Integrated Management Formulation Model

Kirui Kipngeno

October 19, 2018

OPTIMIZATION WITH ONLY RISKY ASSETS

In this problem, we are trying to do portfolio optimization with only 4 assets with all of them being risky assets. What we are trying to see is which assets will have a higher allocations by the model i.e, we want to see the weights that the model will allocate to each of the assets in our portfolio. The process involves obtaining the data from Yahoo Finance and then modifying the data to obtain the portfolio returns which are very important in the model.

Another important thing is that, we are going to solve the problem in different levels of gamma as well as different levels of alpha. Alpha is between 0 and 1 and gamma is a vector of 0,0.1,0.3,0.5,0.7 and 0.9.

The problem to be solved is

$\max_{(x,z,q)}(1-\gamma)\cdot p^\top \Xi \cdot x - \gamma \cdot q - \frac{\gamma}{1-\alpha}\cdot p^\top z$

subtect to:

$-q - \xi_i x \leq z_i$

$x^\top 1 \leq 1, z \geq 0, (x \geq 0)$

and $\gamma, \alpha \in (0,1)$.

Loading all the necessary packages

We first load all the packages that are required for tzhe implementation of our codes throughout the program. The main packages that we need are quantmod, PerformanceAnalytics and linprog. They are the core of our problem. The others are just binary packages that will help us in the implementation.

```
suppressMessages(library(phonTools))
suppressMessages(library(optimbase))
suppressMessages(library(pracma))
suppressMessages(library(quantmod))
suppressMessages(library(PerformanceAnalytics))
suppressMessages(library(linprog))
suppressMessages(library(ggplot2))
```

Obtaining the data

The data to be obtained is in the range of 2014 and 2018 considering the stocks DIS, BABA, JNJ and FB. We only want the adjusted closing prices of our stocks and so those are the ones we consider without missing values. Using the function ROC from the package TTR, we calculate the returns of our data and then annualize the results. We consider equal probabilities for all the trading days.

```
begin <- "2014-01-01" #first day of my trading perios
end <- "2018-01-01" #last day of my trading period
stocks <- c("DIS","BABA","JNJ","FB","BOND") #stocks in my portfolio
#BOND is the risk less asset
suppressMessages(getSymbols(stocks,from=begin, to = end)) #pulling requests
```

```
## [1] "DIS"  "BABA" "JNJ"  "FB"   "BOND"
```

```
Portprices <- na.omit(merge(Ad(DIS),Ad(BABA),Ad(JNJ),Ad(FB)))
Portpricesfree <- na.omit(merge(Ad(DIS),Ad(BABA),Ad(JNJ),Ad(FB),Ad(BOND)))
names(Portprices) <- c("DIS","BABA","JNJ","FB")
names(Portpricesfree) <- c("DIS","BABA","JNJ","FB","BOND")
#calculating the Portfolio returns and normalizing
Portreturns <- ROC(Portprices,type = "discrete")[-1,]*365
Portreturnsfree <- ROC(Portpricesfree,type = "discrete")[-1,]*365
#Renaming the columns
colnames(Portreturns) <- c("DIS","BABA","JNJ.","FB")
colnames(Portreturnsfree) <-  c("DIS","BABA","JNJ","FB","BOND")
#converting the data into a matrix
Portreturns <- as.matrix(Portreturns)
Portreturnsfree <- as.matrix(Portreturnsfree)
#checking the dimensions
dim(Portreturns) #without the risk free asset
```

```
## [1] 826   4
```

```
dim(Portreturnsfree) #with risk free asset
```

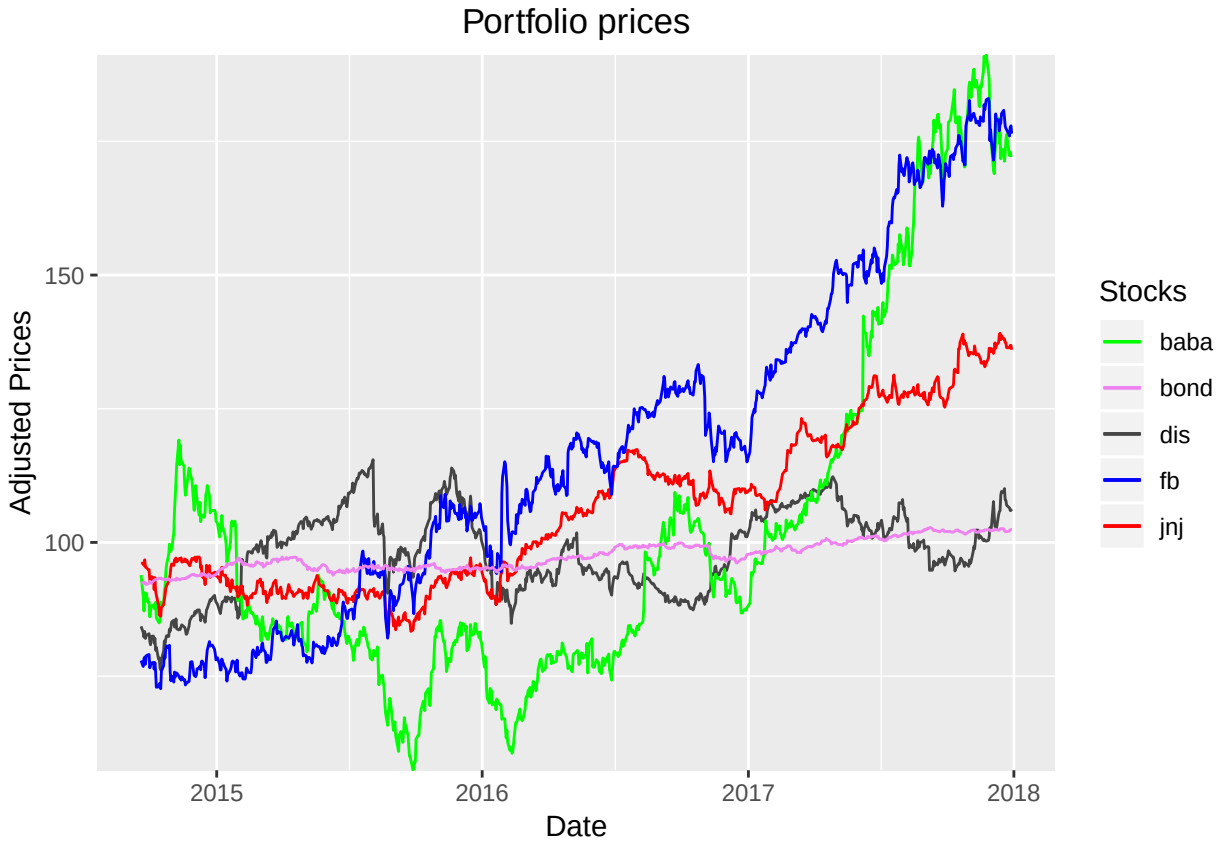
```
## [1] 826   5
```

```
#Equal probability for each trading day
prob <- rep(1/826,826)
```

Having obtained the above data, we plot to see the evolvement of the stocks especially looking at the shocks on the various stocks

Plot for the stocks evolvement

```
ggplot(Portpricesfree, aes(x = index(Portpricesfree))) +
  geom_line(aes(y = Portpricesfree$DIS,color = "dis")) +
  ggtitle("Portfolio prices") +
  geom_line(aes(y = Portpricesfree$BABA, color = "baba")) +
  geom_line(aes(y = Portpricesfree$JNJ, color = "jnj")) +
  geom_line(aes(y = Portpricesfree$FB, color = "fb")) +
  geom_line(aes(y = Portpricesfree$BOND, color = "bond")) +
  xlab("Date") + ylab("Adjusted Prices") +
  theme(plot.title = element_text(hjust = 0.5), panel.border = element_blank()) +
  scale_y_continuous(expand = c(0,0)) +
  scale_colour_manual("Stocks",values=c("jnj"="#FF0000","baba"="#00FF00",
                                        "fb"="#0000FF","dis"="#454545","bond" = "violet"))
```



We are going to look on the solution of the problem when gamma = 0.1,0.3,0.5,0.7 and 0.9 to understand the differences in these levels.

Solving the problem in linprog needs the definition of the Amat matrix, object vector and the constraint vector as in $Ax \leq b$ where A is the Amat matrix, x is the vector to be solved and b is the right hand solution. Object vector is deefined by the objective function of the problem. (see solveLP function in R from package **linprog**.)

We want to see what happens for the stocks in the changing levels of γ as the levels of α increases.

The **Amat** matrix doesn't change in all the levels of γ and so once defined, we will just use the same matrix for all the levels. The **cvec** vector changes and **bvec** vector remains the same through all considered levels of γ.

With gamma=0.1

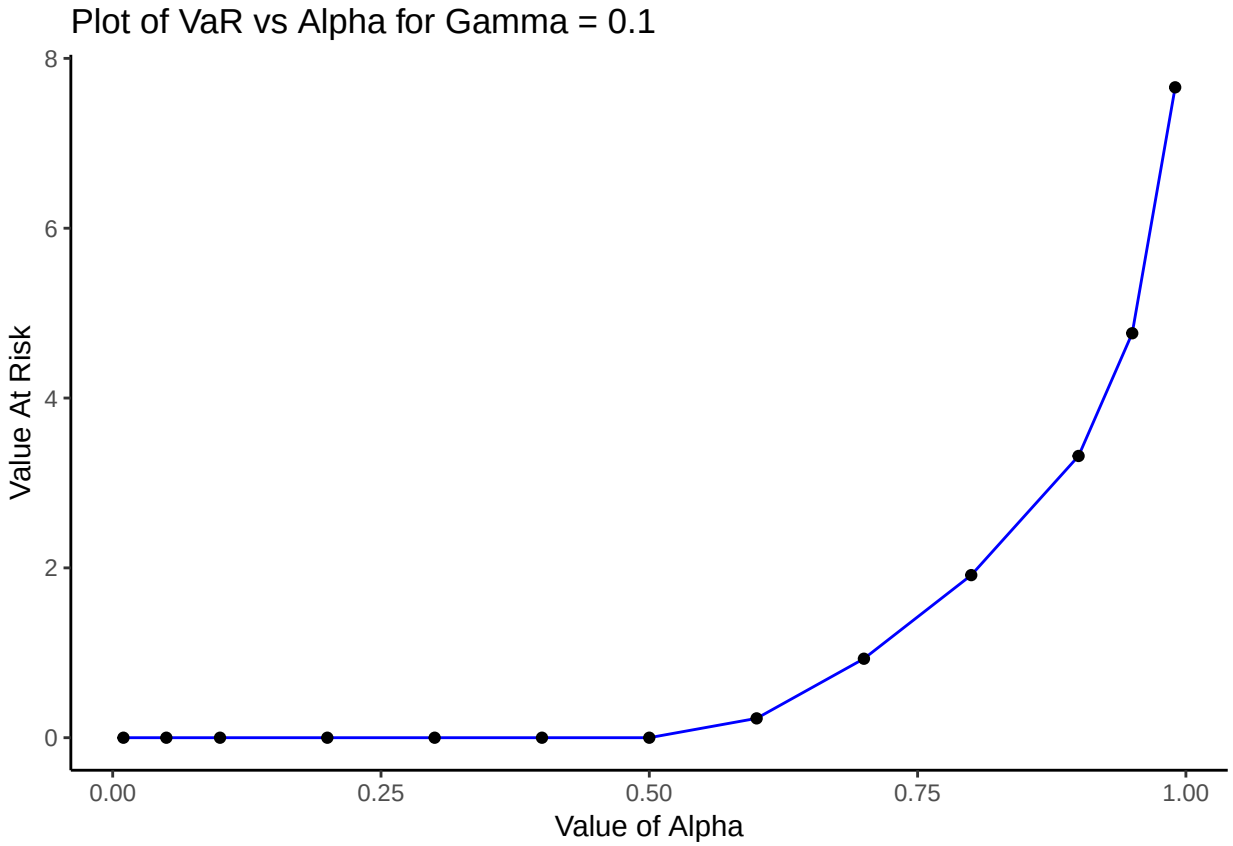
```
gamma1 = 0.1
alpha = c(0.01,0.05,0.1,0.2,0.3,0.4,0.50,0.60,0.70,0.80,0.90,0.95,0.99)
#creating empty vectors to hold the results
dis_result = vector(mode = "numeric")
baba_result = vector(mode = "numeric")
jnj_result = vector(mode = "numeric")
fb_result = vector(mode = "numeric")
q_result = vector(mode = "numeric")
solution1 = vector(mode = "numeric")
#looping over all the levels of alpha and filling up the empty vectors
for (alp in alpha){
```

```
  objvect <- c((1-gamma1)*t(prob)%*%Portreturns,(-gamma1/(1-alp))*t(prob),-gamma1)
  names(objvect)<-c("dis","baba","jnj","fb",rep("z",826),"q")
  rhscons <- c(rep(0,826 ),1,rep(0,830))
  #Construction of matrix Amat
  firstcons = cbind(-Portreturns,-diag(826),-rep(1,826))
  secondcons = (c(rep(1,4),rep(0,827)))
  lessx = cbind(-diag(4),zeros(4,827))
  lessz = cbind(zeros(826,4),-diag(826),rep(0,826))
  Amat = rbind(firstcons,secondcons,lessx,lessz)
  dim(Amat)
  colnames(Amat) = NULL
  rownames(Amat) = NULL
  Solution1 <- solveLP(objvect,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result<-append(dis_result,Solution1$solution[1])
  baba_result<- append(baba_result,Solution1$solution[2])
  jnj_result<-append(jnj_result,Solution1$solution[3])
  fb_result<-append(fb_result,Solution1$solution[4])
  q_result <- append(q_result,Solution1$solution[831])
  solution1 <- append(solution1,Solution1$opt)
}
```

The value q in the problem is the Value-at-Risk and so it is important to look at it also.

Plot of alpha vs q (Value-at-Risk)

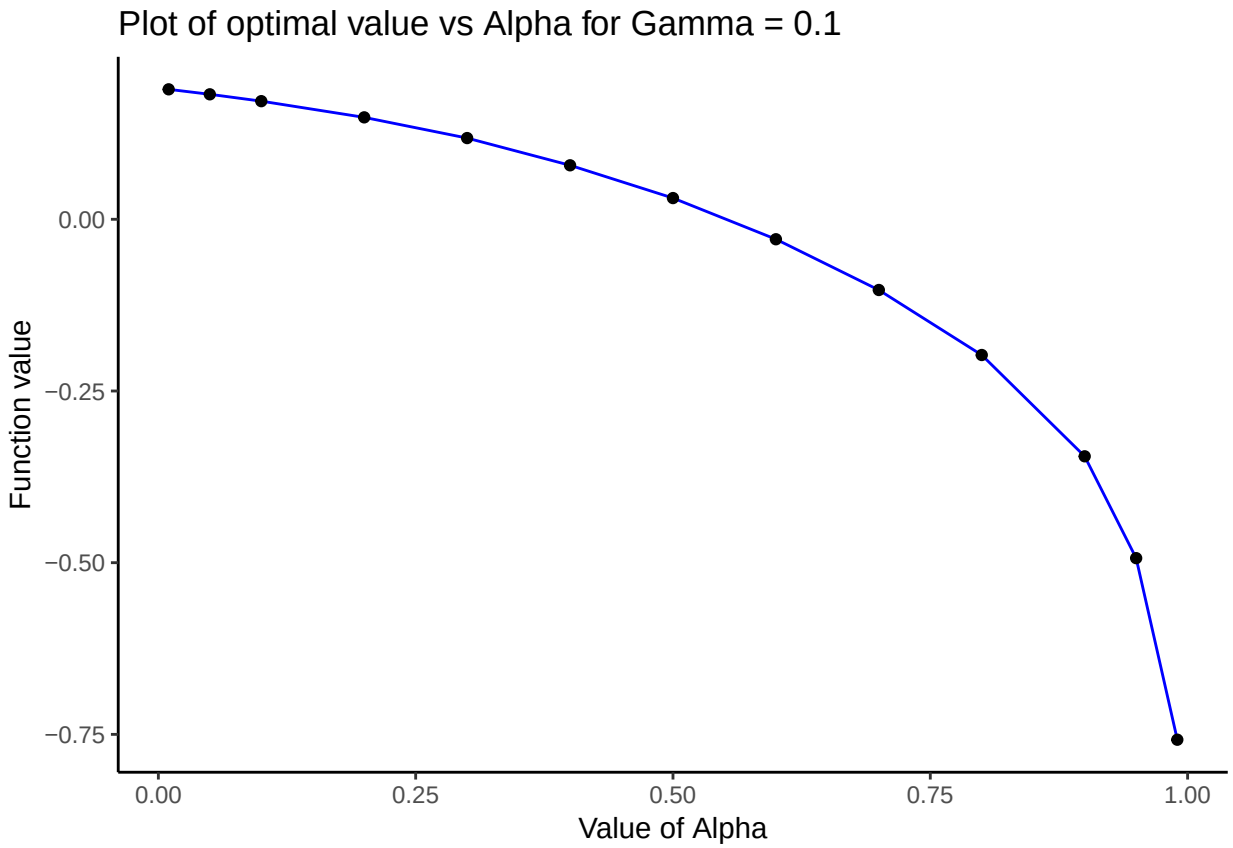
```
#creating a dataframe and writing the results into a CSV file
result_q <- data.frame(alpha,q_result)
write.csv(result_q,file = "VaR_Results.csv")
#Plotting
ggplot(data=result_q, aes(x=alpha, y=q_result, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.1 ",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



It is important to look at the development of the optimal value in the various levels of alpha also.

Plot of alpha vs optimal value

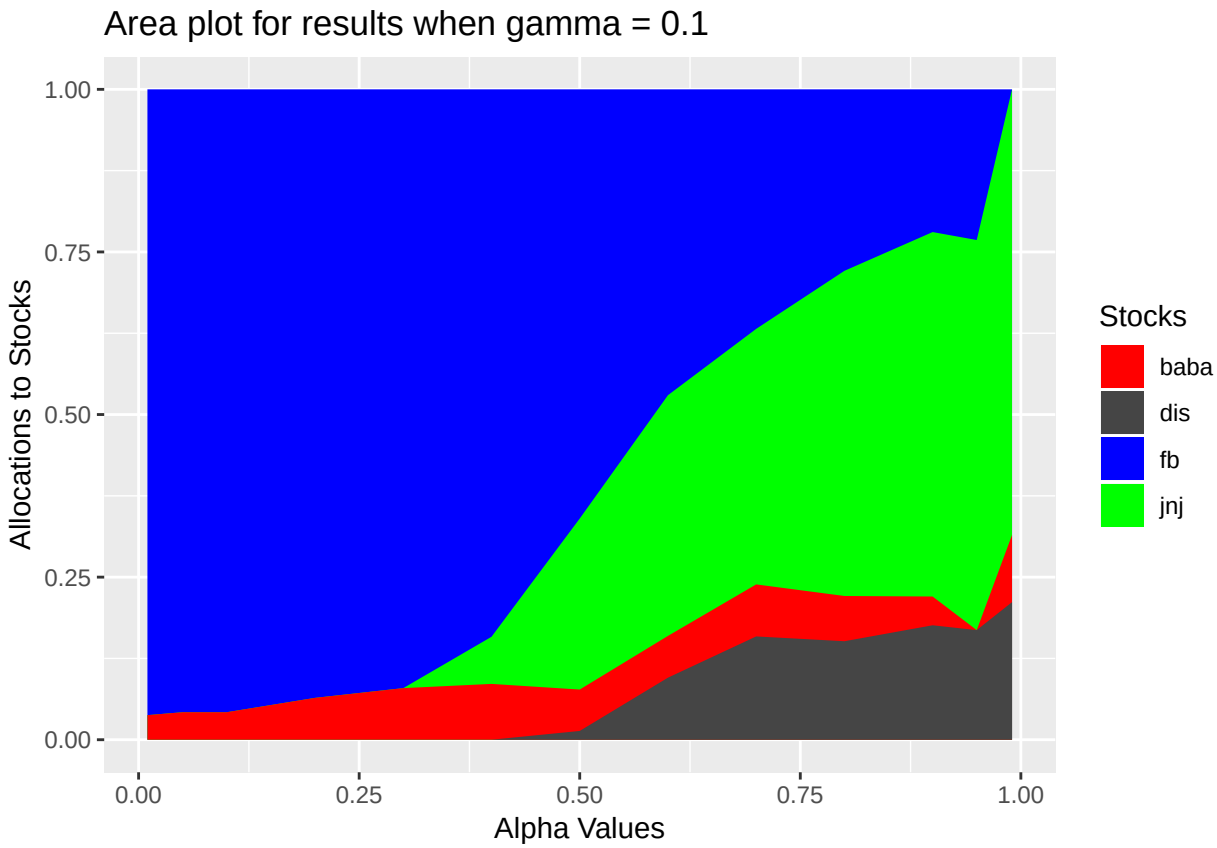
```
result_sol <- data.frame(alpha,solution1)
write.csv(result_sol,file = "Solution_fun.csv")
ggplot(data=result_sol, aes(x=alpha, y=solution1, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of optimal value vs Alpha for Gamma = 0.1 ",x="Value of Alpha", y = "Function value")
  theme_classic()
```



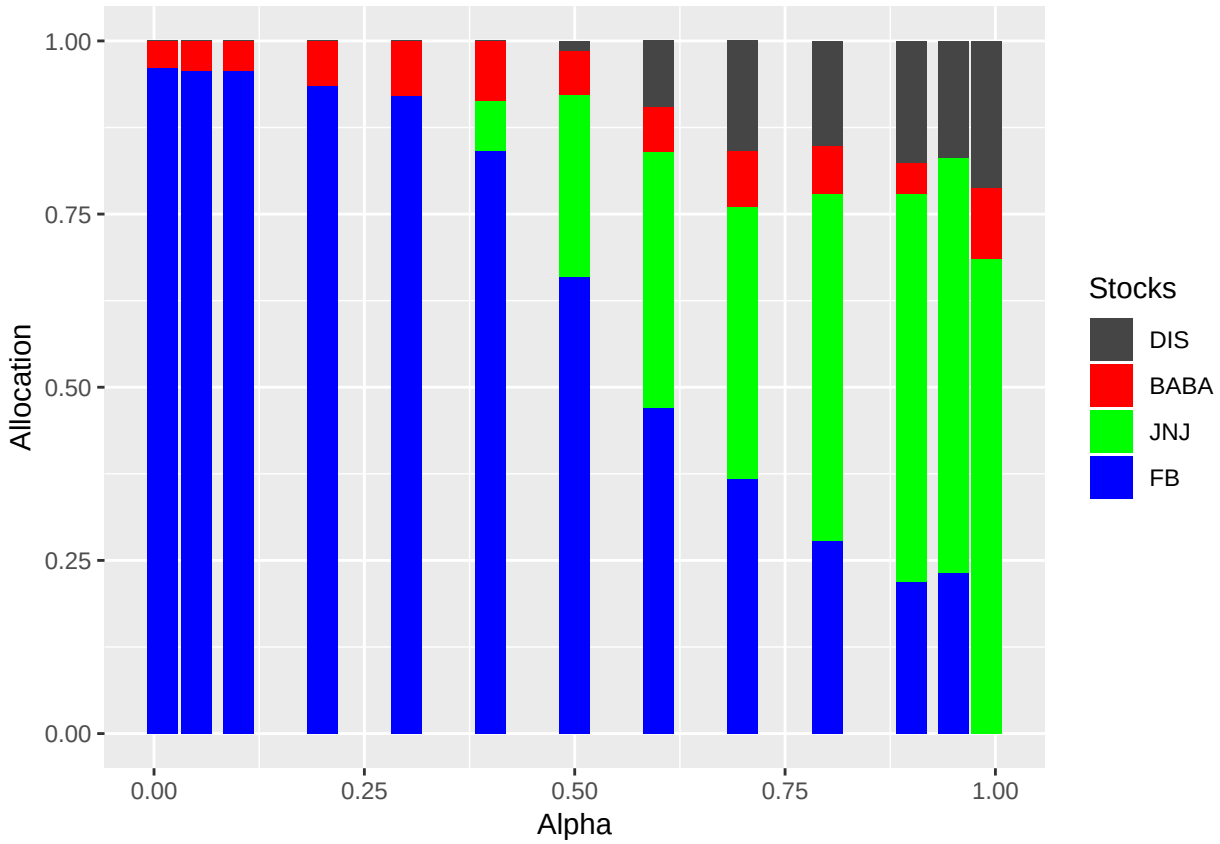
The asset allocation weights of the problem wehn gamma = 0.1 is as follows

Asset allocation for $\gamma = 0.1$

```
result1 <-data.frame(alpha,dis_result,baba_result,jnj_result,fb_result)
write.csv(result1, file = "Results1.csv")
ggplot(result1, aes(x=result1$alpha)) +
  geom_area(aes(y=result1$dis_result+result1$baba_result+result1$jnj_result+
                  result1$fb_result, fill="fb"))+
  geom_area(aes(y=result1$dis_result+result1$baba_result+result1$jnj_result, fill="jnj")) +
  geom_area(aes(y=result1$dis_result+result1$baba_result, fill="baba")) +
  geom_area(aes(y=result1$dis_result,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations to Stocks") +
  labs(title="Area plot for results when gamma = 0.1") +
  scale_fill_manual("Stocks",values =
                      c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"))
```




```
# Using stacked barplots
library(reshape2)
data1 <- result1
names(data1) <- c("alpha","DIS","BABA","JNJ","FB")
mdata <- melt(data1, id=c("alpha"))
names(mdata) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata, stat="identity")+
  scale_fill_manual("Stocks",values =
                      c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545"))
```

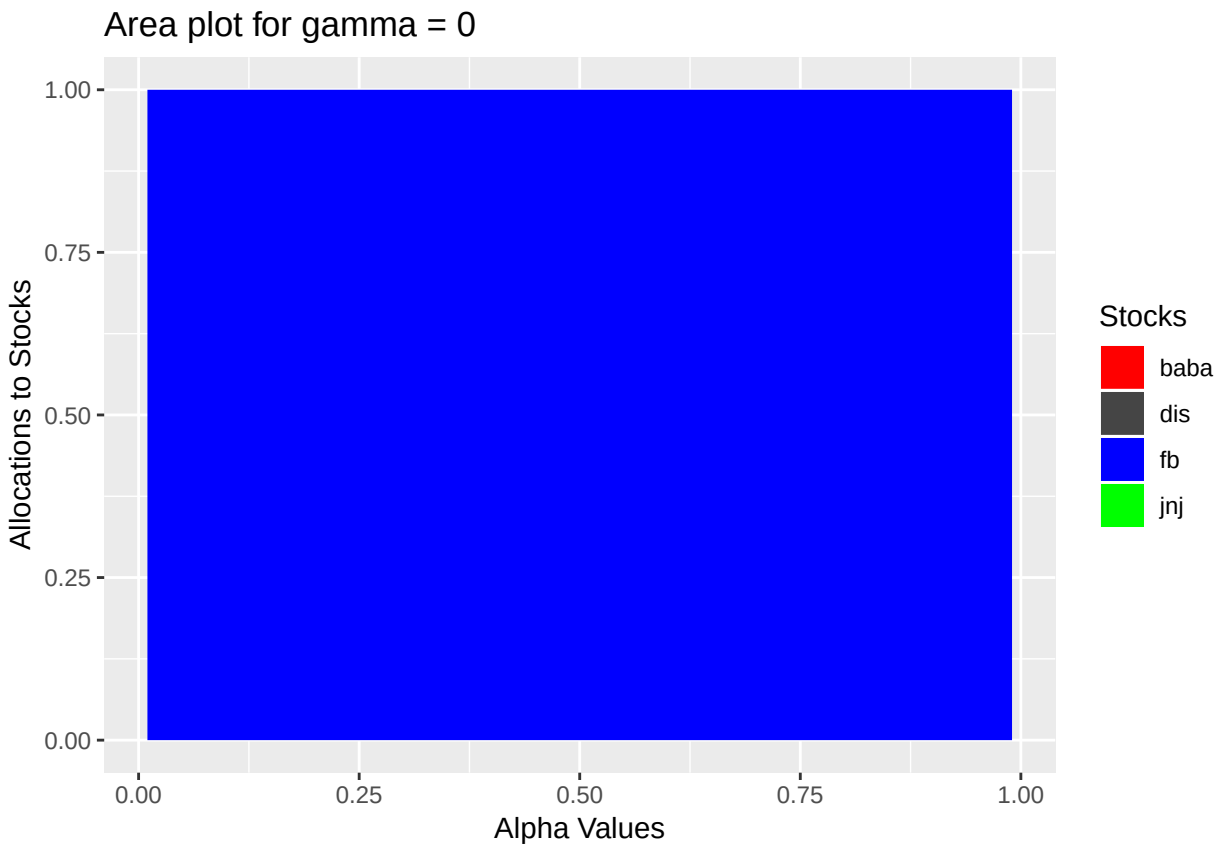


With gamma=0

```
gamma0 = 0.0
dis_result0 = vector(mode = "numeric")
baba_result0 = vector(mode = "numeric")
jnj_result0 = vector(mode = "numeric")
fb_result0 = vector(mode = "numeric")
q_result0 = vector(mode = "numeric")
for (alp in alpha){
  objvect0 <- c((1-gamma0)*t(prob)%*%Portreturns,(-gamma0/(1-alp))*t(prob),-gamma0)
  names(objvect0)<-c("dis","baba","jnj","fb",rep("z",826),"q")

  Solution0 <- solveLP(objvect0,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result0 <- append(dis_result0,Solution0$solution[1])
  baba_result0 <- append(baba_result0,Solution0$solution[2])
  jnj_result0 <- append(jnj_result0,Solution0$solution[3])
  fb_result0<- append(fb_result0,Solution0$solution[4])
  q_result0 <- append(q_result0,Solution0$solution[831])
}
```

Asset allocation for $\gamma = 0.0$

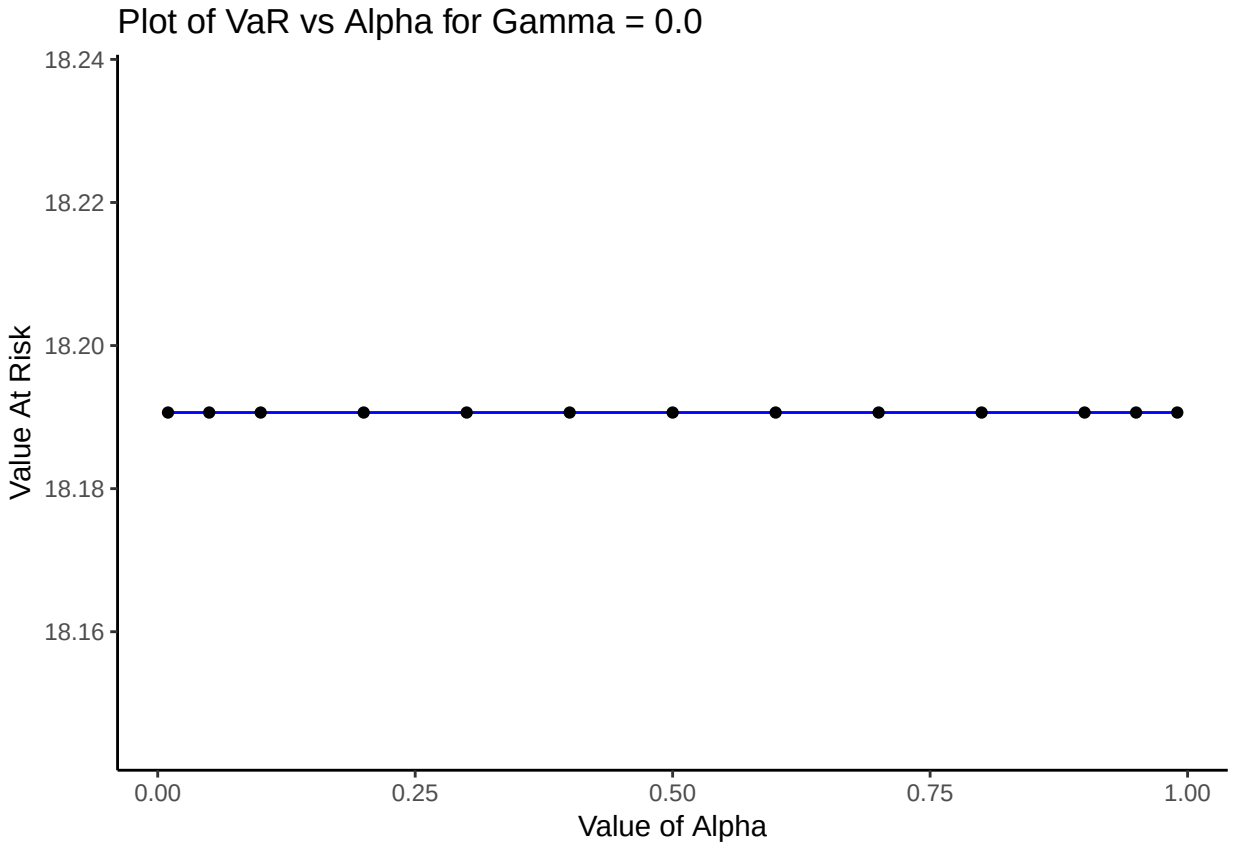
```
result0 <-  data.frame(alpha,dis_result0,baba_result0,jnj_result0,fb_result0)
write.csv(result0, file = "Results0.csv")
ggplot(result0, aes(x=result0$alpha)) +
  geom_area(aes(y=result0$dis_result0+result0$baba_result0+result0$jnj_result0+result0$fb_result0, fill=
  geom_area(aes(y=result0$dis_result0+result0$baba_result0+result0$jnj_result0, fill="jnj")) +
  geom_area(aes(y=result0$dis_result0+result0$baba_result0, fill="baba")) +
  geom_area(aes(y=result0$dis_result0,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations to Stocks") +
  labs(title="Area plot for gamma = 0") +  # title and caption
  scale_fill_manual("Stocks",values = c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"
```



Note that the choice $\gamma = 0$ identifies the stock with the highest return during the considered period and puts all possible weights onto this stock. In this case, the stock itself is FB.

Plot of alpha vs q for gamma = 0

```
result_q0 <- data.frame(alpha,q_result0)
write.csv(result_q0,file = "VaR_Results0.csv")
ggplot(data=result_q0, aes(x=alpha, y=q_result0, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.0 ",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



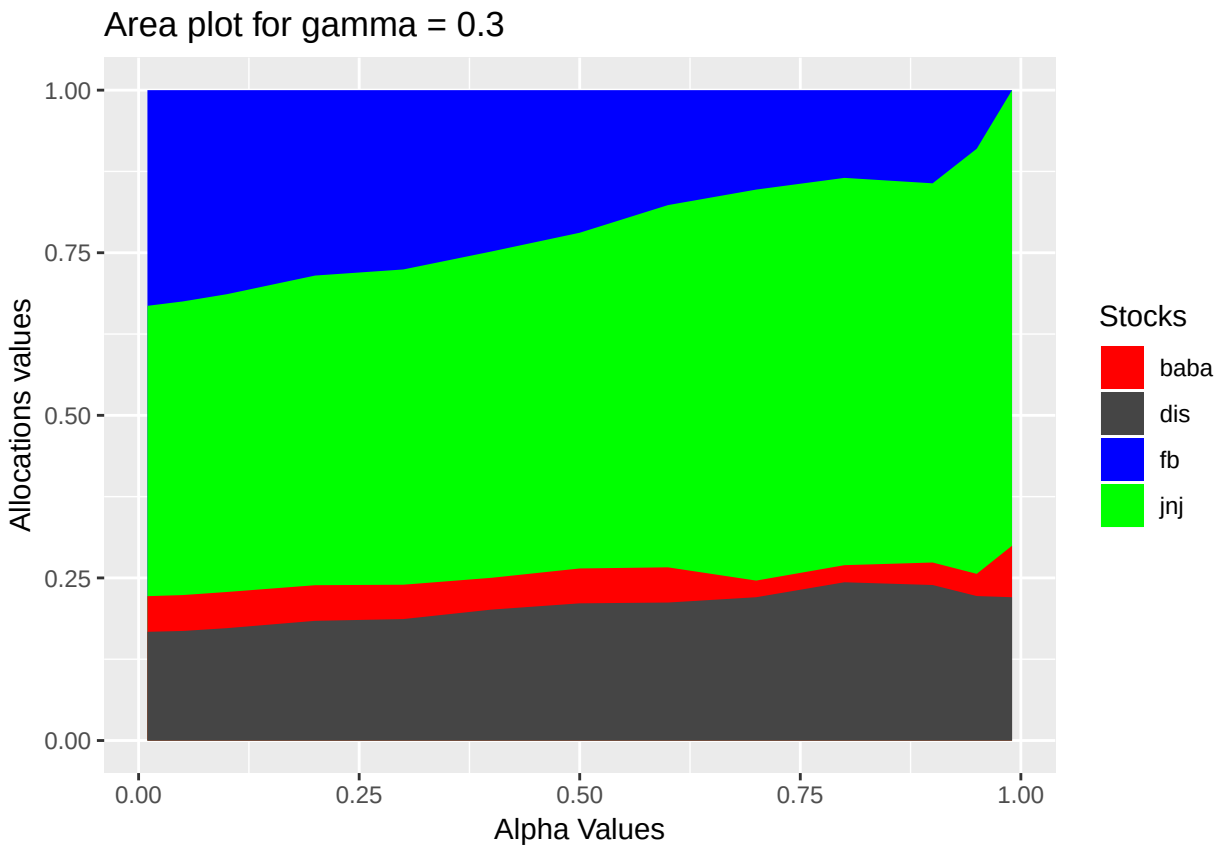
With gamma = 0.3

```
gamma3 = 0.3
dis_result3 = vector(mode = "numeric")
baba_result3 = vector(mode = "numeric")
jnj_result3 = vector(mode = "numeric")
fb_result3 = vector(mode = "numeric")
q_result3 = vector(mode = "numeric")
for (alp in alpha){
  objvect3 <- c((1-gamma3)*t(prob)%*%Portreturns,(-gamma3/(1-alp))*t(prob),-gamma3)
  names(objvect3)<-c("dis","baba","jnj","fb",rep("z",826),"q")
  Solution3 <- solveLP(objvect3,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result3 <- append(dis_result3,Solution3$solution[1])
  baba_result3 <- append(baba_result3,Solution3$solution[2])
  jnj_result3 <- append(jnj_result3,Solution3$solution[3])
  fb_result3<- append(fb_result3,Solution3$solution[4])
  q_result3 <- append(q_result3,Solution3$solution[831])
}
```

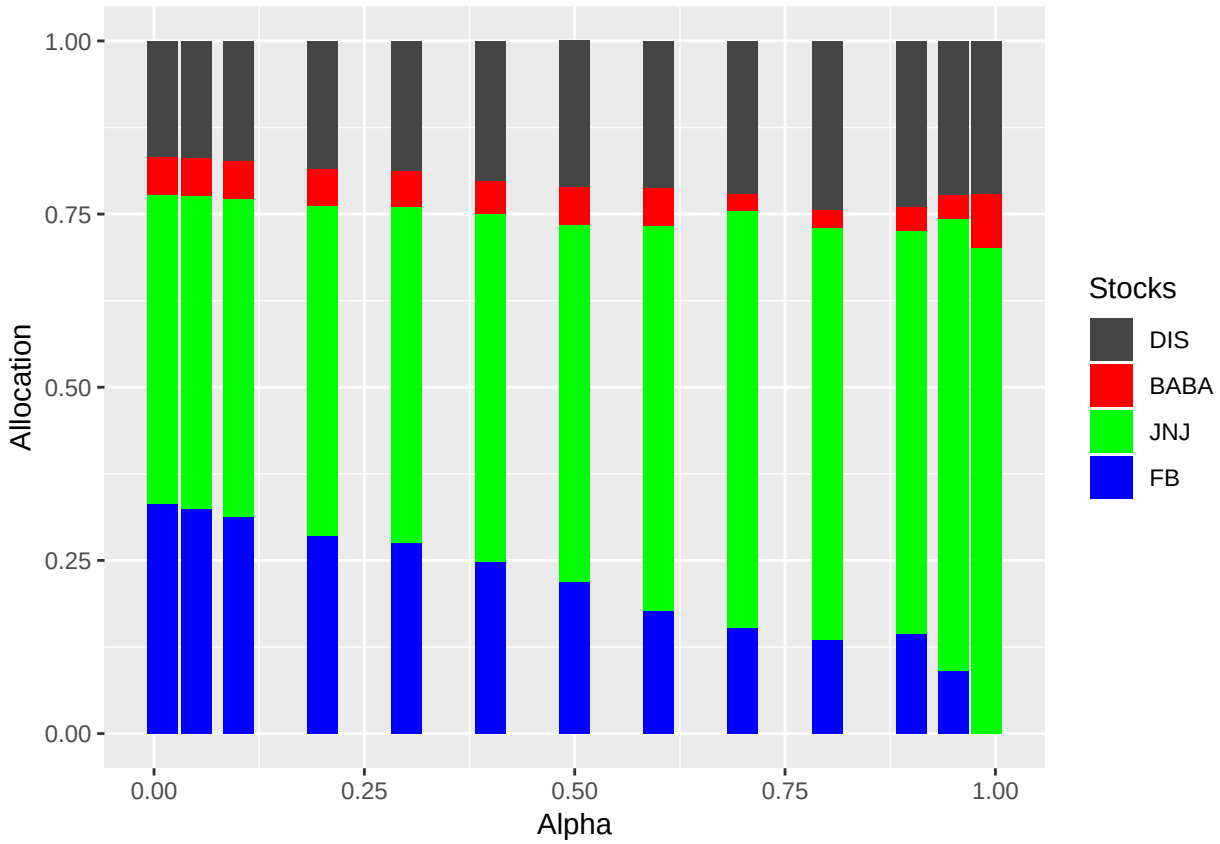
Asset allocation for $\gamma = 0.3$

```
result3 <-  data.frame(alpha,dis_result3,baba_result3,jnj_result3,fb_result3)
write.csv(result3, file = "Results3.csv")
ggplot(result3, aes(x=result3$alpha)) +
```

```
  geom_area(aes(y=result3$dis_result3+result3$baba_result3+result3$jnj_result3+result3$fb_result3, fill=
  geom_area(aes(y=result3$dis_result3+result3$baba_result3+result3$jnj_result3, fill="jnj")) +
  geom_area(aes(y=result3$dis_result3+result3$baba_result3, fill="baba")) +
  geom_area(aes(y=result3$dis_result3,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations values") +
  labs(title="Area plot for gamma = 0.3") +  # title and caption
  scale_fill_manual("Stocks",values = c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"
```

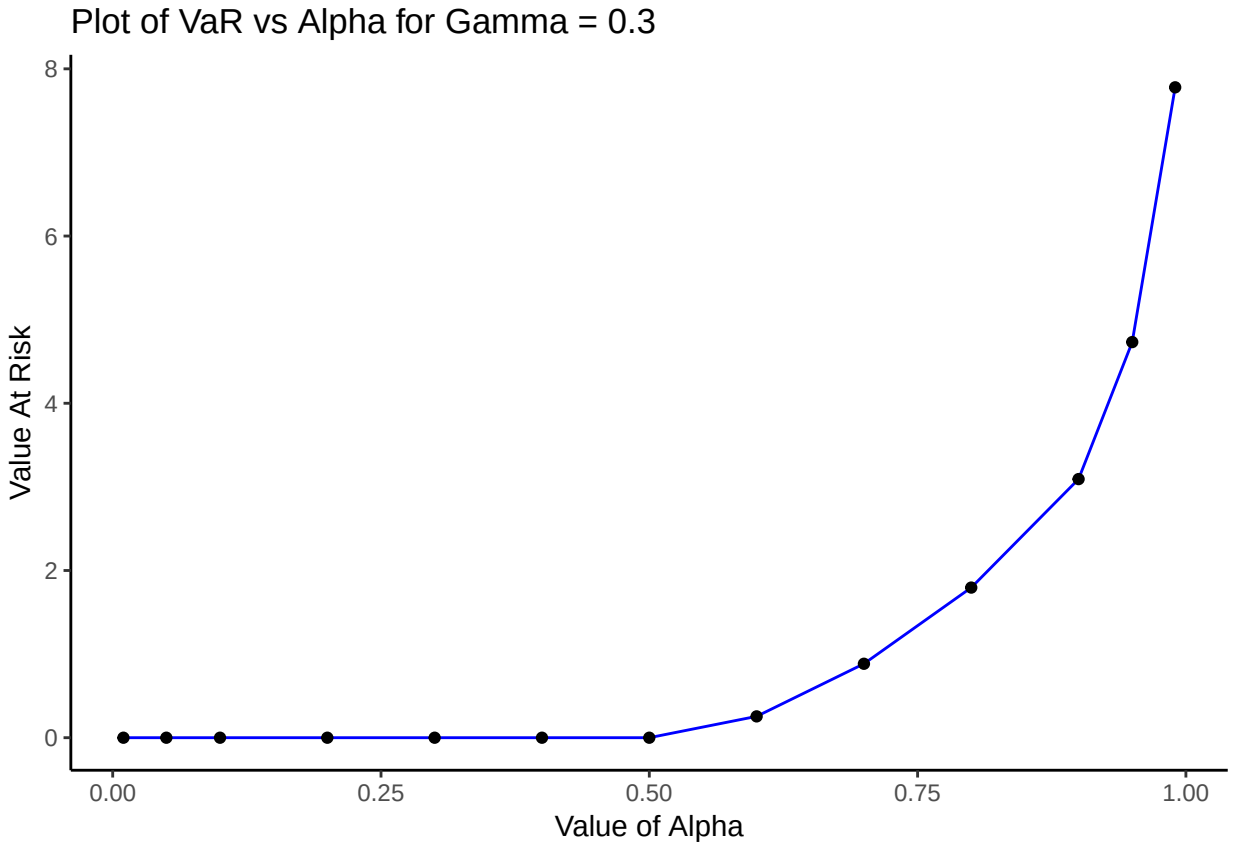



```
data3 <- result3
names(data3) <- c("alpha","DIS","BABA","JNJ","FB")
mdata3 <- melt(data3, id=c("alpha"))
names(mdata3) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata3, stat="identity")+
  scale_fill_manual("Stocks",values =
                    c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545"))
```



Plot of alpha vs q for gamma = 0.3

```
result_q3 <- data.frame(alpha,q_result3)
write.csv(result_q3,file = "VaR_Results3.csv")
ggplot(data=result_q3, aes(x=alpha, y=q_result3, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.3 ",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



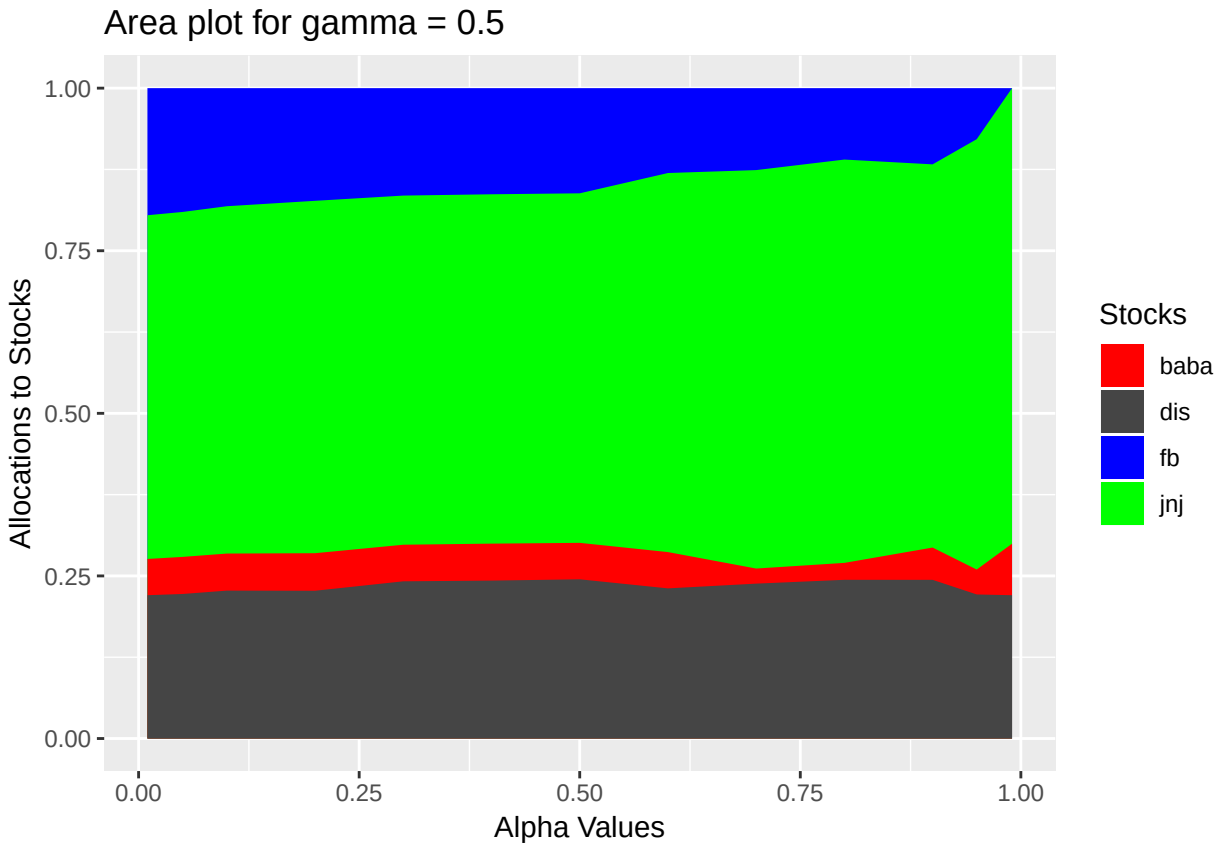
With gamma=0.5

```
gamma5 = 0.5
dis_result5 = vector(mode = "numeric")
baba_result5 = vector(mode = "numeric")
jnj_result5 = vector(mode = "numeric")
fb_result5 = vector(mode = "numeric")
q_result5 = vector(mode = "numeric")
for (alp in alpha){
  objvect5 <- c((1-gamma5)*t(prob)%*%Portreturns,(-gamma5/(1-alp))*t(prob),-gamma5)
  names(objvect5)<-c("dis","baba","jnj","fb",rep("z",826),"q")
  Solution5 <- solveLP(objvect5,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result5 <- append(dis_result5,Solution5$solution[1])
  baba_result5 <- append(baba_result5,Solution5$solution[2])
  jnj_result5 <- append(jnj_result5,Solution5$solution[3])
  fb_result5<- append(fb_result5,Solution5$solution[4])
  q_result5 <- append(q_result5,Solution5$solution[831])
}
```

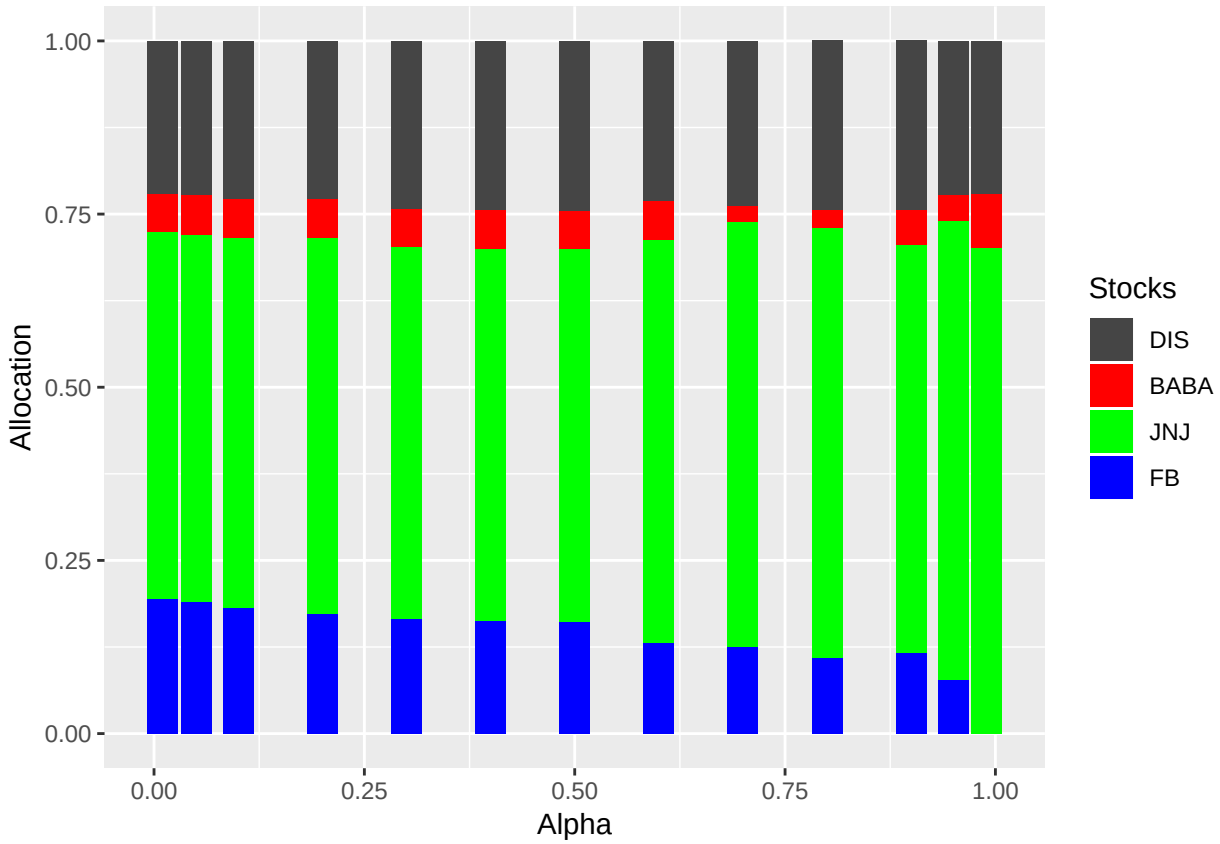
Asset allocation for $\gamma = 0.5$

```
result5 <-  data.frame(alpha,dis_result5,baba_result5,jnj_result5,fb_result5)
write.csv(result5, file = "Results5.csv")
ggplot(result5, aes(x=result5$alpha)) +
```

```
  geom_area(aes(y=result5$dis_result5+result5$baba_result5+result5$jnj_result5+result5$fb_result5, fill=
  geom_area(aes(y=result5$dis_result5+result5$baba_result5+result5$jnj_result5, fill="jnj")) +
  geom_area(aes(y=result5$dis_result5+result5$baba_result5, fill="baba")) +
  geom_area(aes(y=result5$dis_result5,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations to Stocks") +
  labs(title="Area plot for gamma = 0.5") +  # title and caption
  scale_fill_manual("Stocks",values = c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"
```

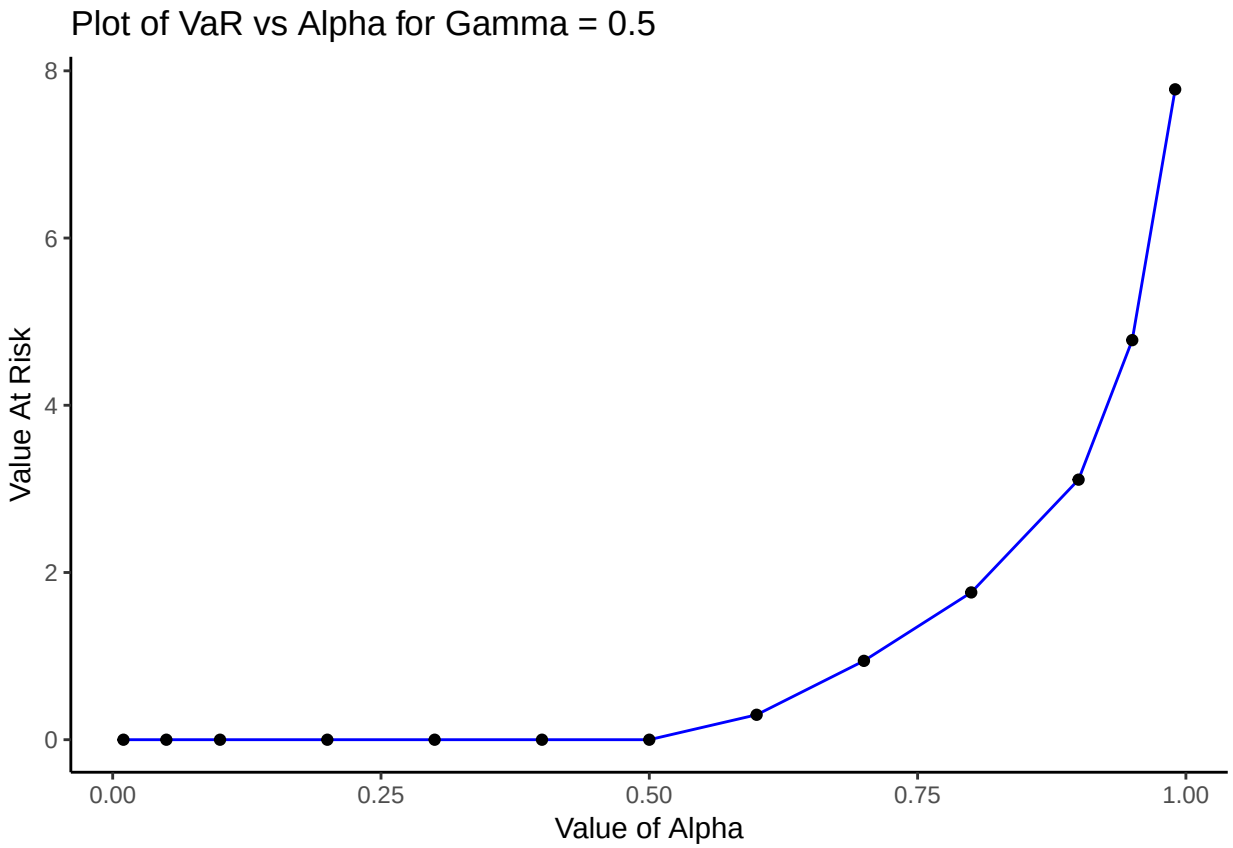



```
data5 <- result5
names(data5) <- c("alpha","DIS","BABA","JNJ","FB")
mdata5 <- melt(data5, id=c("alpha"))
names(mdata5) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata5, stat="identity")+
  scale_fill_manual("Stocks",values =
                    c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545"))
```



Plot of alpha vs q for gamma = 0.5

```
result_q5 <- data.frame(alpha,q_result5)
write.csv(result_q5,file = "VaR_Results5.csv")
ggplot(data=result_q5, aes(x=alpha, y=q_result5, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.5 ",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



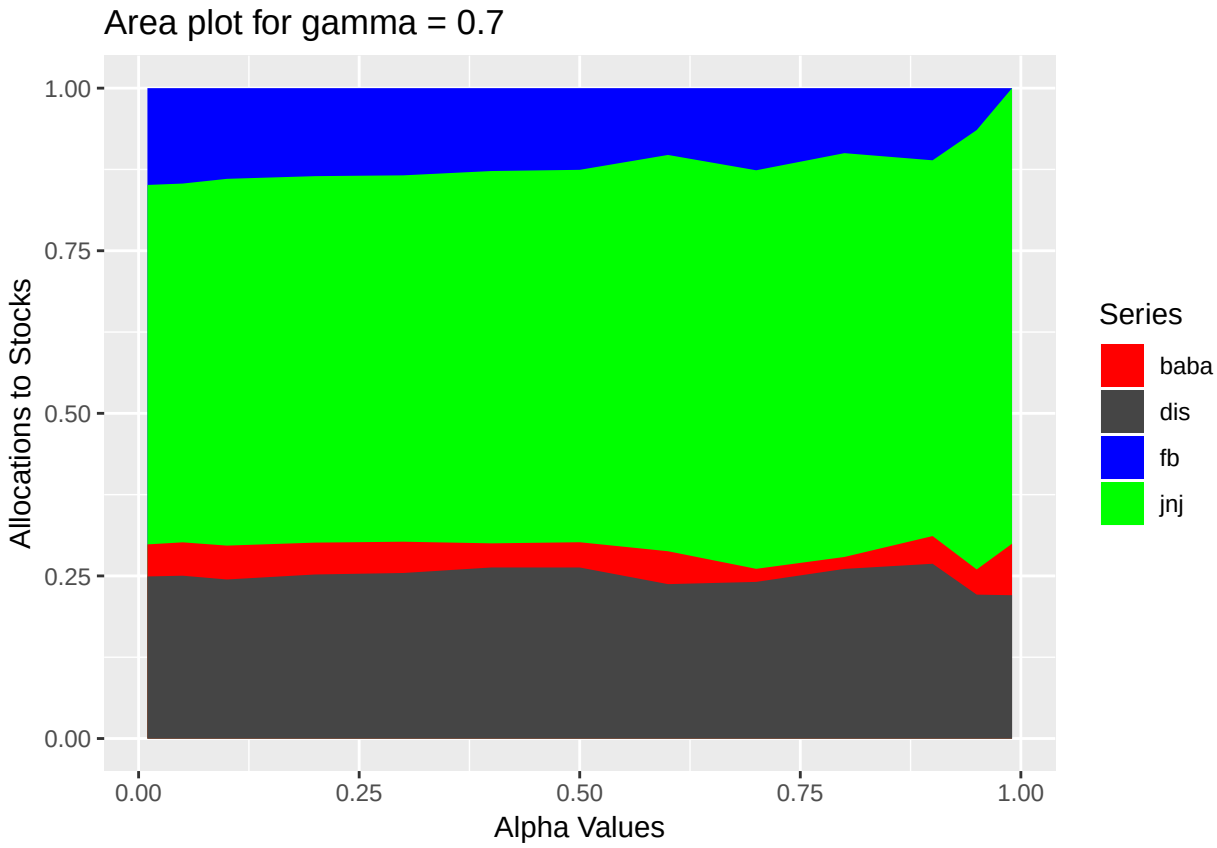
With gamma=0.7

```
gamma7 = 0.7
dis_result7 = vector(mode = "numeric")
baba_result7 = vector(mode = "numeric")
jnj_result7 = vector(mode = "numeric")
fb_result7 = vector(mode = "numeric")
q_result7 = vector(mode = "numeric")
for (alp in alpha){
  objvect7 <- c((1-gamma7)*t(prob)%*%Portreturns,(-gamma7/(1-alp))*t(prob),-gamma7)
  names(objvect7)<-c("dis","baba","jnj","fb",rep("z",826),"q")
  Solution7 <- solveLP(objvect7,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result7<-append(dis_result7,Solution7$solution[1])
  baba_result7<-append(baba_result7,Solution7$solution[2])
  jnj_result7<-append(jnj_result7,Solution7$solution[3])
  fb_result7<-append(fb_result7,Solution7$solution[4])
  q_result7 <- append(q_result7,Solution7$solution[831])
}
```

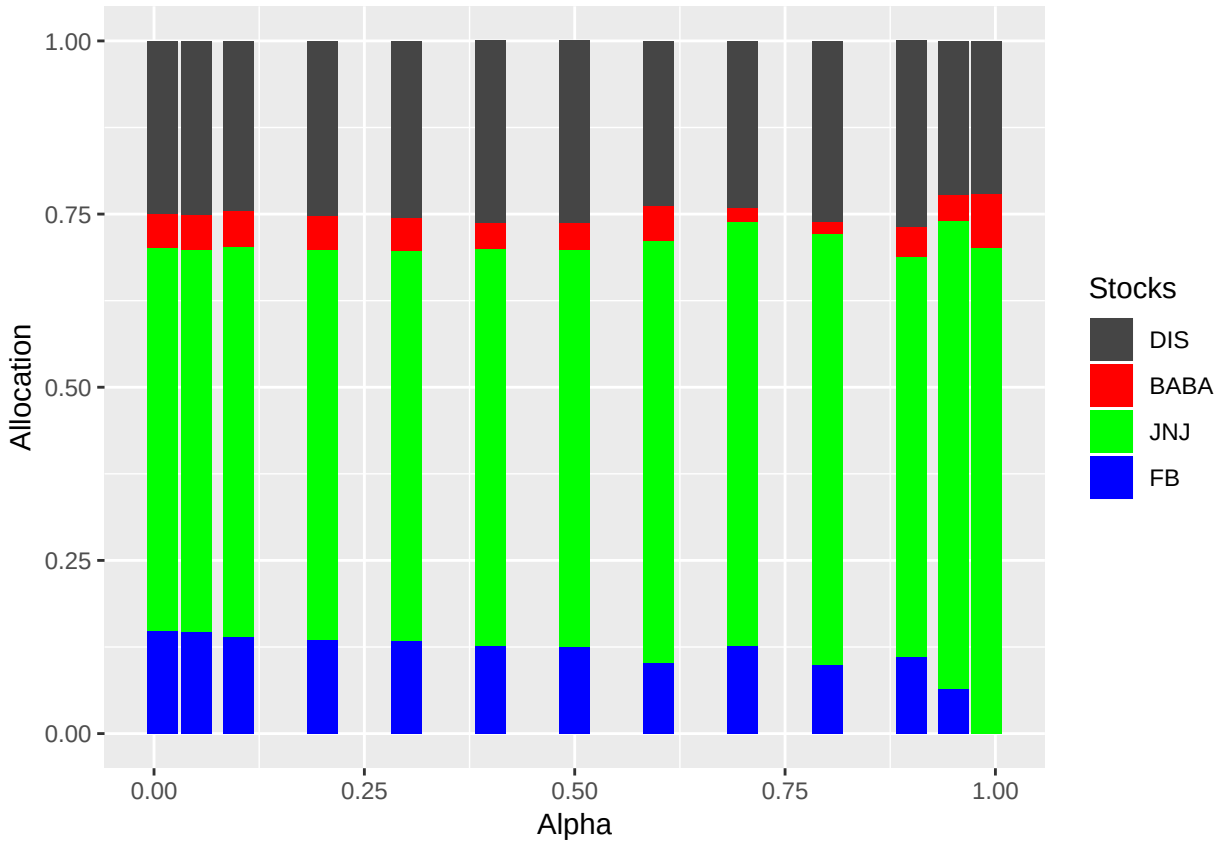
Asset allocation for $\gamma = 0.7$

```
result7 <-  data.frame(alpha,dis_result7,baba_result7,jnj_result7,fb_result7)
write.csv(result7, file = "Results7.csv")
ggplot(result7, aes(x=result7$alpha)) +
```

```
  geom_area(aes(y=result7$dis_result7+result7$baba_result7+result7$jnj_result7+result7$fb_result7, fill=
  geom_area(aes(y=result7$dis_result7+result7$baba_result7+result7$jnj_result7, fill="jnj")) +
  geom_area(aes(y=result7$dis_result7+result7$baba_result7, fill="baba")) +
  geom_area(aes(y=result7$dis_result7,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations to Stocks") +
  labs(title="Area plot for gamma = 0.7") +
  scale_fill_manual("Series",values = c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"
```

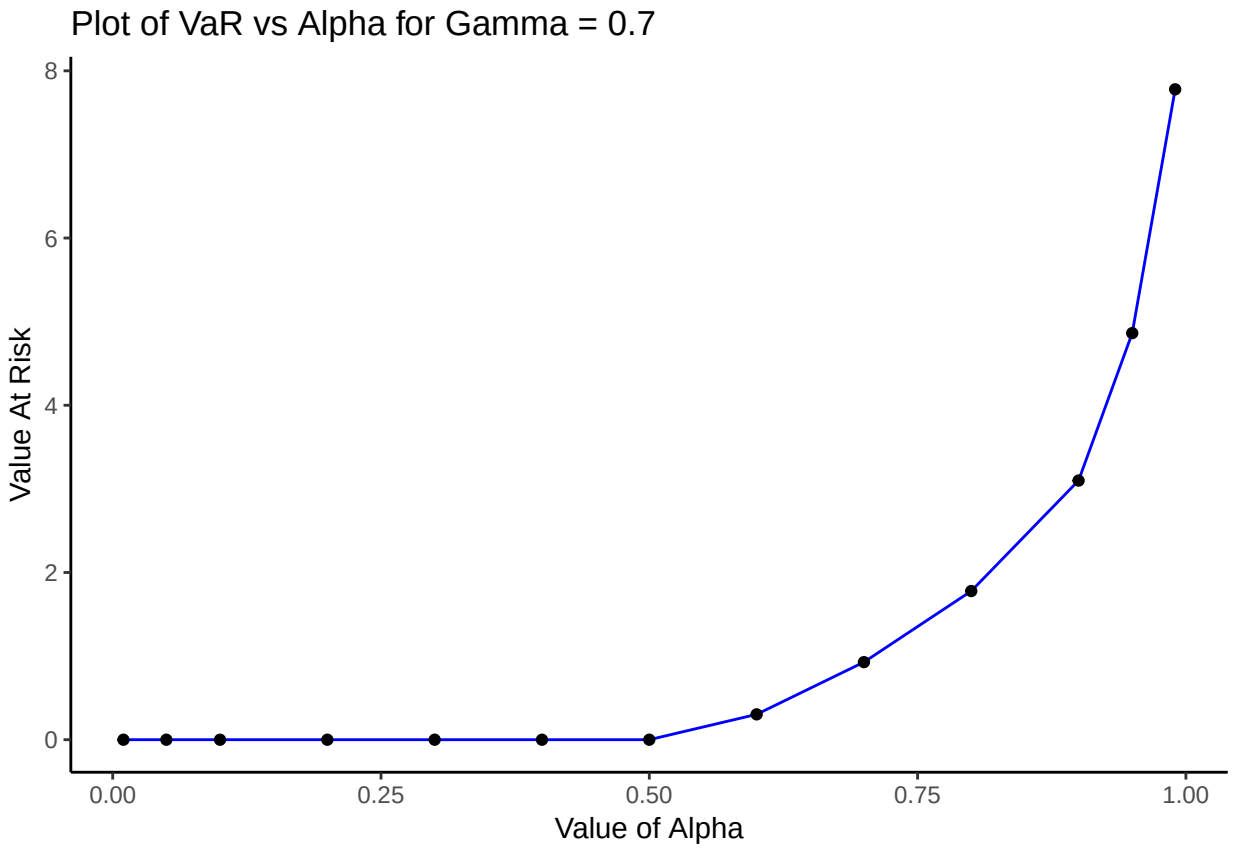



```
data7 <- result7
names(data7) <- c("alpha","DIS","BABA","JNJ","FB")
mdata7 <- melt(data7, id=c("alpha"))
names(mdata7) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata7, stat="identity")+
  scale_fill_manual("Stocks",values =
                      c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545"))
```



Plot of alpha vs q for gamma = 0.7

```
result_q7 <- data.frame(alpha,q_result7)
write.csv(result_q7,file = "VaR_Results7.csv")
ggplot(data=result_q7, aes(x=alpha, y=q_result7, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.7 ",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



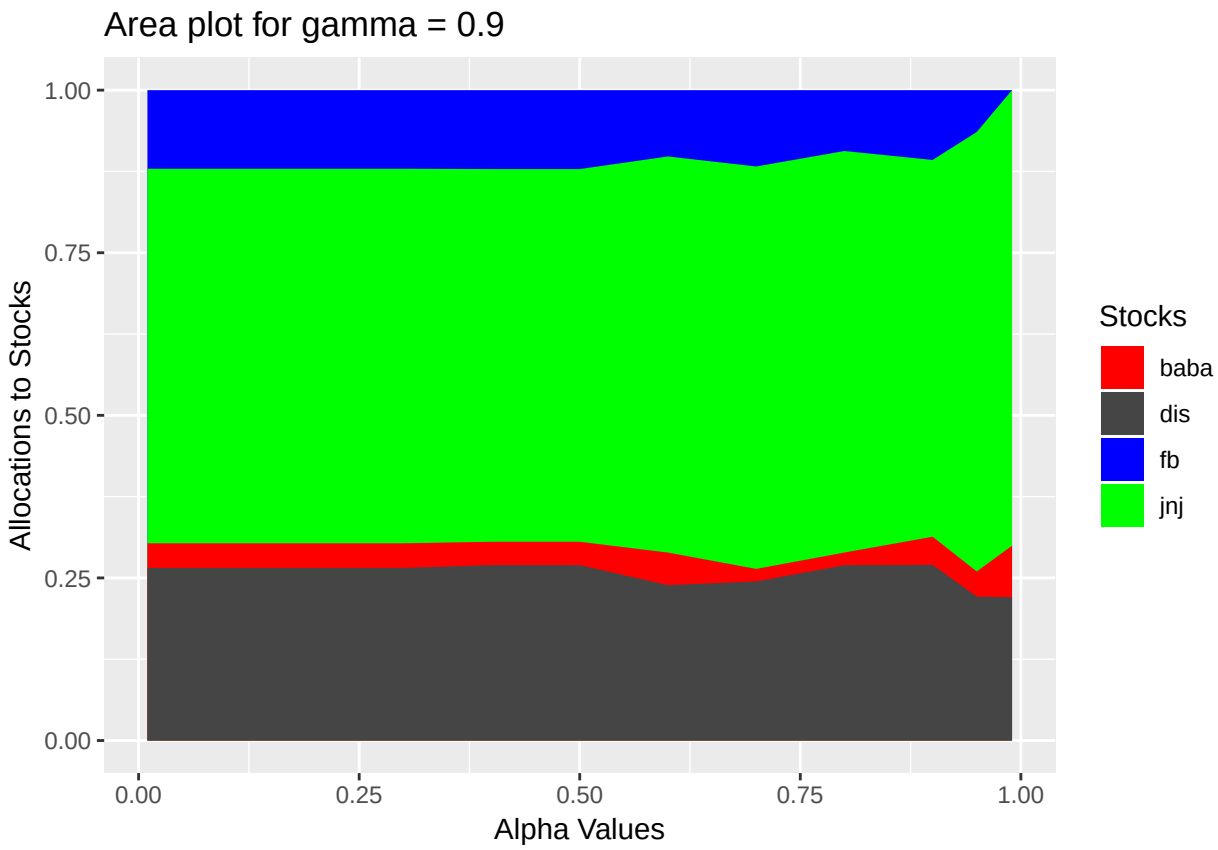
With gamma=0.9

```
gamma9 = 0.9
dis_result9 = vector(mode = "numeric")
baba_result9 = vector(mode = "numeric")
jnj_result9 = vector(mode = "numeric")
fb_result9 = vector(mode = "numeric")
q_result9 = vector(mode = "numeric")
for (alp in alpha){
  objvect9 <- c((1-gamma9)*t(prob)%*%Portreturns,(-gamma9/(1-alp))*t(prob),-gamma9)
  names(objvect9)<-c("dis","baba","jnj","fb",rep("z",826),"q")
  Solution9 <- solveLP(objvect9,rhscons,Amat,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",830)),lpSolve = TRUE)
  dis_result9<-append(dis_result9,Solution9$solution[1])
  baba_result9<-append(baba_result9,Solution9$solution[2])
  jnj_result9<-append(jnj_result9,Solution9$solution[3])
  fb_result9<-append(fb_result9,Solution9$solution[4])
  q_result9 <- append(q_result9,Solution9$solution[831])
}
```

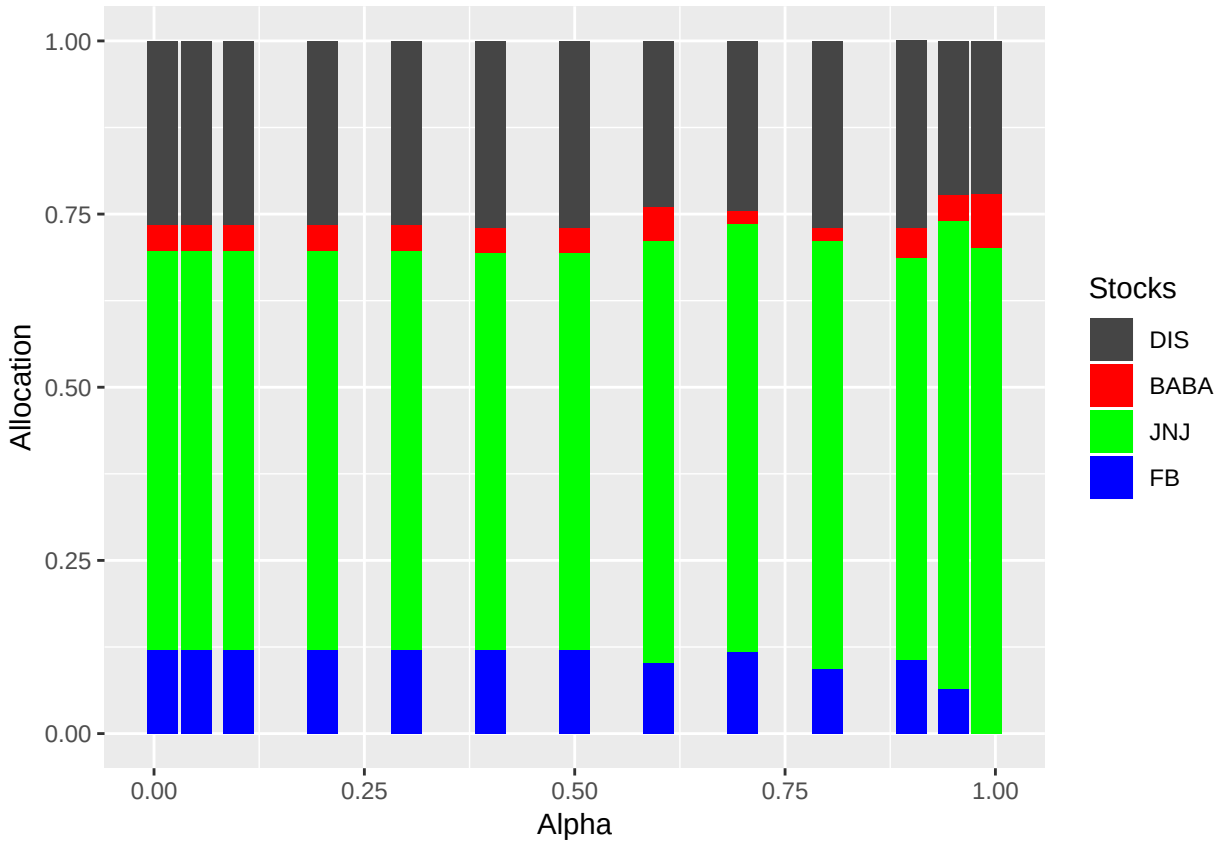
Asset allocation for $\gamma = 0.9$

```
result9 <-  data.frame(alpha,dis_result9,baba_result9,jnj_result9,fb_result9)
write.csv(result9, file = "Results9.csv")
ggplot(result9, aes(x=result9$alpha)) +
```

```
  geom_area(aes(y=result9$dis_result9+result9$baba_result9+result9$jnj_result9+result9$fb_result9, fill=
  geom_area(aes(y=result9$dis_result9+result9$baba_result9+result9$jnj_result9, fill="jnj")) +
  geom_area(aes(y=result9$dis_result9+result9$baba_result9, fill="baba")) +
  geom_area(aes(y=result9$dis_result9,fill = 'dis')) +
  xlab("Alpha Values") + ylab("Allocations to Stocks") +
  labs(title="Area plot for gamma = 0.9")+
  scale_fill_manual("Stocks",values = c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545"
```

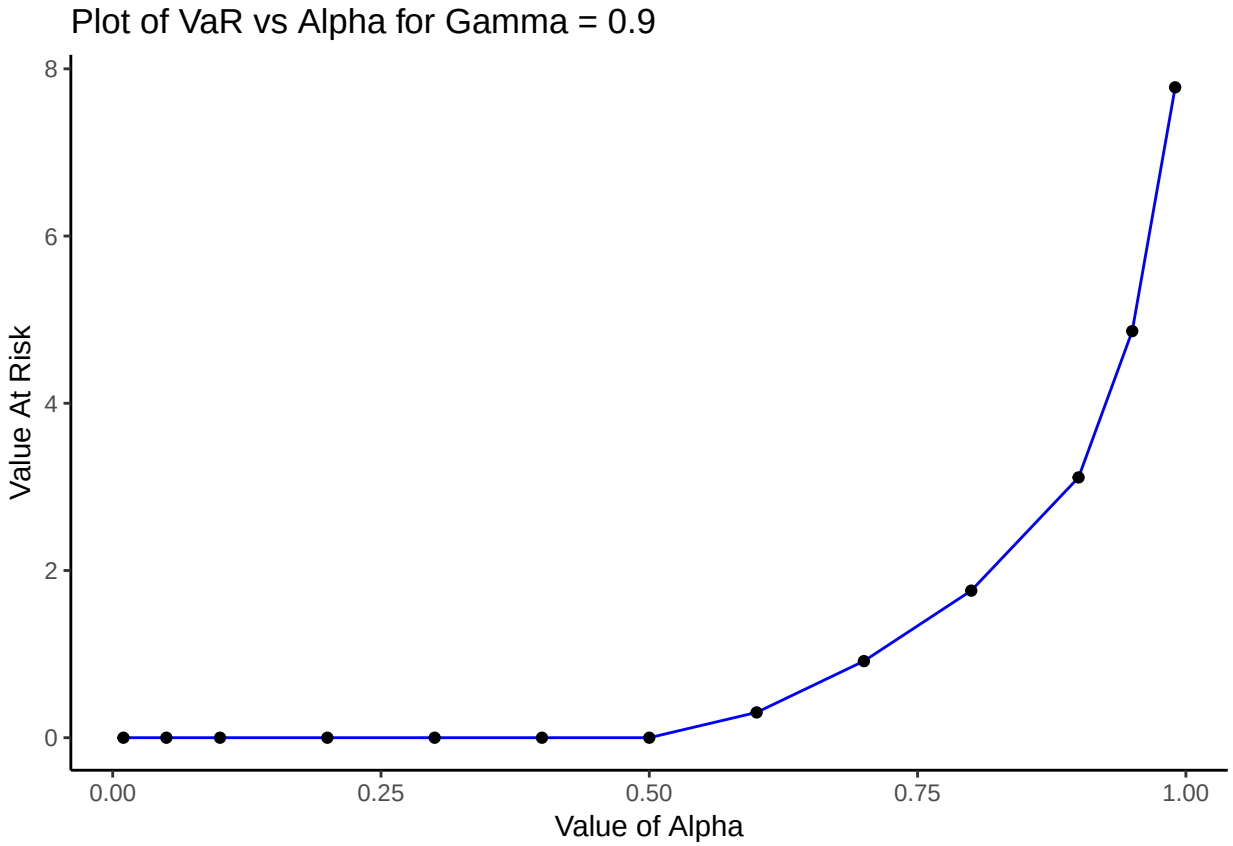



```
data9 <- result9
names(data9) <- c("alpha","DIS","BABA","JNJ","FB")
mdata9 <- melt(data9, id=c("alpha"))
names(mdata9) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata9, stat="identity")+
  scale_fill_manual("Stocks",values =
                    c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545"))
```



Plot of alpha vs q(value at risk) for gamma = 0.9

```
result_q9 <- data.frame(alpha,q_result9)
write.csv(result_q9,file = "VaR_Results9.csv")
ggplot(data=result_q9, aes(x=alpha, y=q_result9, group=1)) +
  geom_line(color="blue")+
  geom_point()+
  labs(title="Plot of VaR vs Alpha for Gamma = 0.9",x="Value of Alpha", y = "Value At Risk")+
  theme_classic()
```



OPTIMIZATION WITH A RISK-LESS ASSET ADDED

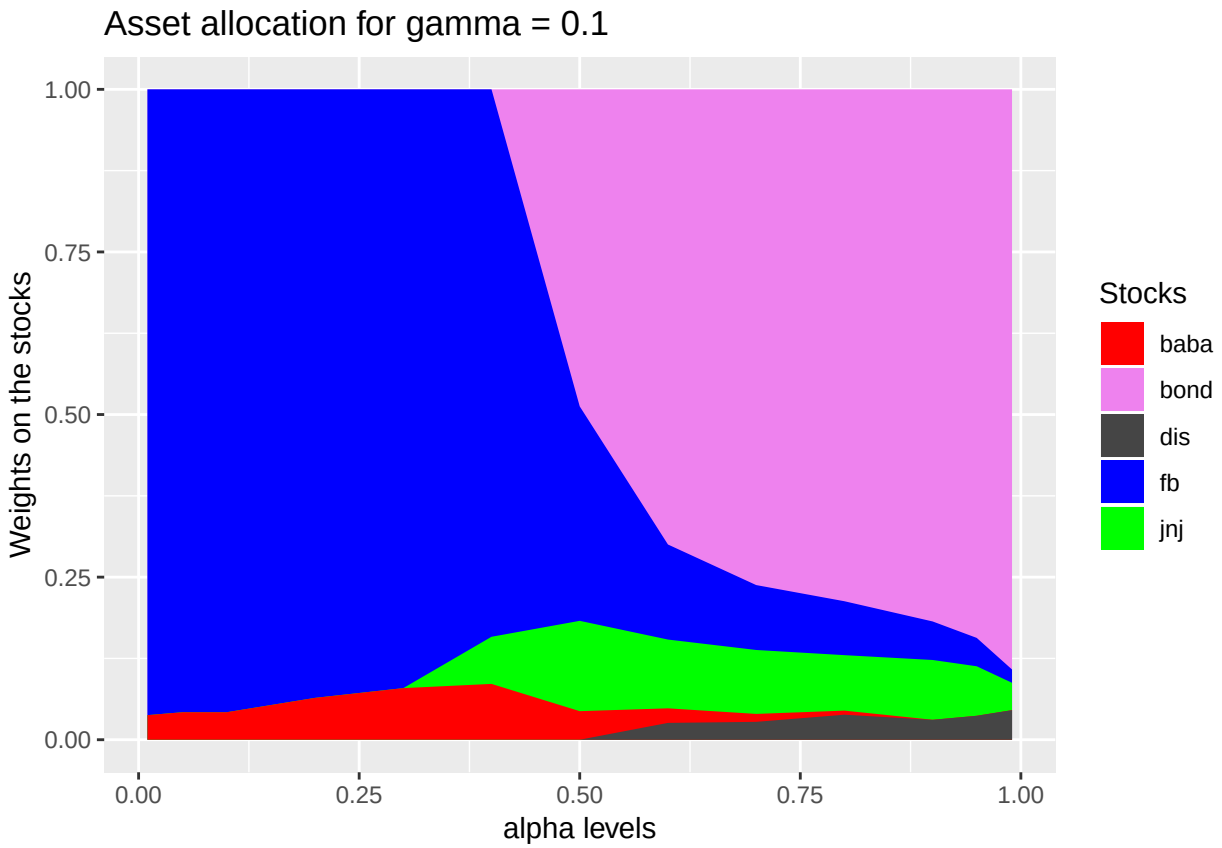
Having added a risk less asset(BOND), we will have a total of 5 assets in our portfolio. The main problem is to show what ahhpens when a risk less asset is included in the portfolio. The main result for this model is that, for higher levels of γ, the risk less asset becomes more and more important i.e, the risk less asset gets more and more weights as the level of γ increases.

Solving and asset allocation for $\gamma = 0.1$

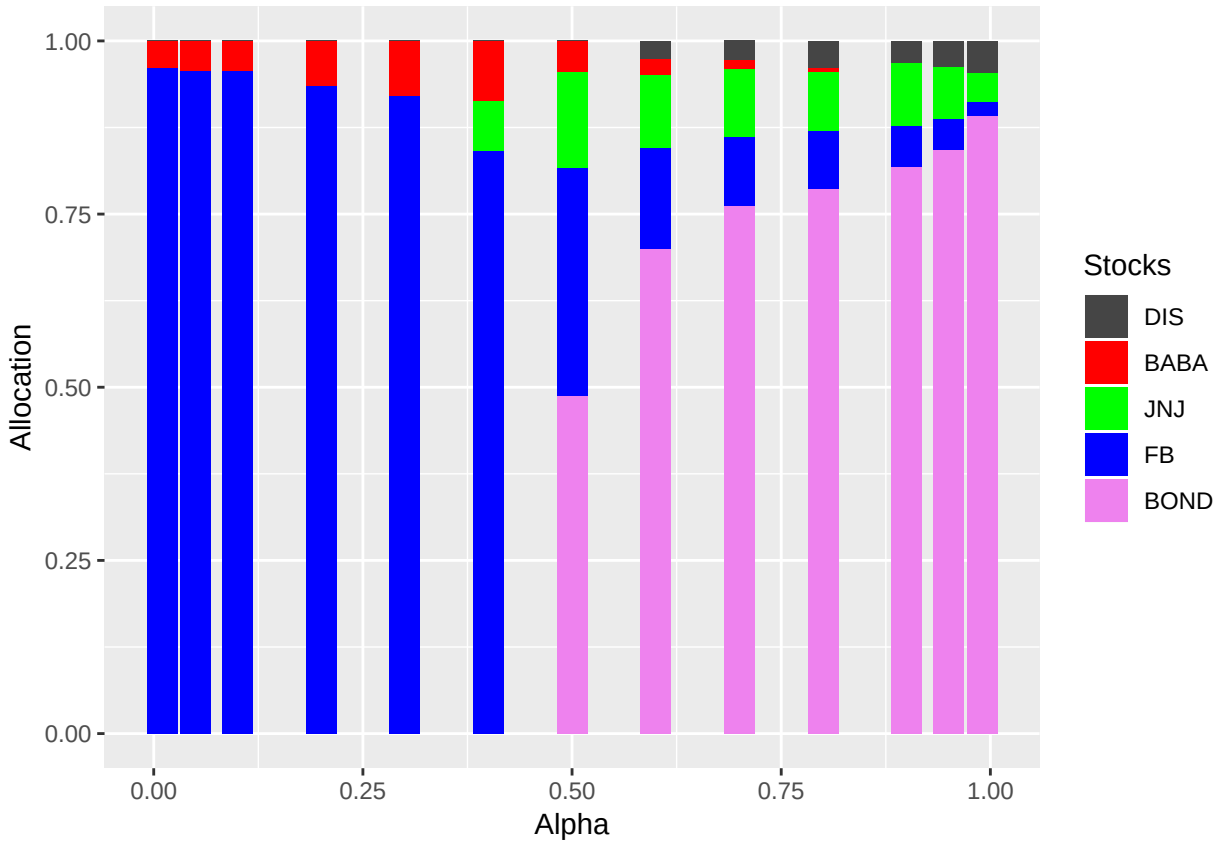
The same way as in the solution of the model with only risky assets, the **Amatfree** matrix doesn't change and therefore once created, we will use the same matrix for all levels of γ that we are considering.

```
gammafree1 = 0.1
dis_resultfree1 = vector(mode = "numeric")
baba_resultfree1 = vector(mode = "numeric")
jnj_resultfree1 = vector(mode = "numeric")
fb_resultfree1 = vector(mode = "numeric")
bond_resultfree1 = vector(mode = "numeric")
for (alp in alpha){
  objvectfree1 <- c((1-gammafree1)*t(prob)%*%Portreturnsfree,(-gammafree1/(1-alp))*t(prob),-gammafree1)
  names(objvectfree1)<-c("dis","baba","jnj","fb","bond",rep("z",826),"q")
  rhsconsfree1 <- c(rep(0,826 ),1,rep(0,831))
#Construction of matrix Amat
  firstconsfree = cbind(-Portreturnsfree,-diag(826),-rep(1,826))
```

```
  secondconsfree = (c(rep(1,5),rep(0,827)))
  lessxfree = cbind(-diag(5),zeros(5,827))
  lesszfree = cbind(zeros(826,5),-diag(826),rep(0,826))
  Amatfree = rbind(firstconsfree,secondconsfree,lessxfree,lesszfree)
  dim(Amatfree)
  colnames(Amatfree) = NULL
  rownames(Amatfree) = NULL
#solution of the problem using linprog
  Solutionfree1 <- solveLP(objvectfree1,rhsconsfree1,Amatfree,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",831)),lpSolve = TRUE)
  dis_resultfree1<-append(dis_resultfree1,Solutionfree1$solution[1])
  baba_resultfree1<- append(baba_resultfree1,Solutionfree1$solution[2])
  jnj_resultfree1<-append(jnj_resultfree1,Solutionfree1$solution[3])
  fb_resultfree1<-append(fb_resultfree1,Solutionfree1$solution[4])
  bond_resultfree1<-append(bond_resultfree1,Solutionfree1$solution[5])
}
  resultfree1 <-data.frame(alpha,dis_resultfree1,baba_resultfree1,jnj_resultfree1,
                           fb_resultfree1,bond_resultfree1)
  write.csv(resultfree1, file = "Resultsfree01.csv")
  ggplot(resultfree1, aes(x=resultfree1$alpha)) +
    geom_area(aes(y=resultfree1$dis_resultfree1+resultfree1$baba_resultfree1+
                    resultfree1$jnj_resultfree1+resultfree1$fb_resultfree1+
                    resultfree1$bond_resultfree1, fill="bond"))+
    geom_area(aes(y=resultfree1$dis_resultfree1+resultfree1$baba_resultfree1+
                    resultfree1$jnj_resultfree1+resultfree1$fb_resultfree1, fill="fb")) +
    geom_area(aes(y=resultfree1$dis_resultfree1+resultfree1$baba_resultfree1+
                    resultfree1$jnj_resultfree1, fill="jnj")) +
    geom_area(aes(y=resultfree1$dis_resultfree1+resultfree1$baba_resultfree1,fill = 'baba')) +
    geom_area(aes(y=resultfree1$dis_resultfree1,fill = "dis"))+
    xlab("alpha levels") + ylab("Weights on the stocks") +
    labs(title="Asset allocation for gamma = 0.1") +  # title and caption
    scale_fill_manual("Stocks",
                      values =c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545","bond"=
```



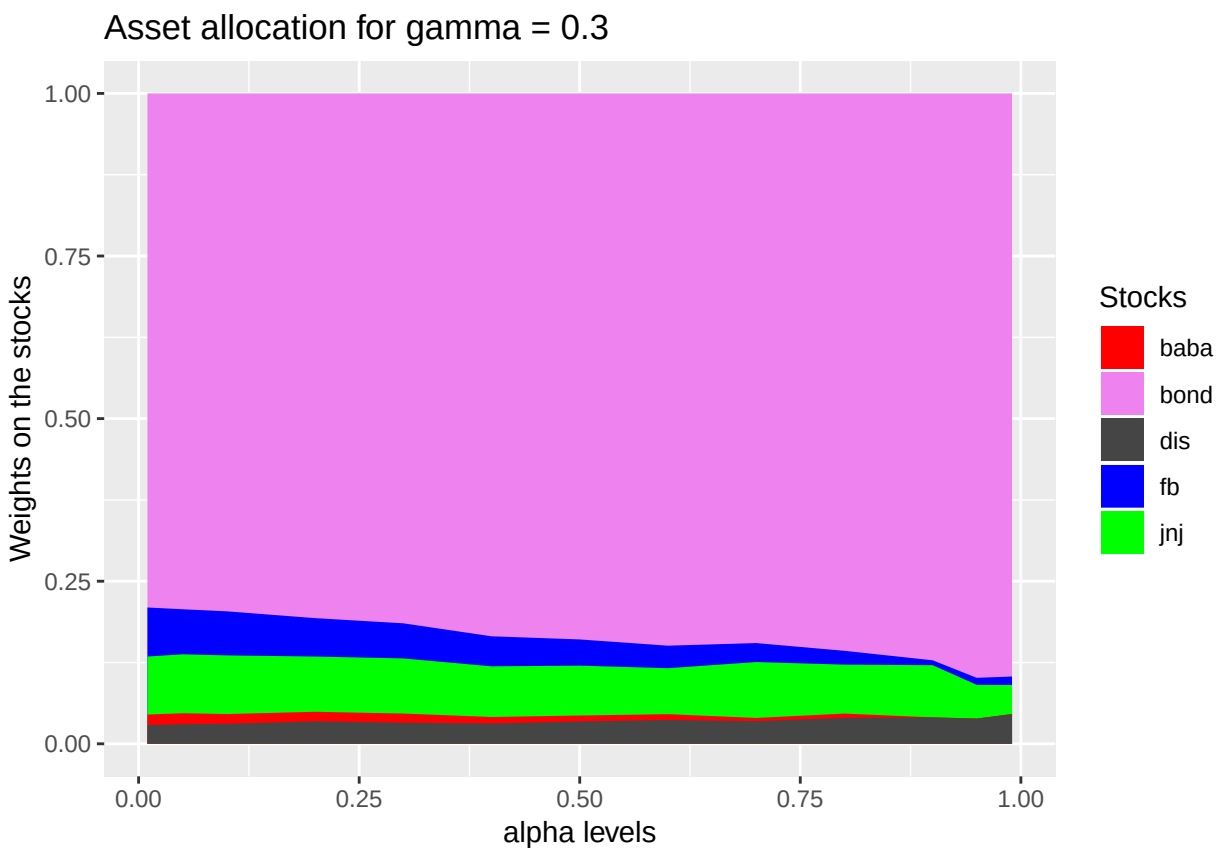

```
# Using stacked barplots
library(reshape2)
data1 <- resultfree1
names(data1) <- c("alpha","DIS","BABA","JNJ","FB","BOND")
mdata1 <- melt(data1, id=c("alpha"))
names(mdata1) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata1, stat="identity")+
  scale_fill_manual("Stocks",values =
                      c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545","BOND" = "viole
```



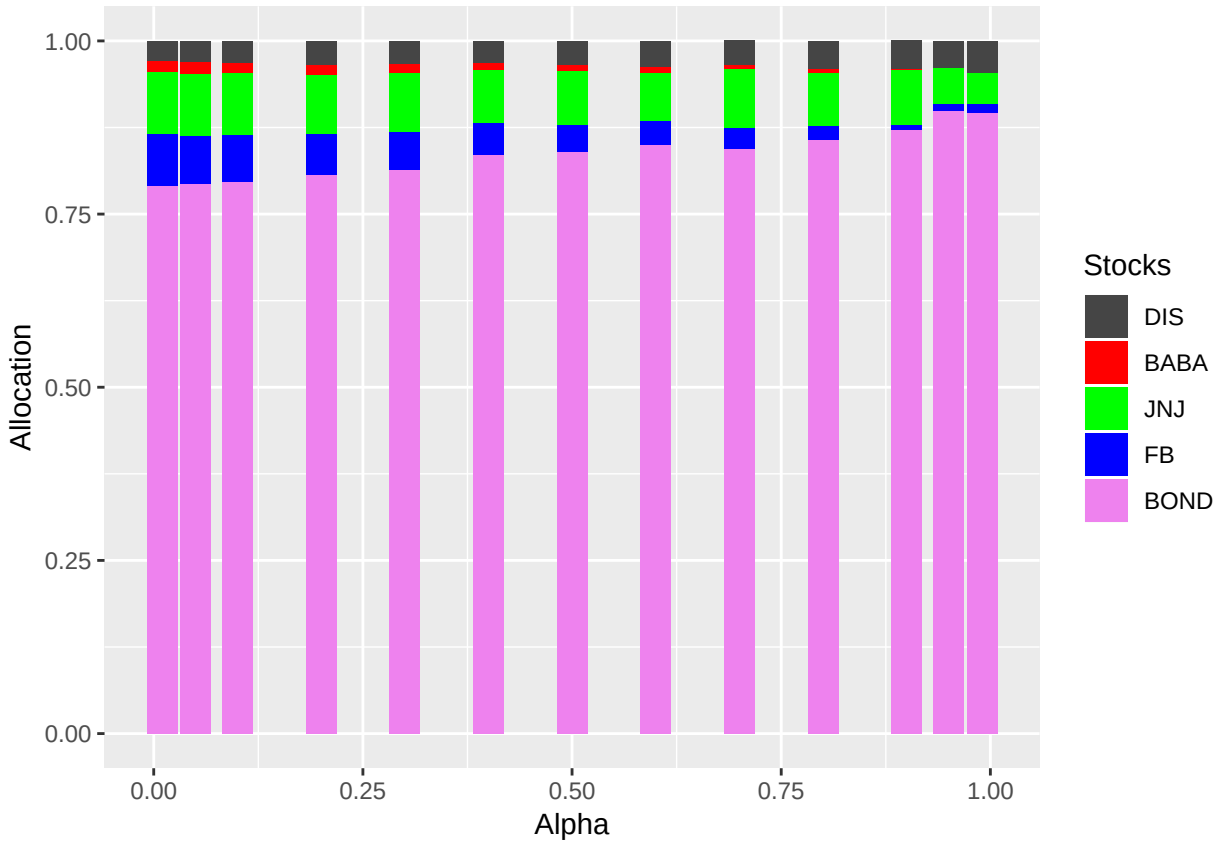
Solving and asset allocation for γfree $= 0.3$

```
gammafree3 = 0.3
dis_resultfree3 = vector(mode = "numeric")
baba_resultfree3 = vector(mode = "numeric")
jnj_resultfree3 = vector(mode = "numeric")
fb_resultfree3 = vector(mode = "numeric")
bond_resultfree3 = vector(mode = "numeric")
for (alp in alpha){
  objvectfree3 <- c((1-gammafree3)*t(prob)%*%Portreturnsfree,(-gammafree3/(1-alp))*t(prob),-gammafree3)
  names(objvectfree3)<-c("dis","baba","jnj","fb","bond",rep("z",826),"q")
  rhsconsfree3 <- c(rep(0,826 ),1,rep(0,831))
#solution of the problem using linprog
  Solutionfree3 <- solveLP(objvectfree3,rhsconsfree3,Amatfree,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",831)),lpSolve = TRUE)
  dis_resultfree3<-append(dis_resultfree3,Solutionfree3$solution[1])
  baba_resultfree3<- append(baba_resultfree3,Solutionfree3$solution[2])
  jnj_resultfree3<-append(jnj_resultfree3,Solutionfree3$solution[3])
  fb_resultfree3<-append(fb_resultfree3,Solutionfree3$solution[4])
  bond_resultfree3<-append(bond_resultfree3,Solutionfree3$solution[5])
}
  resultfree3 <-data.frame(alpha,dis_resultfree3,baba_resultfree3,
                           jnj_resultfree3,fb_resultfree3,bond_resultfree3)
  write.csv(resultfree3, file = "Resultsfree3.csv")
  ggplot(resultfree3, aes(x=resultfree3$alpha)) +
```

```
    geom_area(aes(y=resultfree3$dis_resultfree3+resultfree3$baba_resultfree3+
                    resultfree3$jnj_resultfree3+resultfree3$fb_resultfree3+
                    resultfree3$bond_resultfree3, fill="bond"))+
    geom_area(aes(y=resultfree3$dis_resultfree3+resultfree3$baba_resultfree3+
                    resultfree3$jnj_resultfree3+resultfree3$fb_resultfree3, fill="fb")) +
    geom_area(aes(y=resultfree3$dis_resultfree3+resultfree3$baba_resultfree3+
                    resultfree3$jnj_resultfree3, fill="jnj")) +
    geom_area(aes(y=resultfree3$dis_resultfree3+resultfree3$baba_resultfree3,fill = 'baba')) +
    geom_area(aes(y=resultfree3$dis_resultfree3,fill = "dis"))+
    xlab("alpha levels") + ylab("Weights on the stocks") +
    labs(title="Asset allocation for gamma = 0.3") +  # title and caption
    scale_fill_manual("Stocks",
                      values =c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545","bond"=
```



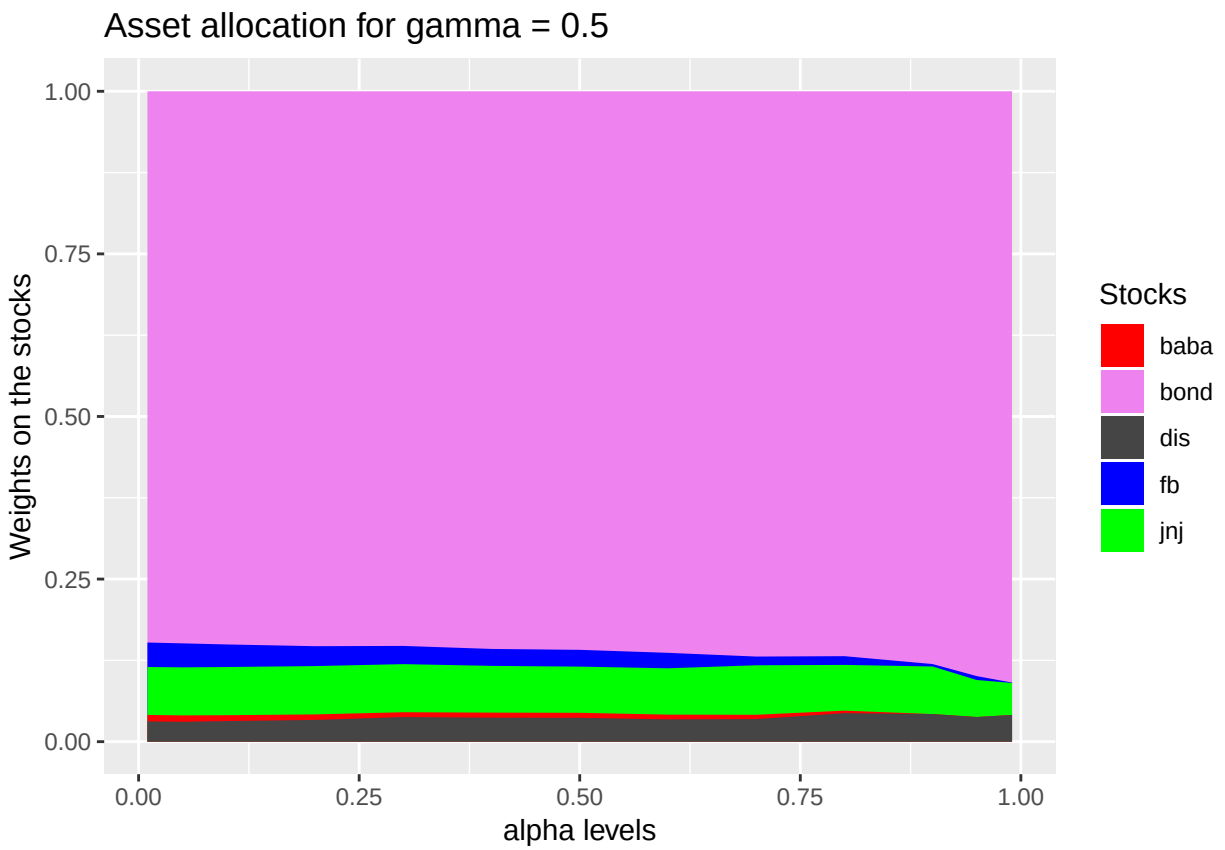

```
# Using stacked barplots
library(reshape2)
data3 <- resultfree3
names(data3) <- c("alpha","DIS","BABA","JNJ","FB","BOND")
mdata3 <- melt(data3, id=c("alpha"))
names(mdata3) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata3, stat="identity")+
  scale_fill_manual("Stocks",values =
                      c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545","BOND" = "viole
```



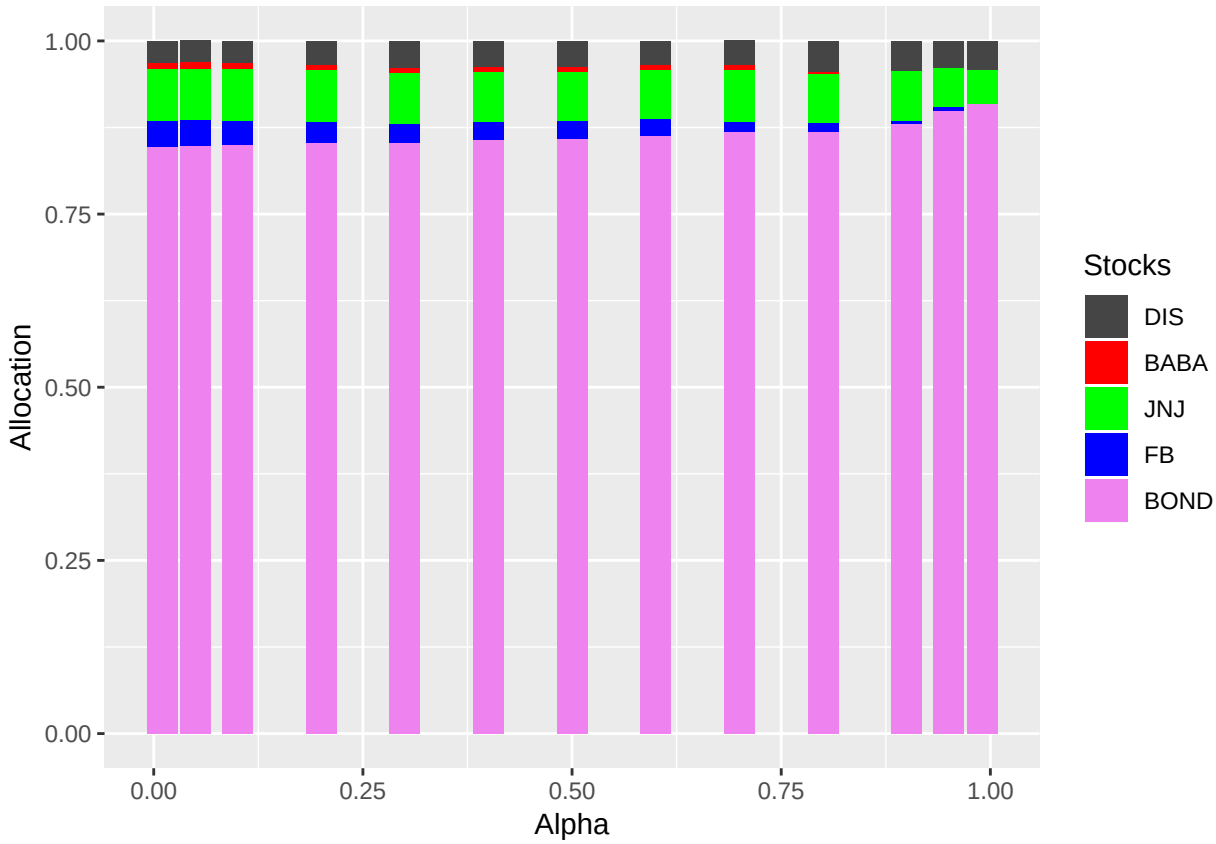
Solving and asset allocation for γfree $= 0.5$

```
gammafree5 = 0.5
dis_resultfree5 <- vector(mode = "numeric")
baba_resultfree5 <- vector(mode = "numeric")
jnj_resultfree5 <- vector(mode = "numeric")
fb_resultfree5 <- vector(mode = "numeric")
bond_resultfree5 <- vector(mode = "numeric")
for (alp in alpha){
  objvectfree5 <- c((1-gammafree5)*t(prob)%*%Portreturnsfree,(-gammafree5/(1-alp))*t(prob),-gammafree5)
  names(objvectfree5)<-c("dis","baba","jnj","fb","bond",rep("z",826),"q")
  rhsconsfree5 <- c(rep(0,826 ),1,rep(0,831))
  Solutionfree5 <- solveLP(objvectfree5,rhsconsfree5,Amatfree,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",831)),lpSolve = TRUE)
  dis_resultfree5<-append(dis_resultfree5,Solutionfree5$solution[1])
  baba_resultfree5<- append(baba_resultfree5,Solutionfree5$solution[2])
  jnj_resultfree5<-append(jnj_resultfree5,Solutionfree5$solution[3])
  fb_resultfree5<-append(fb_resultfree5,Solutionfree5$solution[4])
  bond_resultfree5<-append(bond_resultfree5,Solutionfree5$solution[5])
}
  resultfree5 <-data.frame(alpha,dis_resultfree5,baba_resultfree5,
                           jnj_resultfree5,fb_resultfree5,bond_resultfree5)
  write.csv(resultfree5, file = "Resultsfree5.csv")
  ggplot(resultfree5, aes(x=resultfree5$alpha)) +
    geom_area(aes(y=resultfree5$dis_resultfree5+resultfree5$baba_resultfree5+
```

```
                  resultfree5$jnj_resultfree5+resultfree5$fb_resultfree5+
                  resultfree5$bond_resultfree5, fill="bond"))+
    geom_area(aes(y=resultfree5$dis_resultfree5+resultfree5$baba_resultfree5+
                  resultfree5$jnj_resultfree5+resultfree5$fb_resultfree5, fill="fb")) +
    geom_area(aes(y=resultfree5$dis_resultfree5+resultfree5$baba_resultfree5+
                  resultfree5$jnj_resultfree5, fill="jnj")) +
    geom_area(aes(y=resultfree5$dis_resultfree5+resultfree5$baba_resultfree5,fill = 'baba')) +
    geom_area(aes(y=resultfree5$dis_resultfree5,fill = "dis"))+
    xlab("alpha levels") + ylab("Weights on the stocks") +
    labs(title="Asset allocation for gamma = 0.5") +  # title and caption
    scale_fill_manual("Stocks",
                      values =c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545","bond"=
```



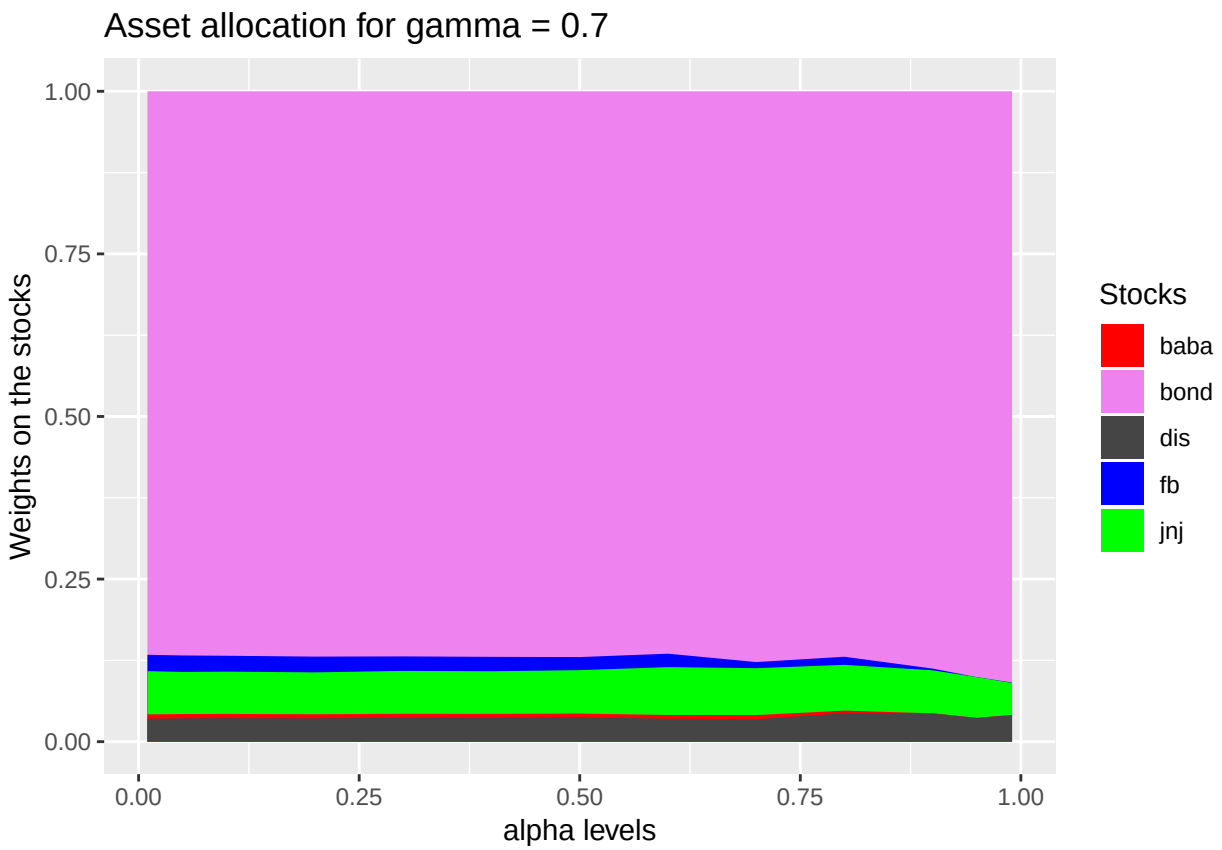

```
# Using stacked barplots
library(reshape2)
data5 <- resultfree5
names(data5) <- c("alpha","DIS","BABA","JNJ","FB","BOND")
mdata5 <- melt(data5, id=c("alpha"))
names(mdata5) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata5, stat="identity")+
  scale_fill_manual("Stocks",values =
                    c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545","BOND" = "viole
```



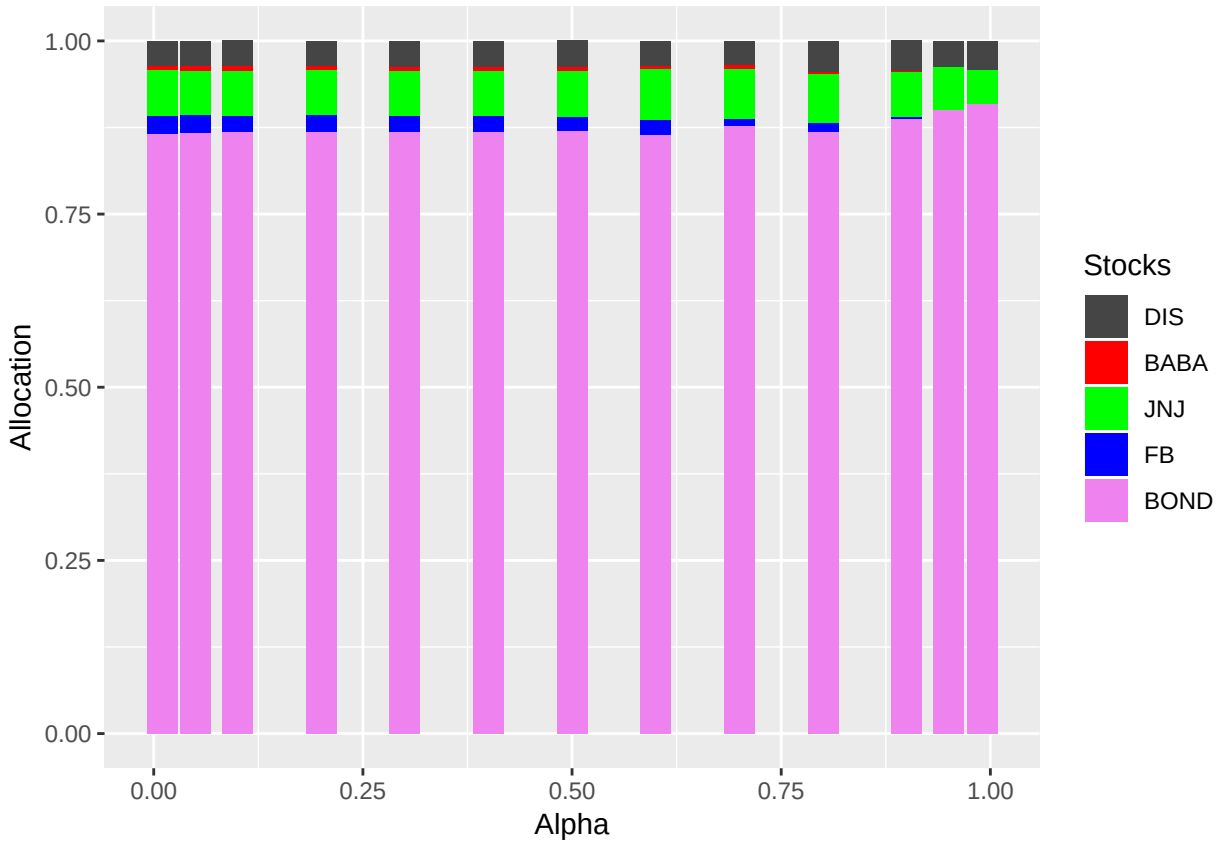
Solving and asset allocation for γfree $= 0.7$

```
gammafree7 = 0.7
dis_resultfree7 = vector(mode = "numeric")
baba_resultfree7 = vector(mode = "numeric")
jnj_resultfree7 = vector(mode = "numeric")
fb_resultfree7 = vector(mode = "numeric")
bond_resultfree7 = vector(mode = "numeric")
for (alp in alpha){
  objvectfree7 <- c((1-gammafree7)*t(prob)%*%Portreturnsfree,(-gammafree7/(1-alp))*t(prob),-gammafree7)
  names(objvectfree7)<-c("dis","baba","jnj","fb","bond",rep("z",826),"q")
  rhsconsfree7 <- c(rep(0,826 ),1,rep(0,831))
  Solutionfree7 <- solveLP(objvectfree7,rhsconsfree7,Amatfree,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",831)),lpSolve = TRUE)
  dis_resultfree7<-append(dis_resultfree7,Solutionfree7$solution[1])
  baba_resultfree7<- append(baba_resultfree7,Solutionfree7$solution[2])
  jnj_resultfree7<-append(jnj_resultfree7,Solutionfree7$solution[3])
  fb_resultfree7<-append(fb_resultfree7,Solutionfree7$solution[4])
  bond_resultfree7<-append(bond_resultfree7,Solutionfree7$solution[5])
}
  resultfree7 <-data.frame(alpha,dis_resultfree7,baba_resultfree7,
                           jnj_resultfree7,fb_resultfree7,bond_resultfree7)
  write.csv(resultfree7, file = "Resultsfree7.csv")
  ggplot(resultfree7, aes(x=resultfree7$alpha)) +
    geom_area(aes(y=resultfree7$dis_resultfree7+resultfree7$baba_resultfree7+
```

```
                  resultfree7$jnj_resultfree7+resultfree7$fb_resultfree7+
                  resultfree7$bond_resultfree7, fill="bond"))+
    geom_area(aes(y=resultfree7$dis_resultfree7+resultfree7$baba_resultfree7+
                  resultfree7$jnj_resultfree7+resultfree7$fb_resultfree7, fill="fb")) +
    geom_area(aes(y=resultfree7$dis_resultfree7+resultfree7$baba_resultfree7+
                  resultfree7$jnj_resultfree7, fill="jnj")) +
    geom_area(aes(y=resultfree7$dis_resultfree7+resultfree7$baba_resultfree7,fill = 'baba')) +
    geom_area(aes(y=resultfree7$dis_resultfree7,fill = "dis"))+
    xlab("alpha levels") + ylab("Weights on the stocks") +
    labs(title="Asset allocation for gamma = 0.7") +  # title and caption
    scale_fill_manual("Stocks",
                      values =c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545","bond"=
```



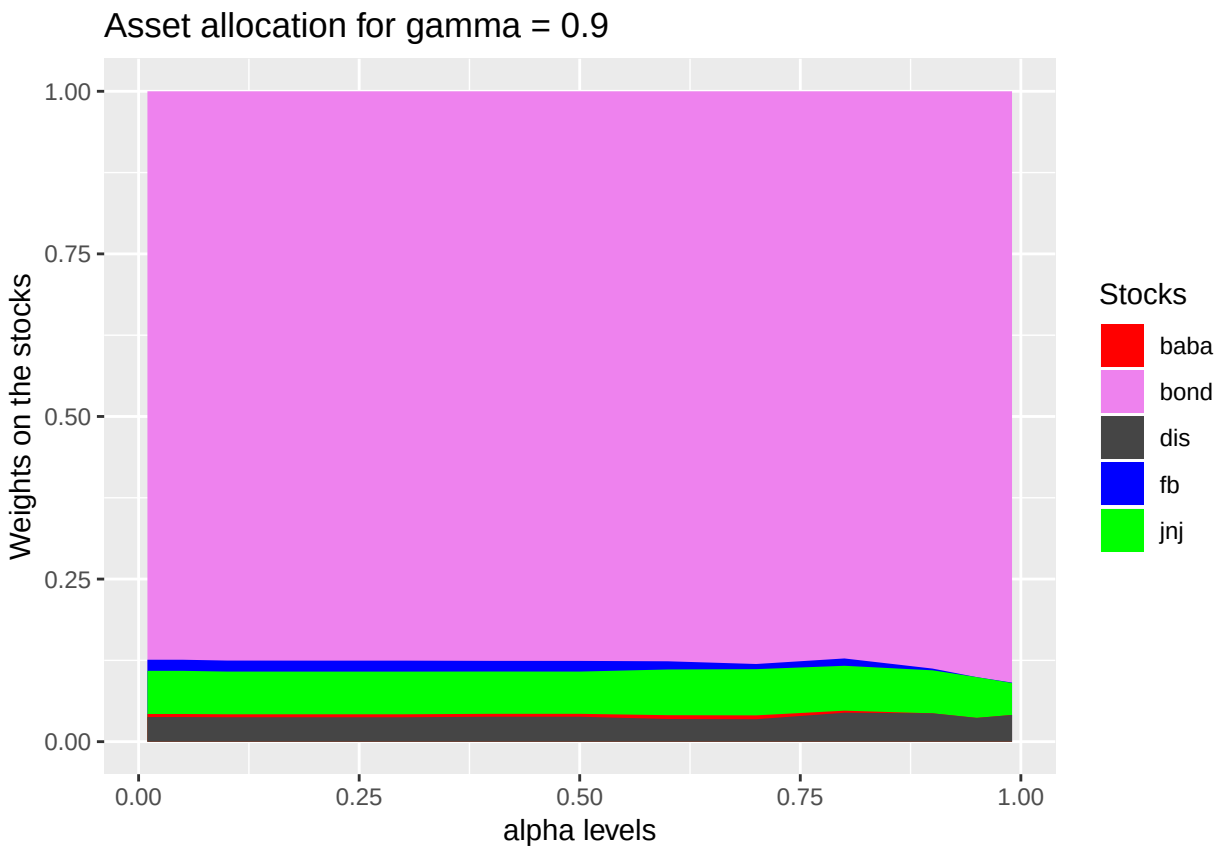

```
# Using stacked barplots
library(reshape2)
data7 <- resultfree7
names(data7) <- c("alpha","DIS","BABA","JNJ","FB","BOND")
mdata7 <- melt(data7, id=c("alpha"))
names(mdata7) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata7, stat="identity")+
  scale_fill_manual("Stocks",values =
                    c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545","BOND" = "viole
```



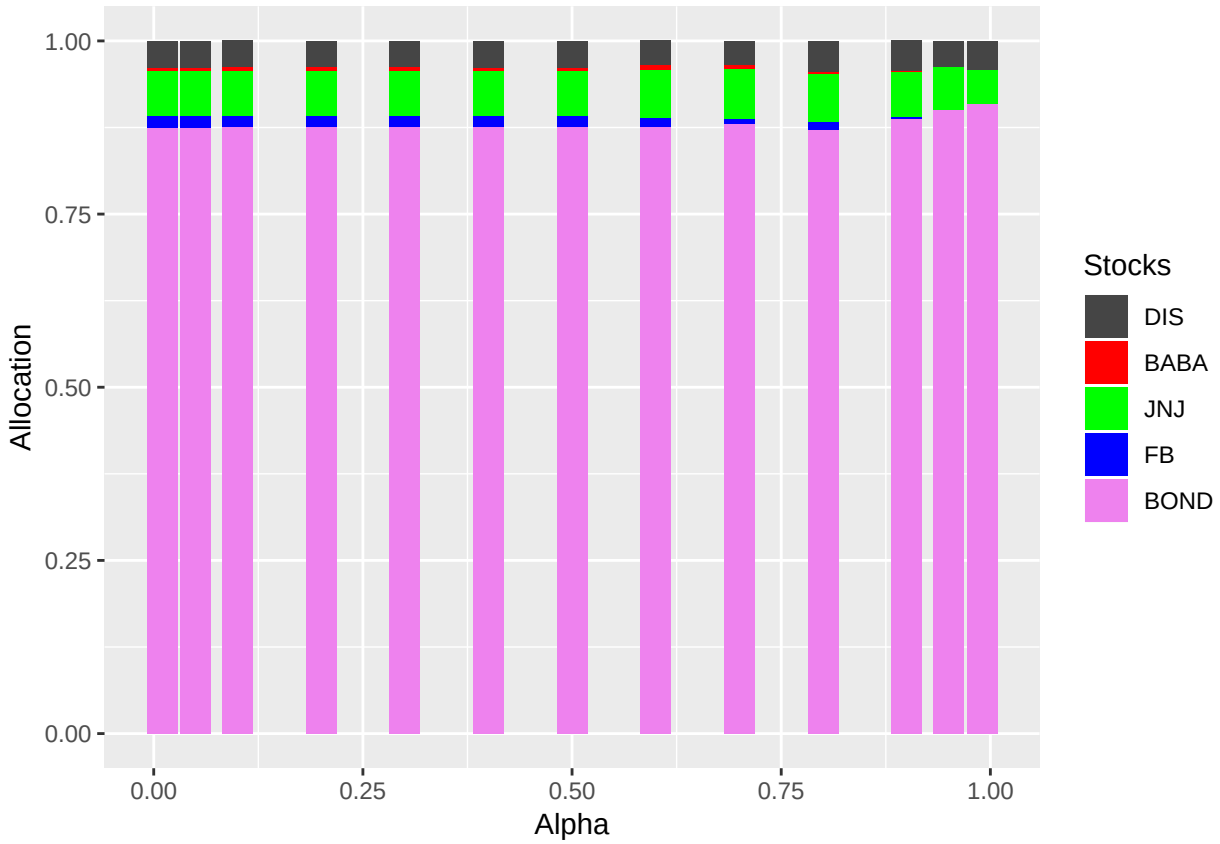
Solving and asset allocation for γfree = 0.9

```
gammafree9 = 0.9
dis_resultfree9 = vector(mode = "numeric")
baba_resultfree9 = vector(mode = "numeric")
jnj_resultfree9 = vector(mode = "numeric")
fb_resultfree9 = vector(mode = "numeric")
bond_resultfree9 = vector(mode = "numeric")
for (alp in alpha){
  objvectfree9 <- c((1-gammafree9)*t(prob)%*%Portreturnsfree,(-gammafree9/(1-alp))*t(prob),-gammafree9)
  names(objvectfree9)<-c("dis","baba","jnj","fb","bond",rep("z",826),"q")
  rhsconsfree9 <- c(rep(0,826 ),1,rep(0,831))
  Solutionfree9 <- solveLP(objvectfree9,rhsconsfree9,Amatfree,maximum = TRUE,
          const.dir = c(rep("<=",826),"=",rep("<=",831)),lpSolve = TRUE)
  dis_resultfree9<-append(dis_resultfree9,Solutionfree9$solution[1])
  baba_resultfree9<- append(baba_resultfree9,Solutionfree9$solution[2])
  jnj_resultfree9<-append(jnj_resultfree9,Solutionfree9$solution[3])
  fb_resultfree9<-append(fb_resultfree9,Solutionfree9$solution[4])
  bond_resultfree9<-append(bond_resultfree9,Solutionfree9$solution[5])
}
  resultfree9 <-data.frame(alpha,dis_resultfree9,baba_resultfree9,
                           jnj_resultfree9,fb_resultfree9,bond_resultfree9)
  write.csv(resultfree9, file = "Resultsfree9.csv")
  ggplot(resultfree9, aes(x=resultfree9$alpha)) +
    geom_area(aes(y=resultfree9$dis_resultfree9+resultfree9$baba_resultfree9+
```

```
                  resultfree9$jnj_resultfree9+resultfree9$fb_resultfree9+
                  resultfree9$bond_resultfree9, fill="bond"))+
    geom_area(aes(y=resultfree9$dis_resultfree9+resultfree9$baba_resultfree9+
                  resultfree9$jnj_resultfree9+resultfree9$fb_resultfree9, fill="fb")) +
    geom_area(aes(y=resultfree9$dis_resultfree9+resultfree9$baba_resultfree9+
                  resultfree9$jnj_resultfree9, fill="jnj")) +
    geom_area(aes(y=resultfree9$dis_resultfree9+resultfree9$baba_resultfree9,fill = 'baba')) +
    geom_area(aes(y=resultfree9$dis_resultfree9,fill = "dis"))+
    xlab("alpha levels") + ylab("Weights on the stocks") +
    labs(title="Asset allocation for gamma = 0.9") +  # title and caption
    scale_fill_manual("Stocks",
                      values =c("fb"="#0000FF","jnj"="#00FF00","baba"="#FF0000","dis"="#454545","bond"=
```




```
# Using stacked barplots
library(reshape2)
data9 <- resultfree9
names(data9) <- c("alpha","DIS","BABA","JNJ","FB","BOND")
mdata9 <- melt(data9, id=c("alpha"))
names(mdata9) <- c("Alpha","Stocks","Allocation")
ggplot() + geom_bar(aes(y = Allocation, x = Alpha, fill = Stocks), data = mdata9, stat="identity")+
  scale_fill_manual("Stocks",values =
                      c("FB"="#0000FF","JNJ"="#00FF00","BABA"="#FF0000","DIS"="#454545","BOND" = "viole
```



The formation of the models almost resembles each other in the different levels of γ and that is why in the different levels of γ, the programs almost resembles each other. Throughout the different levels of γ, what doesn't change is the **Amat** matrix and so you realize that we use the same Amat matrix throughout the programs for the different levels of γ.

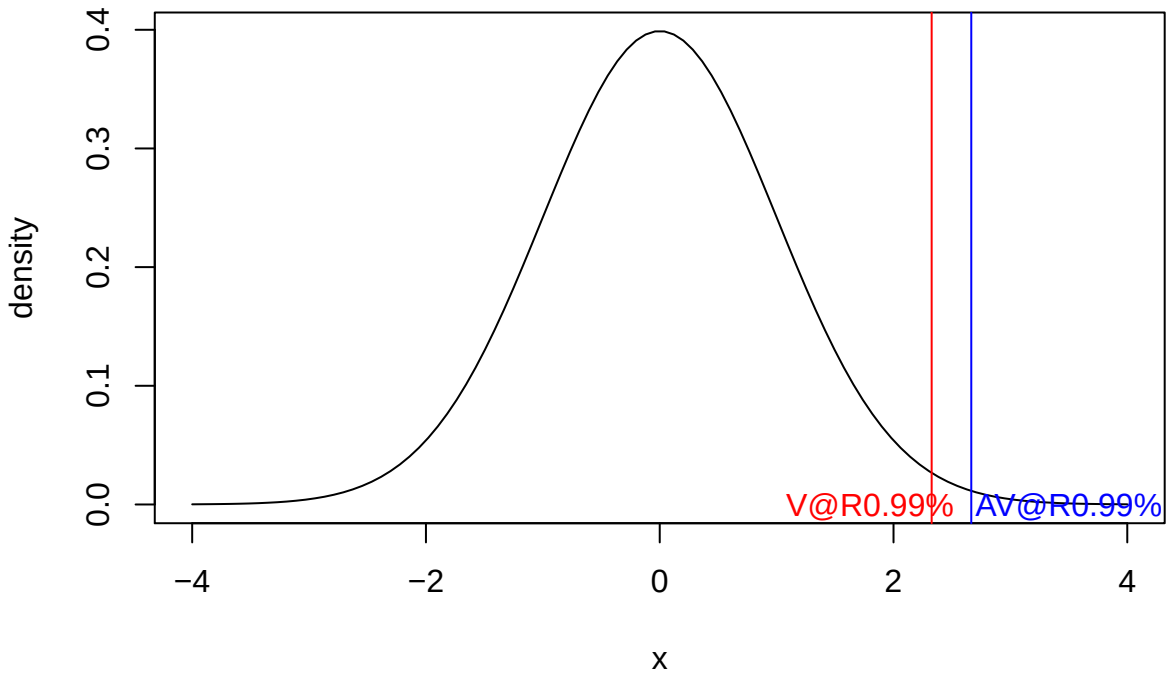
In the following, we want to just show the relationship between AV@R and V@R using a standard normally distributed data with mean = 0 and standard deviation = 1, It isn't part of this Integrated Management Formulation model but it is just part of the prperties of AV@R as always been greater than the V@R at all levels of α.

Relationship between the Average Value-at-Risk and Value-at-Risk

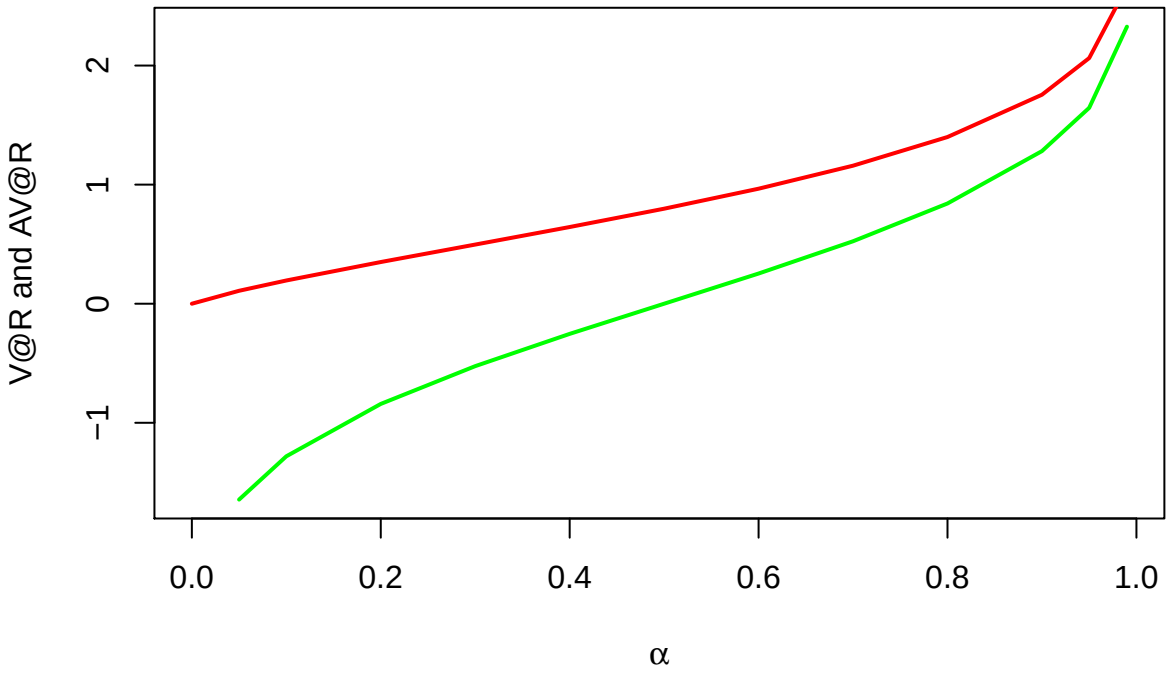
The following shows the relationship between Value-at-Risk and Average Value-at-Risk for a standard normal distribution data. The main agenda is to show that the AVaR is always greater than the VaR at all levels of alpha.

```
suppressMessages(suppressWarnings(library(QRM)))
mu = 0
sigma = 1
x <- seq(from = -4*sigma, to = 4*sigma, length.out = 100)
density <- dnorm(x, mean = mu, sd = sigma)
plot(x,density,type = "l")

VaR99 <- qnorm(0.99, mean = mu, sd = sigma) #Value-at-Risk at level 99%
ES99 <- ESnorm(0.99, mu = mu, sd = sigma) #Average Value-at-Risk at level 99%

abline(v = VaR99, col = "red")
```

```
text(1.8,0.0, "V@R0.99%", col = "red")
abline(v = ES99, col = "blue")
text(3.5,0.0, "AV@R0.99%", col = "blue")
```




```
var <- vector(mode = "numeric")
avar <- vector(mode = "numeric")
alpha <- c(0.0,0.05,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.90,0.95,0.99)
for (alp in alpha){
  var <- append(var,qnorm(alp,mean = mu, sd = sigma))
  avar <- append(avar,ESnorm(alp,mu = mu , sd = sigma))
}
plot(alpha,var,type="l", col="green", xlab = expression(alpha), ylab = "V@R and AV@R",lwd = 2)
lines(alpha,avar,col = "red",lwd = 2)
```


V@R and AV@R
2
1
0
−1
0.0
0.2
0.4
0.6
0.8
1.0
α