Final Project Statistical Algorithm

Eugeniah Arthur

11/28/2020

Verification Procedure of the least Squares Solver.

In this analysis, verification of the LLS solvers based on exactly representable solutions was explored.

The code below was run for pairs (n=4,k=1),(n=8,k=3) ,(n=16,k=7),(n=32, k=15)and (n=64, k=31) for m =0 to 20 where z= 2^m. Non consecutive rows were deleted to create the Zielke Matrices with different condition numbers. Also, general response vectors with different magnitude of angles between a randomly sampled response and Im(X) were generated. These were used to find the relative errors betweeen the LLS solution and the true parameter values whose exact solution could be found. The simulation was run a 100 times for each pair; the relative errors, mean relative errors and the standard deviation of the relative errors of each iterate was stored. Overall, 105 mean relative errors were generated and there was 4 columns of relative errors signifying a very small angle type (err1), medium small angle type (err2), medium large angle type(err3) and the very large angle type (err4).

Code for Generating relative errors

```
source("zielke1.r")
n=64
k=31



for (m in 0:20){
seedind<-m+(n*k)
   set.seed(seedind)
Z <- 2^m

l=seq(2,by=2,length.out=k) # columns to delete

X1 <- XZ1(Z, n)
X <- t(X1[-l, ])
  #test matrix X on page 126
G <- XZ1plus(Z, n)
Xplus <- t(G_rows_deleted(G, l))
```

```
#Calculating Fibronius Condition Number
library(MASS)


xF2 <- sum(X^2)
xplusF2 <- sum(Xplus^2)
KF <- sqrt(xF2 * xplusF2)


LLS.lm <- function(X, y) {
  dat <- data.frame(y=y, X=X)
  L <- lm(y ~ ., data=dat)
  b <- L$coefficients
  m <- length(b)
  b[m] <- - b[1]
  b[1] <- b[1] + b[2]
  b <- b[-2]
  return(b)
}




# Simulation

R <- 100
n1 <- nrow(X)

out <- replicate(R, expr={
  xi <- numeric(4)
  y <- sample(-1000:1000, size=n1, replace=TRUE)

  b0<- Xplus %*% y
  y0 <- X %*% b0
  r0 <- y - y0
  Y <- responses(y0, r0)

  for (i in 1:4) {
    yi <- Y[,i]
    bi <- LLS.lm(X, yi)
    num <- sum((b0 - bi)^2)
    den <- sum(b0^2)
    xi[i] <- sqrt(num / den)
    }
  xi
}
)
```

```
## Appending of report

tout=t(out)


#seed_i=seedind
mean_i=cbind(t(colMeans(tout)),seed_i=seedind,KF,n,Z,k)
sd_i=cbind(t(apply(tout, 2, sd)),seed_i=seedind,KF,n,Z,k)


colnames(mean_i)<-c("err1","err2","err3","err4","Seed_i","Condition
number","n","z","k")

colnames(sd_i)<-c("err1","err2","err3","err4","Seed_i","Condition
number","n","z","k")
write.table(mean_i,file="MeanSimS11.txt",sep="\t", row.names = FALSE,col.names=!
file.exists("MeanSimS11.txt"),append=T)
write.table(sd_i,file="sdSimS11.txt",sep="\t", row.names = FALSE,col.names=!
file.exists("sdSimS11.txt"),append=T)
ind<-replicate(R,expr={seedind})
colnames(tout)<-c("err1","err2","err3","err4")
condition_Number<-KF
err=cbind(tout,ind,condition_Number)

write.table(err,file="trialsimS11.txt",sep="\t", row.names = FALSE,col.names=!
file.exists("trialsimS11.txt"),append=T)

}
```

Data Storage

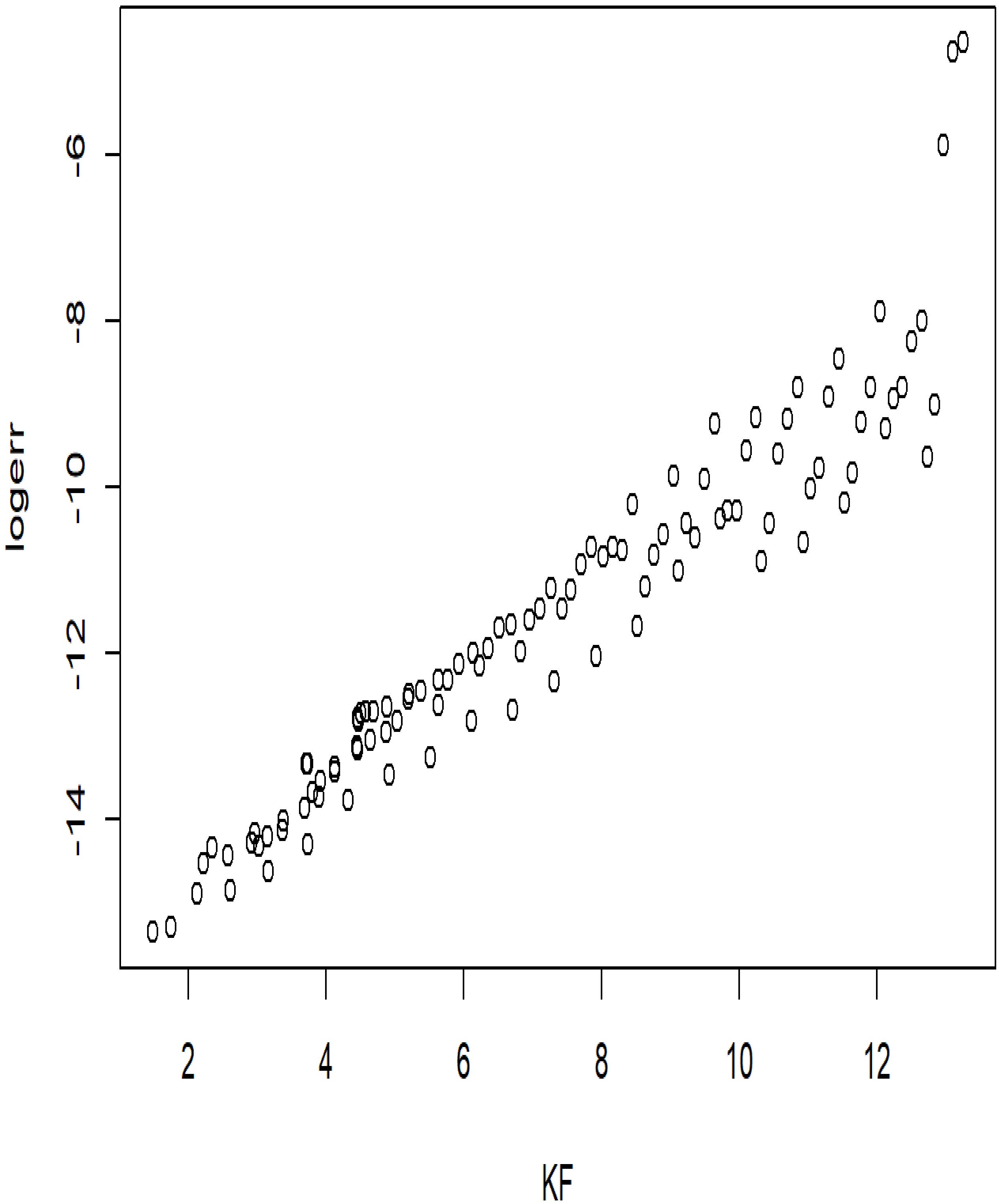
The output of the mean and standard deviation of the relative errors are put into the file “MeanSimS11.txt” and “sdSimS11.txt”. Also, the 100 relative errors of each iteration was put in a file called “trialsimS11.txt”. All the files had columns for the random number seed, the n,z,k and the condition numbers.

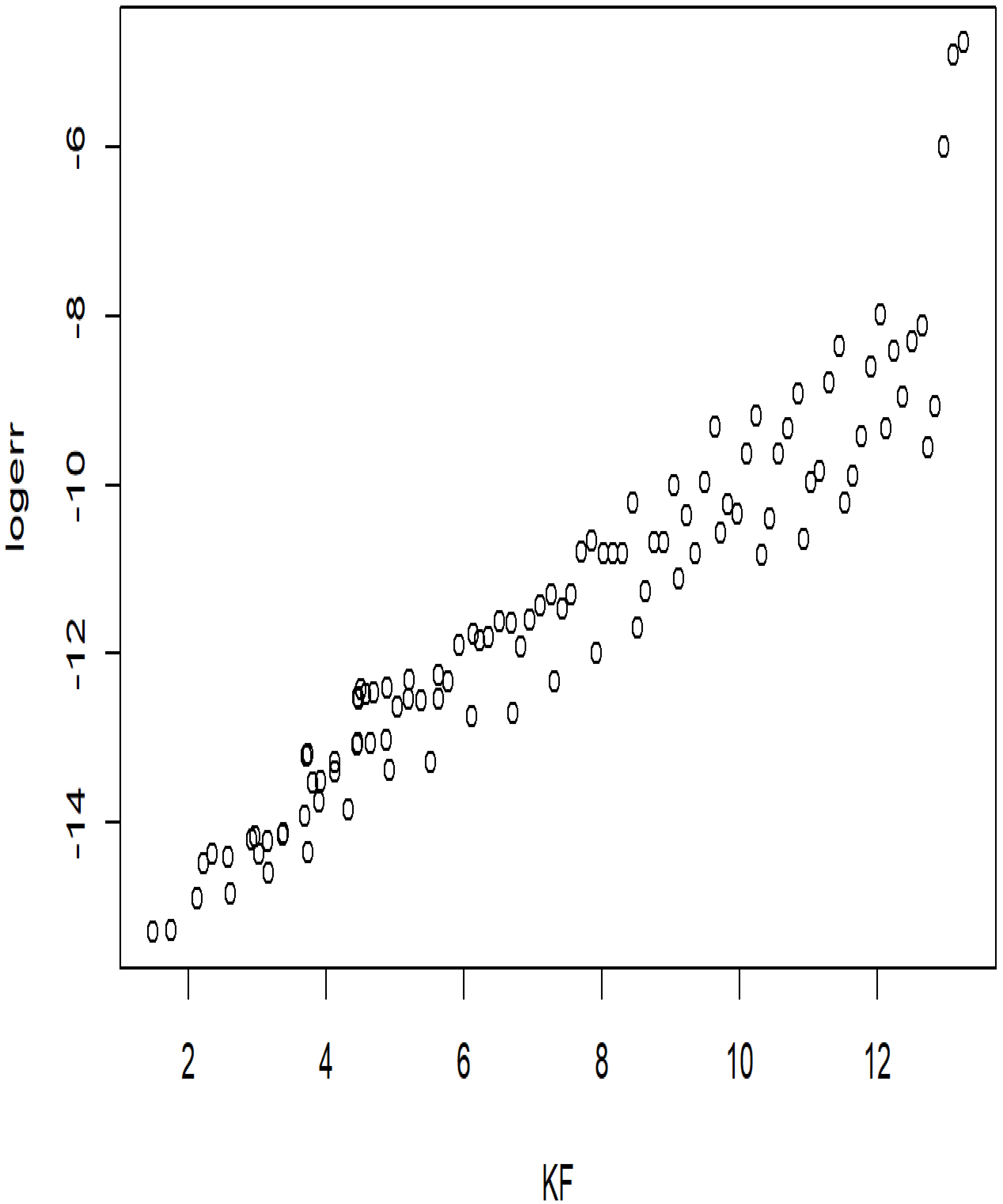
Plot of the Mean Relative errors with Condition Numbers.

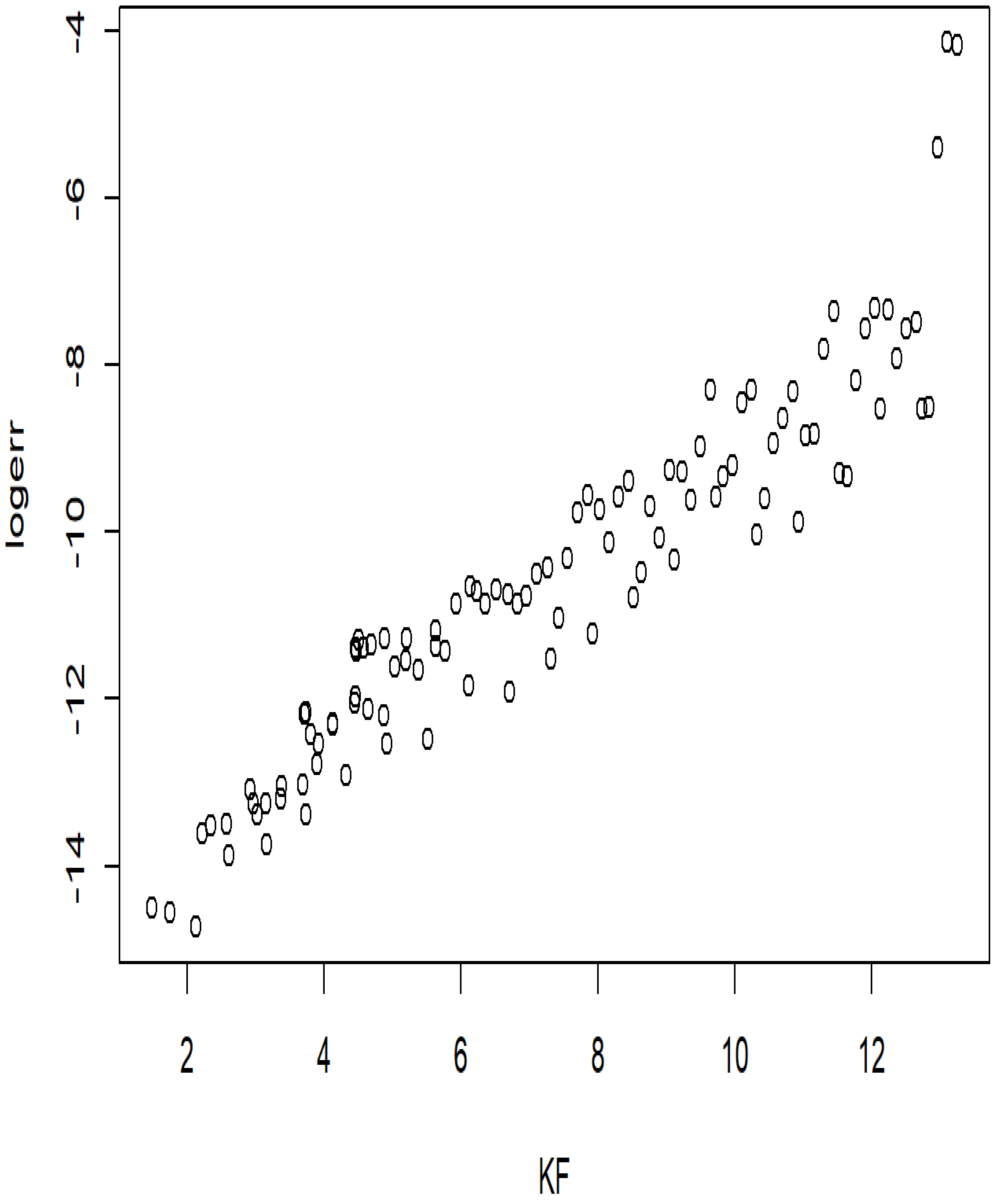
```
MeanSim11 <- read.delim("C:/Users/eugarth/Downloads/MeanSimS11.txt")
head(MeanSim11)

##           err1         err2         err3         err4 Seed_i
## 1 4.370899e-16 5.126978e-16 3.167311e-15 7.969928e-10      4
```

```
## 2 4.959322e-16 5.335832e-16 2.751068e-15 7.425452e-10      5
## 3 1.260665e-15 1.290673e-15 1.876036e-15 3.868972e-10      6
## 4 1.392193e-15 1.444507e-15 1.330723e-14 3.307123e-09      7
## 5 2.355147e-15 2.583912e-15 1.796305e-14 4.723452e-09      8
## 6 4.918936e-15 4.493915e-15 4.206033e-14 1.101777e-08      9
##   Condition.number n  z k
## 1         29.98333 4  1 1
## 2         54.67175 4  2 1
## 3        133.19535 4  4 1
## 4        407.60397 4  8 1
## 5       1426.61452 4 16 1
## 6       5345.80480 4 32 1
```

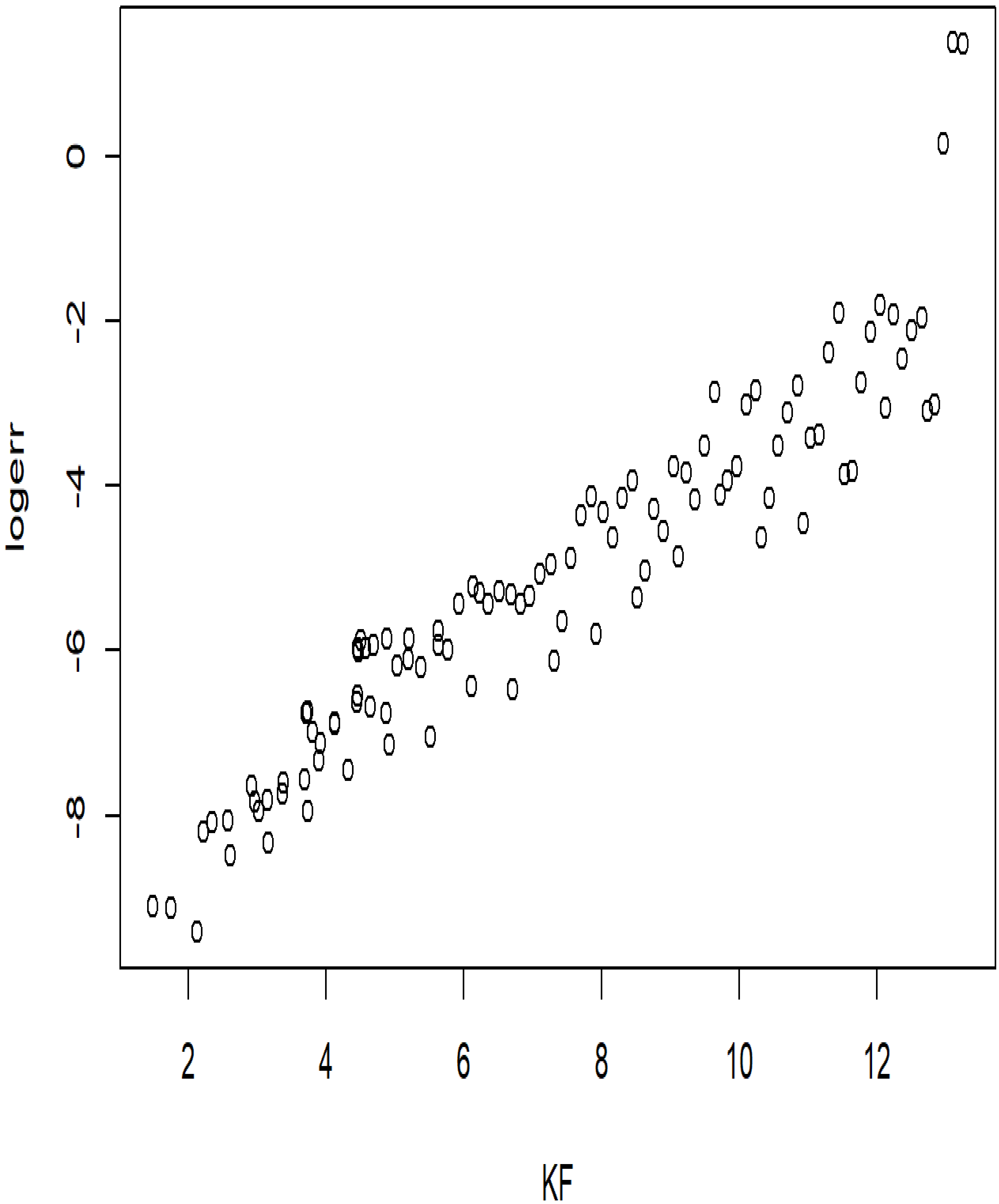
```
for(i in 1:4){
  n=nrow(MeanSim11)
 logerr=numeric(n)
logerr=log10(MeanSim11[,i])
KF=log10(MeanSim11$Condition.number)
plot(KF,logerr)
}
```


logerr
-6
-8
-10
-12
-14
2
4
6
8
10
12
KF


logerr
-6
-8
-10
-12
-14
2
4
6
8
10
12
KF

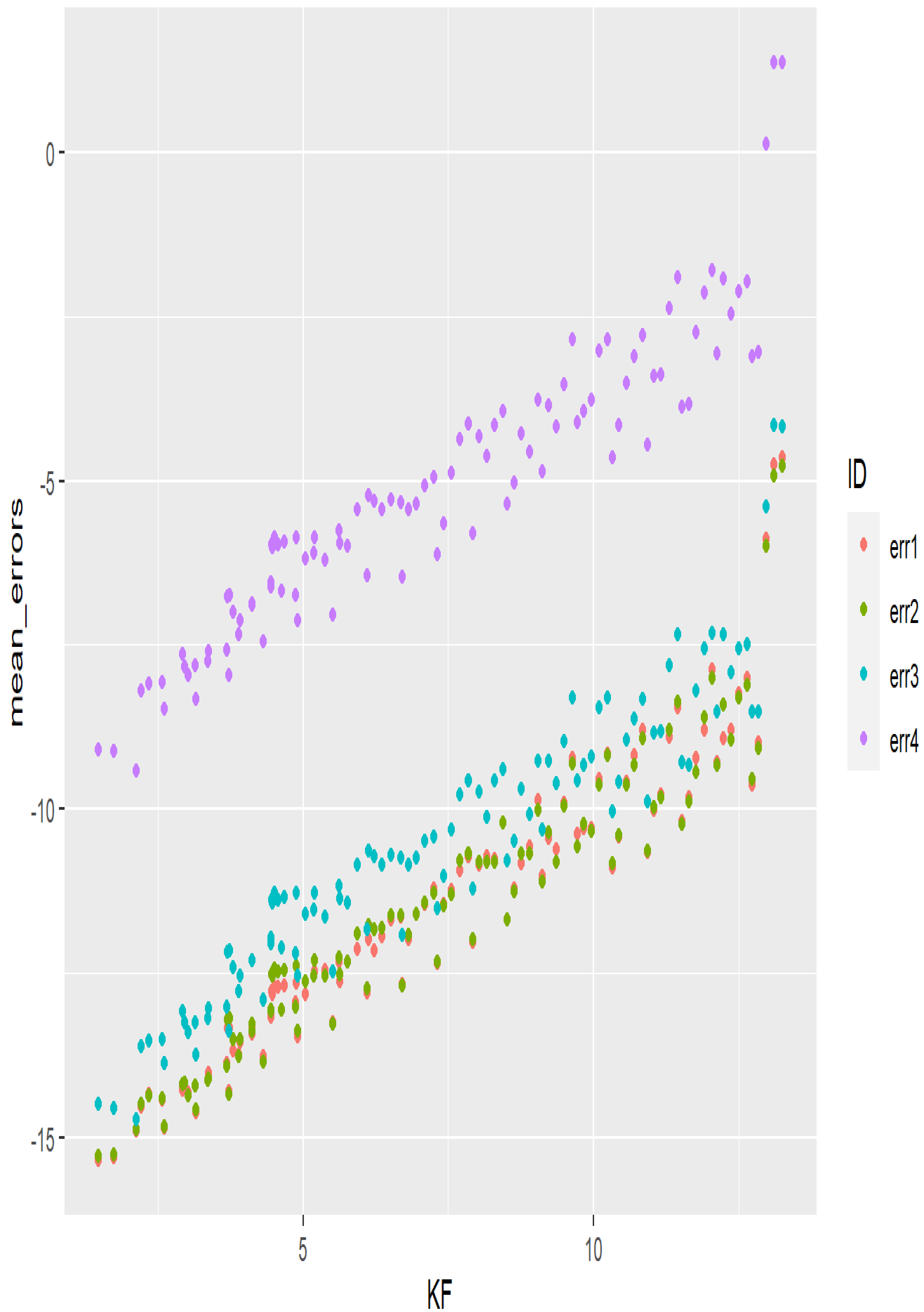


logerr
-4
-6
-8
-10
-12
-14
2
4
6
8
10
12
KF

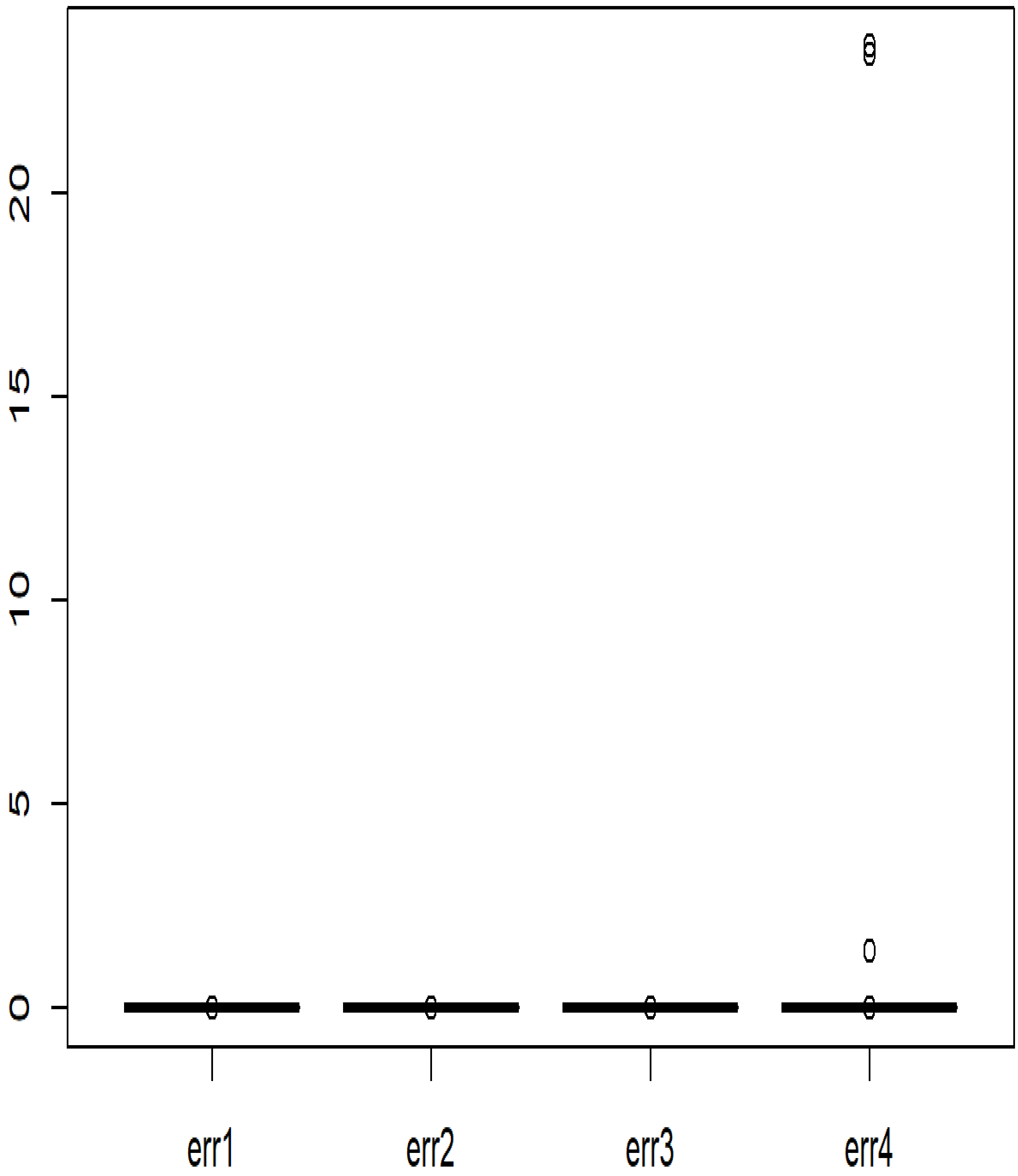



```
library(ggplot2)
err1 =log10(MeanSim11$err1)
err2 = log10( MeanSim11$err2)
```

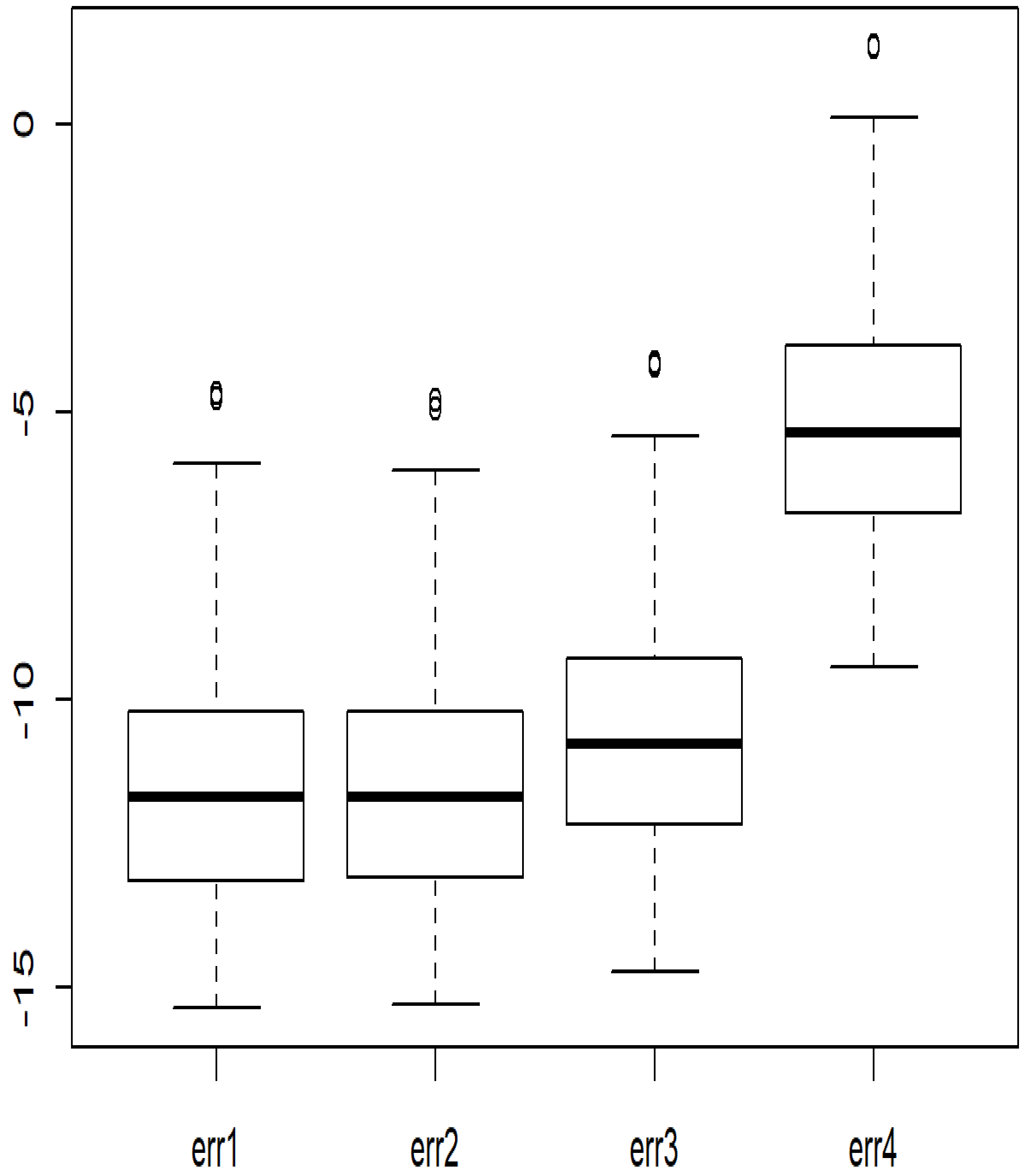
```
err3 =  log10(MeanSim11$err3)
err4 =  log10(MeanSim11$err4)
a_list = list(err1,err2,err3,err4)
a_df = do.call("rbind", lapply(a_list,
                                function(x) data.frame(mean_errors = x,
                                                      KF=
log10( MeanSim11$Condition.number))))
ID_options =c("err1","err2","err3","err4")
a_df$ID = rep(ID_options, sapply(a_list, length))
ggplot(a_df, aes(x =  KF,y = mean_errors, color = ID)) + geom_point()
```




```
boxplot(MeanSim11[,c(1,2,3,4)])
```




```
boxplot(log10(MeanSim11[,c(1,2,3,4)]))
```



0
-5
-10
-15
err1
err2
err3
err4

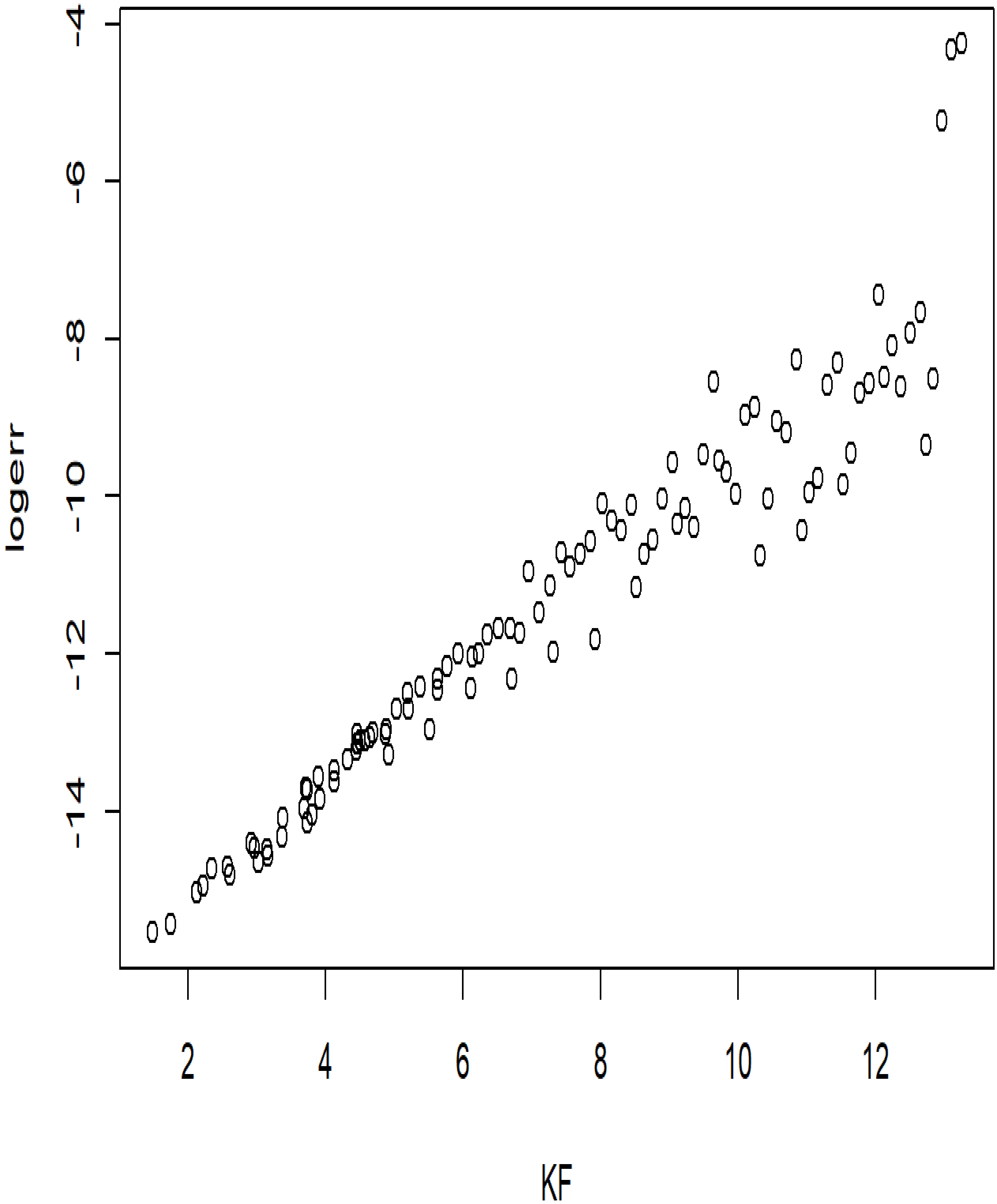
Comment on the plots of Mean Relative Errors and Condition numbers.

The plots show dependency of the relative error on the condition number. It is noticed that as the condition number increased the average relative errors increased. Thus, if the condition number of the test Matrix is high , then LLS solution is expected to deviate highly from the true solution.There seems to be a linear dependency between the condition numbers and the average relative errors. Furthermore, as the angles get larger(from y1 to y4), the mean relative errors becomes larger for a given condition number. However,the very small and medium small angle relative errors are pretty close and hence appears to overlap in the combined plot.

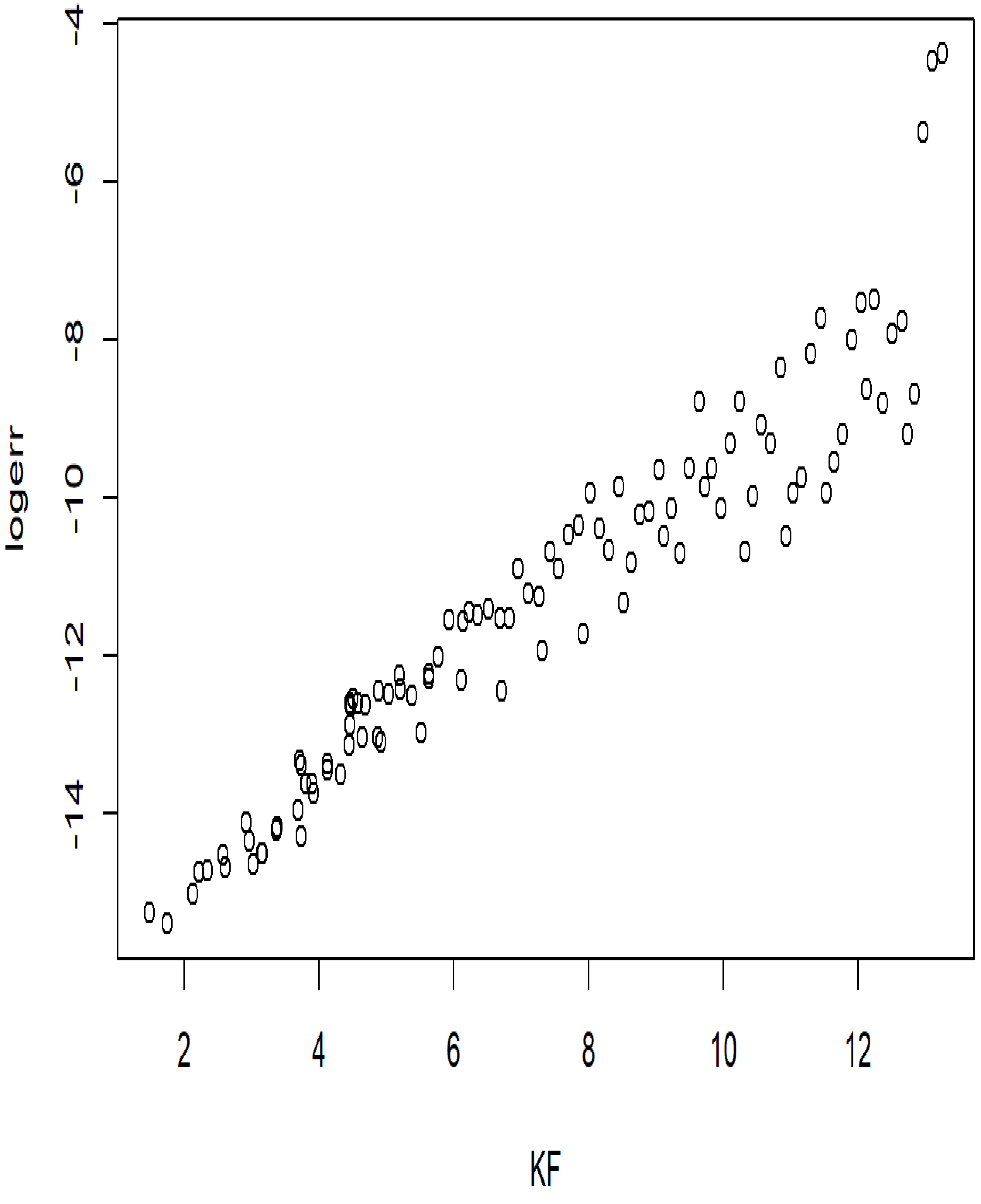
```
sdSimS11 <- read.delim("C:/Users/eugarth/Downloads/sdSimS11.txt")
head(sdSimS11)
```

```
##           err1         err2         err3         err4 Seed_i
## 1 2.890552e-16 5.549744e-16 3.720271e-15 8.890968e-10      4
## 2 3.705162e-16 4.060880e-16 2.171440e-15 7.065606e-10      5
## 3 9.230994e-16 9.810162e-16 1.760256e-15 7.445769e-10      6
## 4 1.554851e-15 2.116668e-15 2.354557e-14 4.881374e-09      7
## 5 2.660073e-15 3.063508e-15 2.618106e-14 6.850731e-09      8
## 6 7.033327e-15 5.182209e-15 8.443297e-14 2.174180e-08      9
##   Condition.number n  z k
## 1         29.98333 4  1 1
## 2         54.67175 4  2 1
## 3        133.19535 4  4 1
## 4        407.60397 4  8 1
## 5       1426.61452 4 16 1
## 6       5345.80480 4 32 1
```

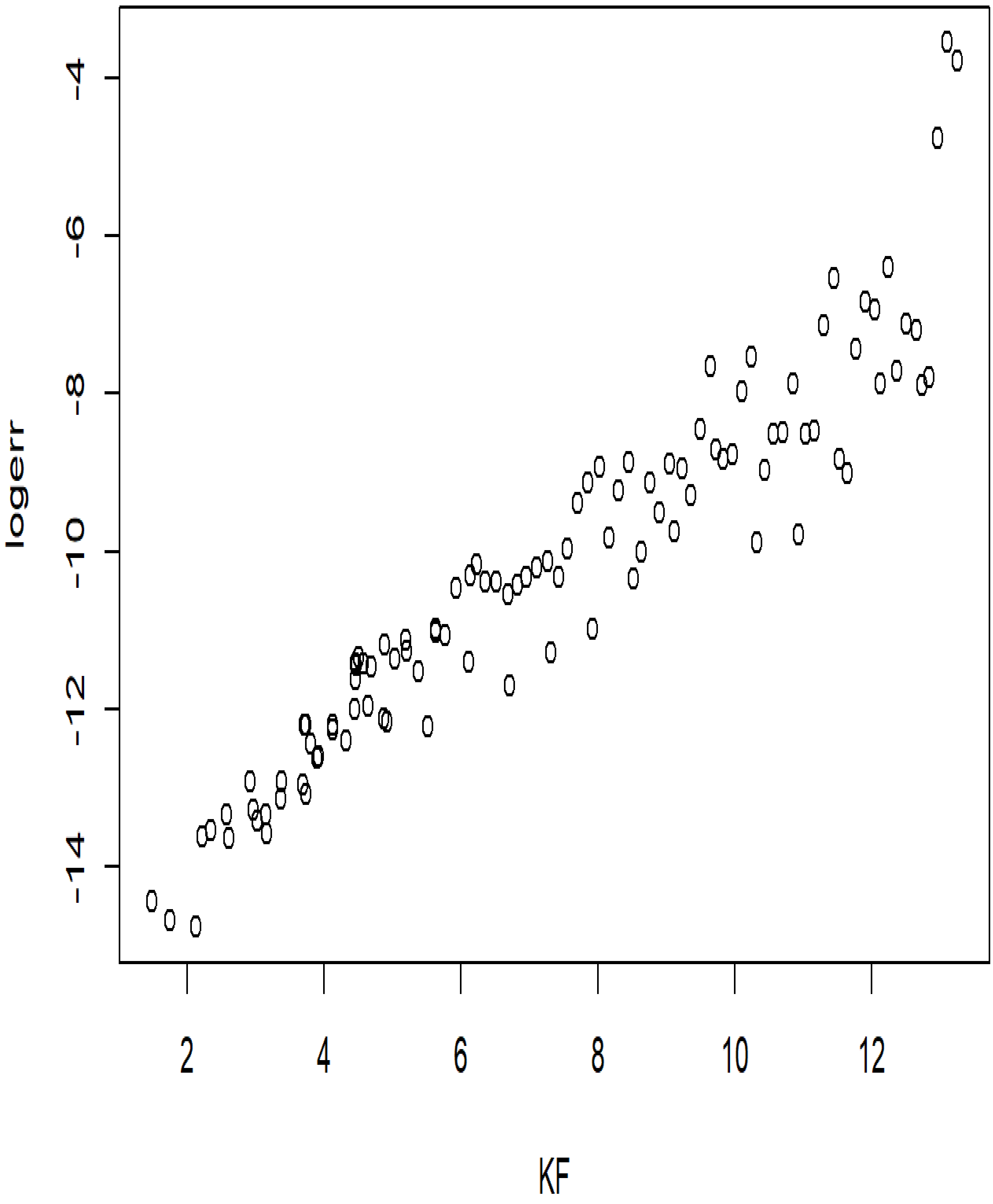
```
for(i in 1:4){
  n=nrow(sdSimS11)
 logerr=numeric(n)
logerr=log10(sdSimS11[,i])
KF=log10(sdSimS11$Condition.number)

plot(KF,logerr)
}
```



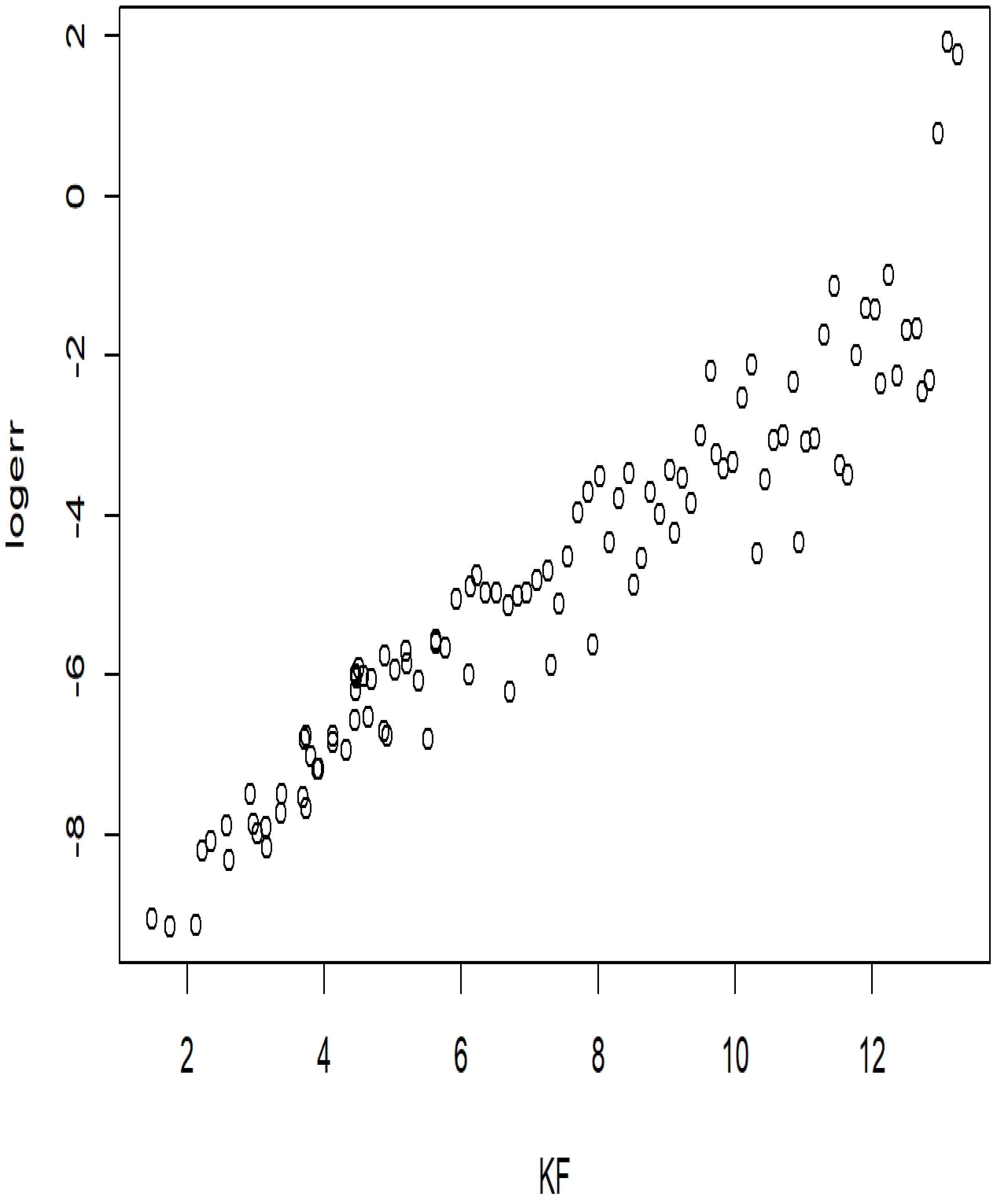
logerr
-4
-6
-8
-10
-12
-14
2
4
6
8
10
12
KF



logerr
-4
-6
-8
-10
-12
-14
2
4
6
8
10
12
KF

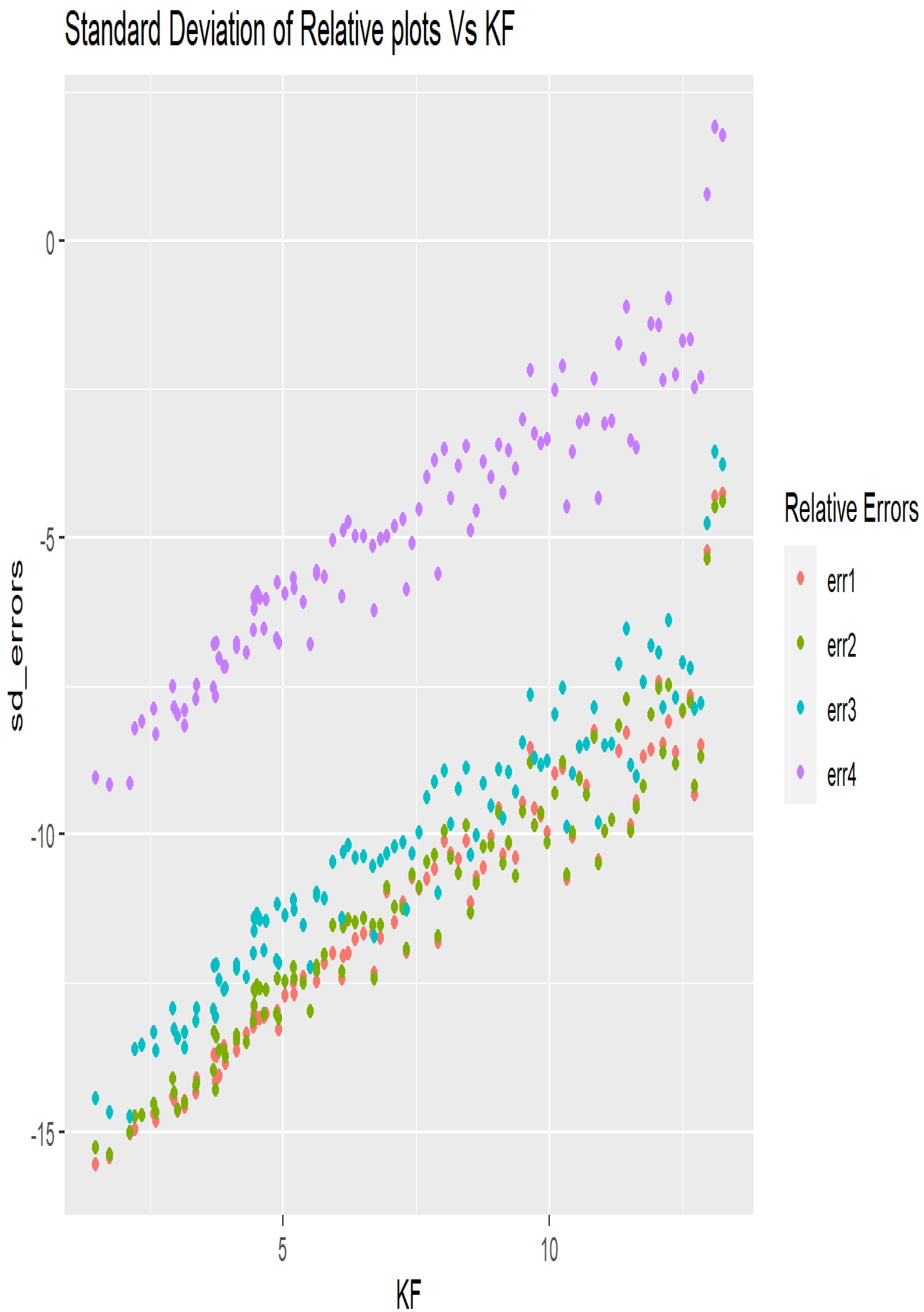


-4
-6
-8
-10
-12
-14
logerr
2
4
6
8
10
12
KF

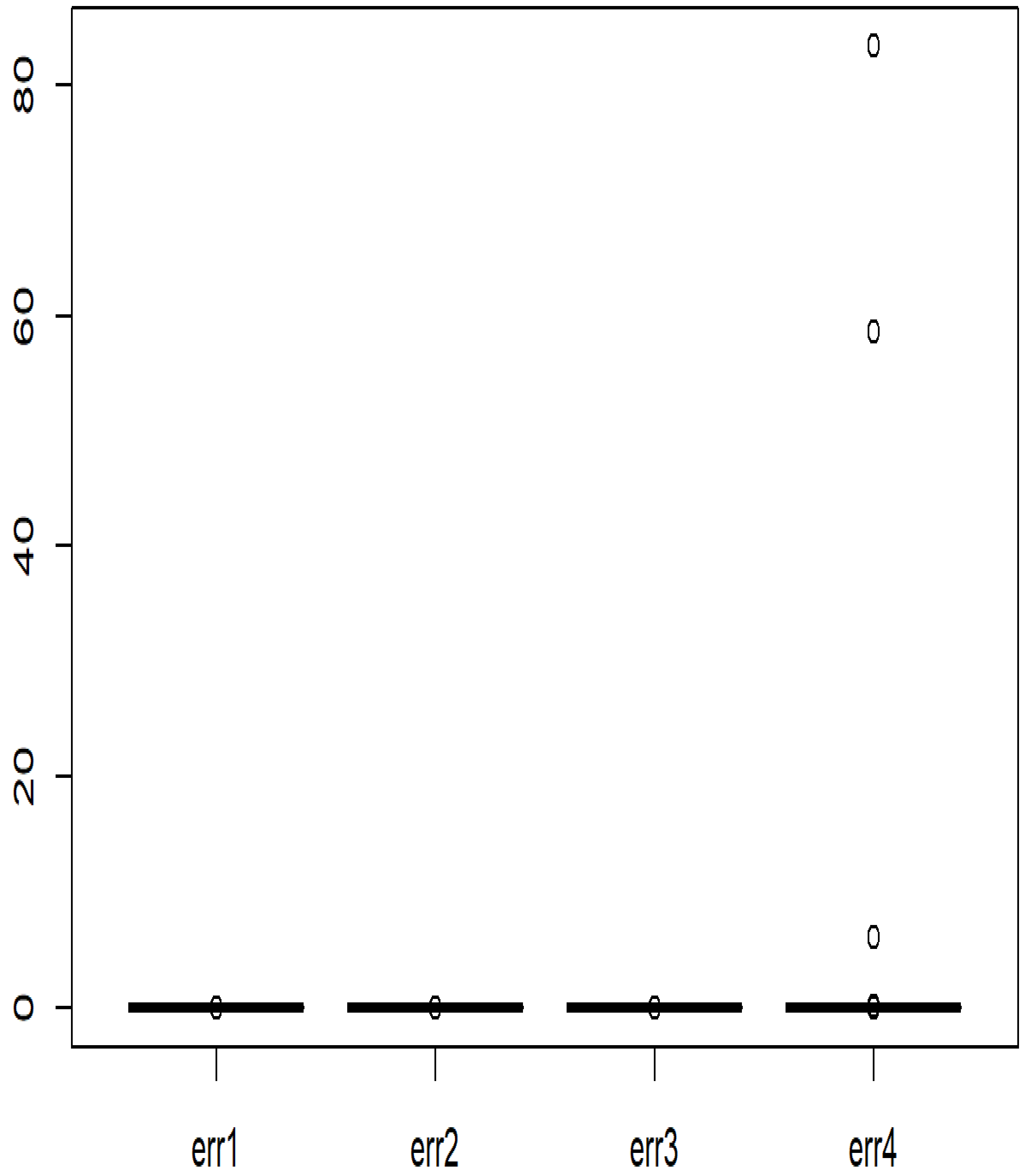



```
serr1 =log10(sdSimS11$err1)
serr2 = log10( sdSimS11$err2)
serr3 =  log10(sdSimS11$err3)
```

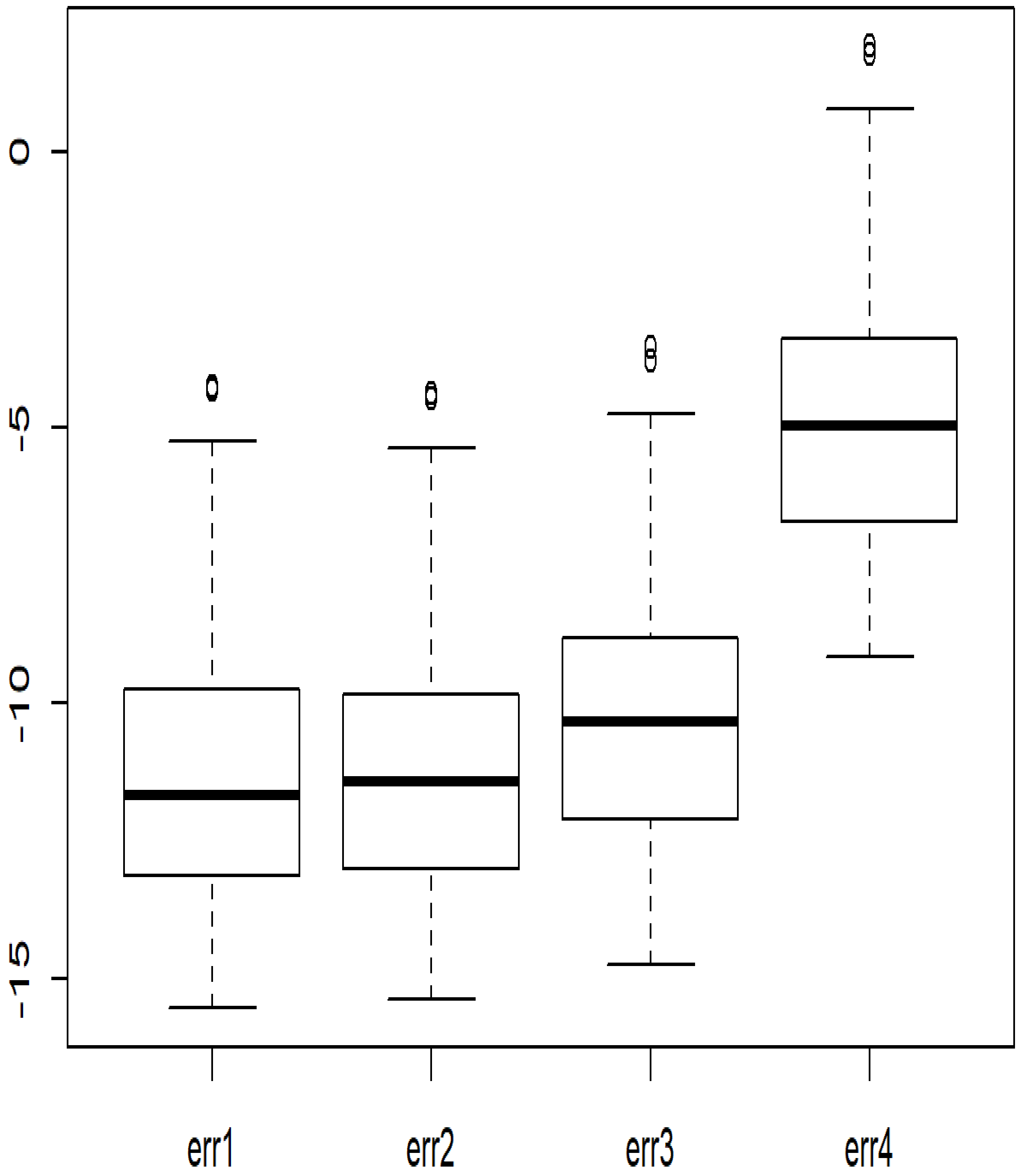
```
serr4 =  log10(sdSimS11$err4)
a_list = list(serr1,serr2,serr3,serr4)
a_df = do.call("rbind", lapply(a_list,
                                function(x) data.frame(sd_errors = x,
                                                       KF=
log10( sdSimS11$Condition.number))))
ID_options =c("err1","err2","err3","err4")
a_df$ID = rep(ID_options, sapply(a_list, length))
sdplot=ggplot(a_df, aes(x =  KF,y = sd_errors, color = ID)) + geom_point()
print(sdplot + labs(colour = "Relative Errors") +ggtitle("Standard Deviation of
Relative plots Vs KF"))
```




```
boxplot(sdSimS11[,c(1,2,3,4)])
```




```
boxplot(log10(sdSimS11[,c(1,2,3,4)]))
```


0
-5
-10
-15
err1
err2
err3
err4

Comment on Plot of Standard Deviation of Relative Errors and Condition numbers .

The plot of the standard deviation of the relative errors verses condition numbers also shows that the the standard deviation increases as the condition number increases. Also, as the angle increases the standard deviation of the relative error increases. The larger angled relative error (err4) appears to have a large overall standard deviation compared to the other smaller angles.