO

# Question 5
# Part A
# No I do not agree with Bill Gross. Assuming returns are i.i.d. (But not necessarily normal). By pure
# to observe a "long-run" average of high returns from stocks. That is, we could be observing returns o
# distribution without calling into question any of our underlying assumptions. What would be flawed ho
# overcorrect and start 'drawing' returns from the left side of the mean of the distribution. This is s
# begin to observe more negative returns from stocks; but as our homework discussed this noise will be
# probability of one.

# Part B
# While "pure intellictual fraud" is a bit too harsh, I do argee with Nassim that the heuristic is misl
# I believe that managers should use it as one metric, or one tool in their tool box to get a snapshot
# is folly; and one, instead, should use a dashboard of risk assessment tools.
```