Homework 9.2 and 9.3

Modern Statistics and Big Data Analytics

Federico Veronesi

2 a) The maximum breakdown point for the Median is 0.5, as it is shown in the slides.

If more than n/2 points are contaminated, the median is contaminated too and the values |X_i – Median(X)| can become arbitrarily large. Therefore the breakdown point of the MAD is 0.5, exactly equal to the one of the Median.

2 b) My guess for the breakdown point of the IQR is 0.25, since the smallest 25% and the larger 25% of the observations don't enter in the formula. They can be arbitrarily far away from the rest of the observations, but the IQR estimator remains the same. For this reason 0.25*n is the maximum number of points that can be substituted in order to NOT break down the estimator. -> BP=0.25*n/n = 0.25

2 c) The α-trimmed mean excludes from the computation of the mean the first α*n observations and the last α*n observations, when data are ordered and with 0<α<0.5. My guess for the Breakdown Point is the parameter α. In a similar way to point 2b, the maximum number of points that the estimator can "handle" is α*n. As bedore, the first α*n obs. and the last α*n obs. don't partecipate to the computation of the mean. -> BP = α*n/n = α. This means that when we increase α, we increase the BP and, thus, the robustness of the estimator, but at the same time we decrease the efficiency since we use less information.

3)

ARE	"N(0,1)"	"t2"	"t4"
mean, median	1.43890109677031	0.0909002839715225	0.955525295995126
mean, huberM	1.04613590711297	0.101982174873521	0.800277458378294
median, huberM	0.727038091402583	1.12191261036622	0.837526187671306

The mean has higher relative efficiency with respect the other two estimators when dealing with a normal (0,1). We expected this result since the mean is the best location estimator for normal distributions. When dealing with t distributions (t2 has heavier tails and, thus, more outliers than t4) the mean is less efficient i.e. has higher variability. HuberM performs better than the median with normal distributions and t4, while the median has the highest efficiency in presence of more extreme outliers (in the t2 distribution case). Overall, the Huber seems a very good trade-off between the two, because its relative efficiency in the two extreme situation (normal(0,1) and t2) is not so bad wr to the optimal estimators, and in the t4 situation it outperforms the median and the mean.

R CODE:

```
### Ex.3
```

```
library(robustbase)
nrep <- 1000

set.seed(250)
listdata1 <- list()
meanlist <- c()
medianlist <- c()
huberlist <- c()
for (i in 1:nrep){
  listdata1[[i]] <- rnorm(20)
  meanlist[i] <- mean(listdata1[[i]])
  medianlist[i] <- median(listdata1[[i]])
  huberlist[i] <- huberM(listdata1[[i]])$mu
}

releffmeanmedian1 <- var(medianlist)/var(meanlist)
releffmeanhuber1 <- var(huberlist)/var(meanlist)
releffhubermedianhuber1 <- var(huberlist)/var(medianlist)

listdata2 <- list()
for (i in 1:nrep){
  listdata2[[i]] <- rt(n=20, df=2)
  meanlist[i] <- mean(listdata2[[i]])
  medianlist[i] <- median(listdata2[[i]])
  huberlist[i] <- huberM(listdata2[[i]])$mu
}

releffmedian2 <- var(medianlist)/var(meanlist)
releffhuber2 <- var(huberlist)/var(meanlist)
releffhubermedianhuber2 <- var(huberlist)/var(medianlist)

listdata3 <- list()
for (i in 1:nrep){
  listdata3[[i]] <- rt(n=20, df=4)
  meanlist[i] <- mean(listdata3[[i]])
  medianlist[i] <- median(listdata3[[i]])
  huberlist[i] <- huberM(listdata3[[i]])$mu
}

releffmedian3 <- var(medianlist)/var(meanlist)
```

```
releffhuber3 <- var(huberlist)/var(meanlist)

releffhubermedianhuber3 <- var(huberlist)/var(medianlist)


table3 <- matrix(c(releffmedian1, releffhuber1, releffhubermedianhuber1, releffmedian2, releffhuber2, releffhubermedianhuber2, releffmedian3, releffhuber3, releffhubermedianhuber3), nrow = 3)

rownames(table3) <- c("mean, median", "mean, huberM", "median, huberM")

colnames(table3) <- c("N(0,1)", "t2", "t4")

write.table(table3, file = "tableex3.txt")
```