analysis

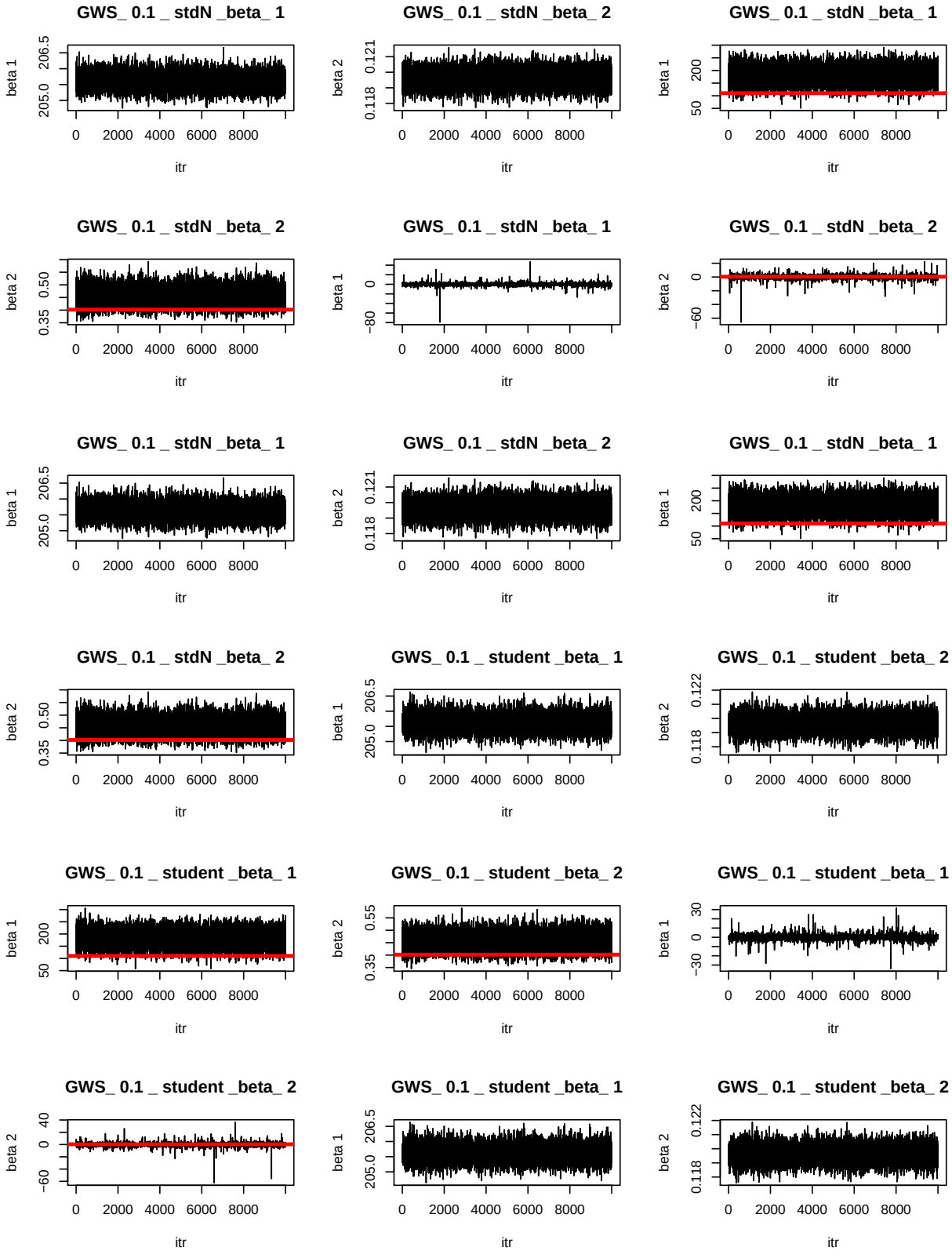
Setup

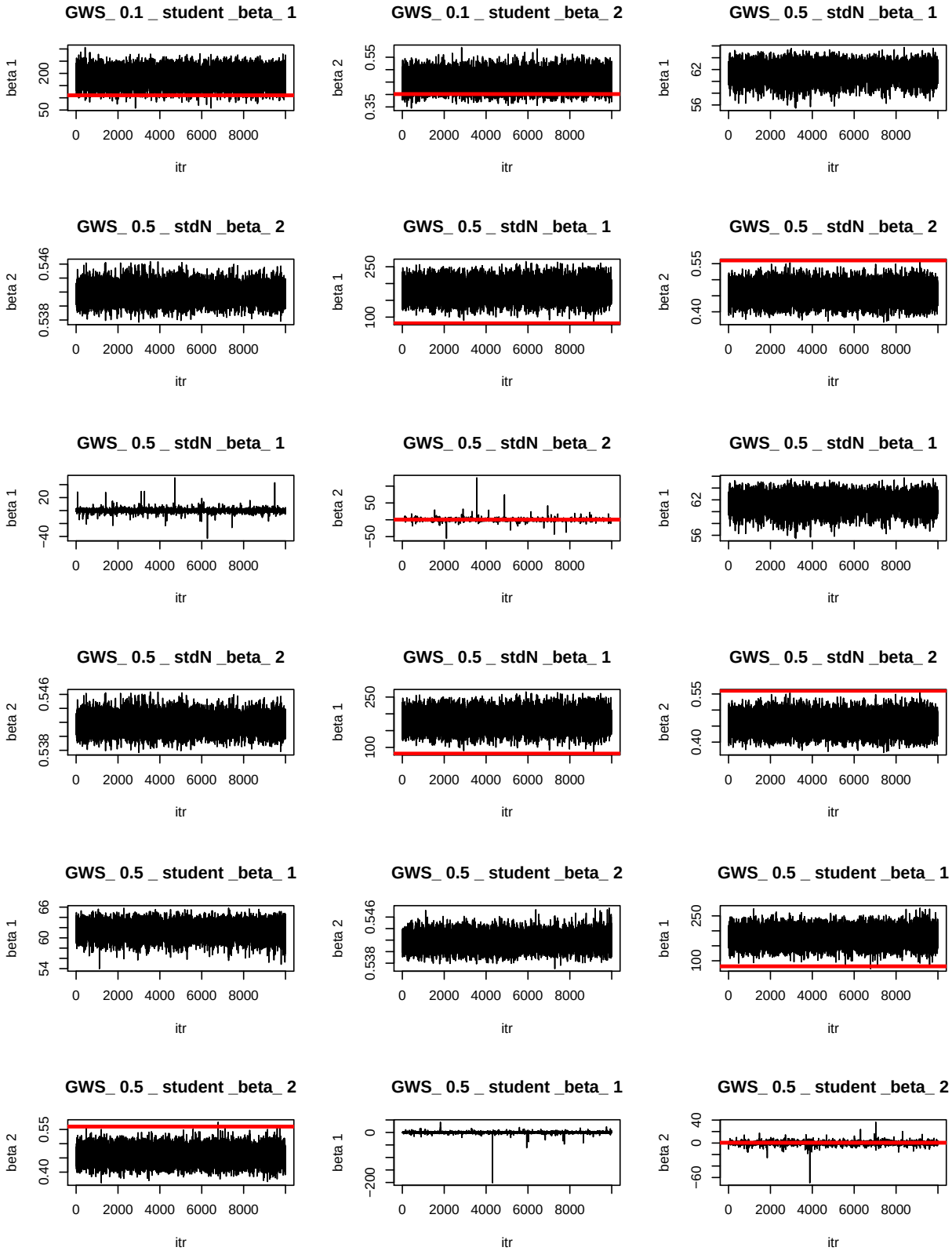
```
itr_total = 12000
quantiles=c(0.1, 0.5, 0.9)
dist_eps_list=c("stdN","student")
d=2
burn_in = 2000

simulation1 = data.frame("Distribution_Eps"=character(0), "Method"=character(0),
                         "p"=numeric(),
                         "beta1_MCSE"=numeric(), "beta1_IF"=numeric(), "beta1_mu"=numeric(),
                         "beta1_sd"=numeric(), "beta1_last"=numeric(),
                         "beta2_MCSE"=numeric(), "beta2_IF"=numeric(), "beta2_mu"=numeric(),
                         "beta2_sd"=numeric(),
                         "beta2_last"=numeric(), stringsAsFactors=F)
```

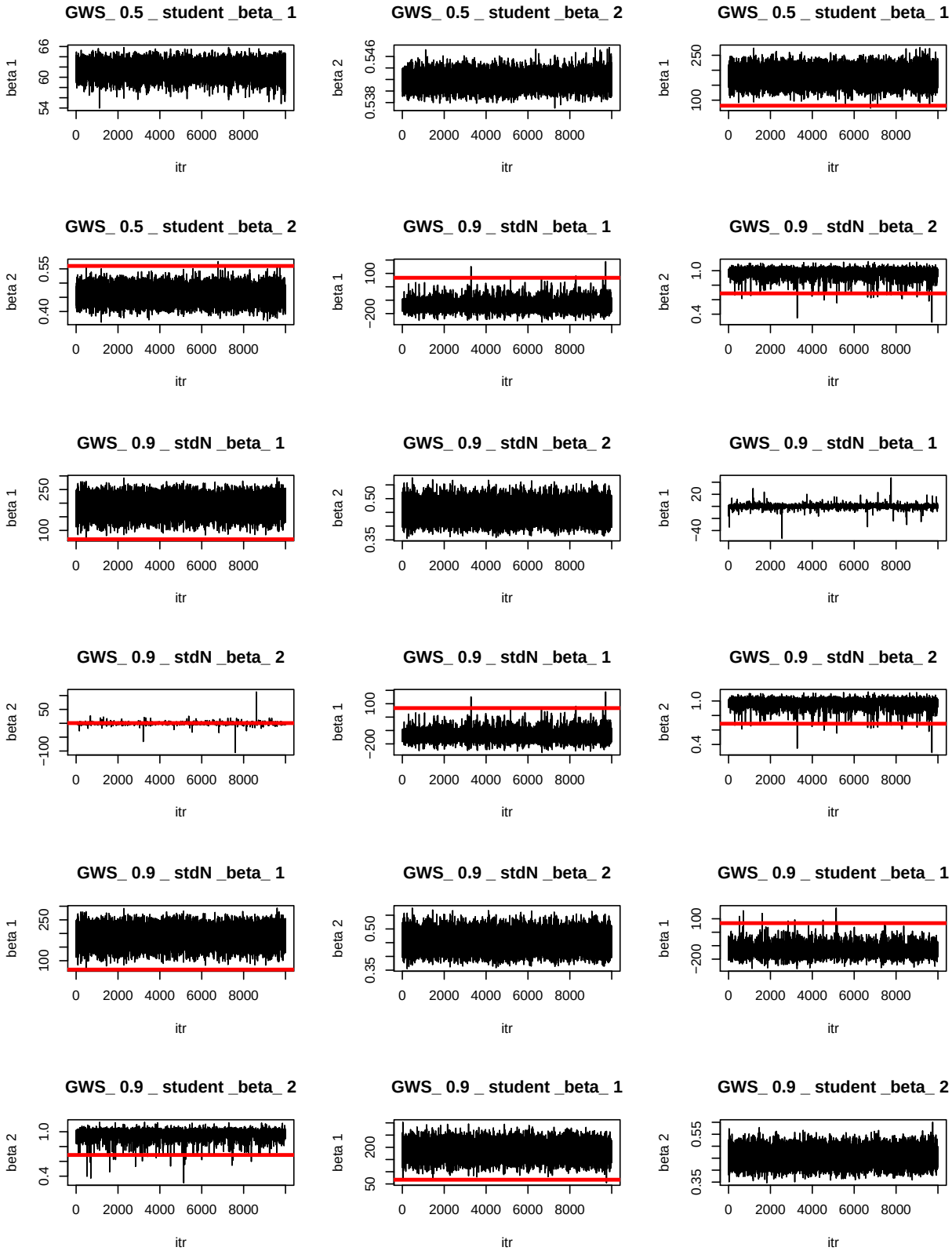
Append data

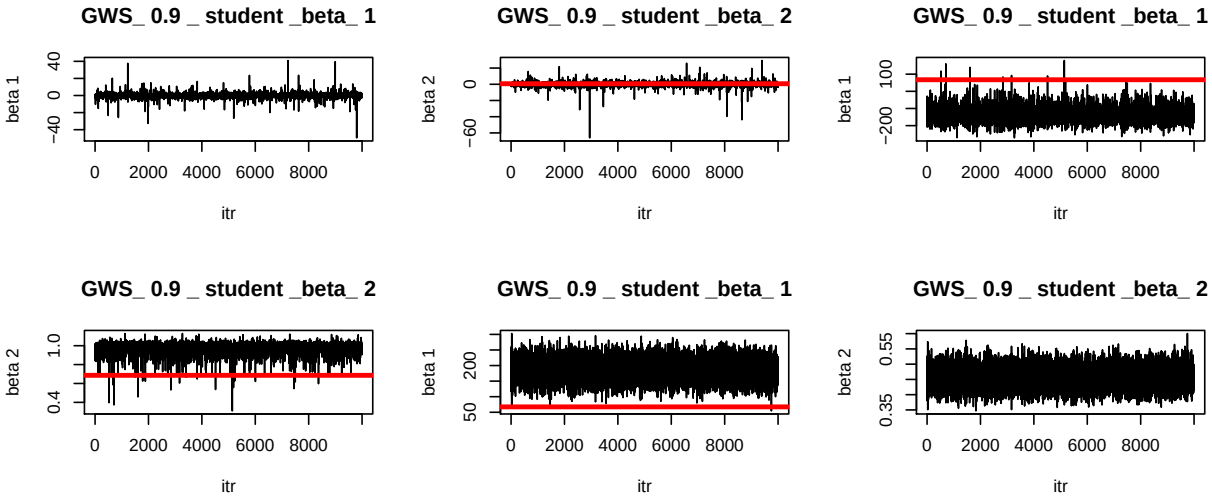
```
par(mfrow=c(2,3))
setwd('C:\\Users\\juyil\\Dropbox\\Study\\GitHub\\Bayesian-gibbs-sampler\\R')
for (p in quantiles){
    rqfit = rq(foodexp ~ income, data = engel, tau = p)
    for (dist_eps in dist_eps_list){
        load(paste0("Engel\\Gibbs with Scale\\","GWS_",p,"_",dist_eps,"_betap.RData"))
        beta_p = list()
        for (i in 1:d){
          beta_p[[i]]=unlist(lapply(beta_p_record, function(x){return(x[i,])}))[(burn_in+1):itr_total]
          plot(beta_p[[i]], type="l", main=paste("GWS_",p,"_",dist_eps,"_beta_",i), xlab="itr", ylab=pas
          abline(h=summary(rqfit)$coefficients[i],col="red",lwd=3)
        }
        new_obs = list(dist_eps, "GWS", as.numeric(p),
                    as.numeric(mcse(beta_p[[1]])$se), as.numeric(length(beta_p[[1]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[1]])), as.numeric(sd(beta_p[[1]])),
                    as.numeric(beta_p[[1]][length(beta_p[[1]])]),
                    as.numeric(mcse(beta_p[[2]])$se), as.numeric(length(beta_p[[2]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[2]])), as.numeric(sd(beta_p[[2]])),
                    as.numeric(beta_p[[2]][length(beta_p[[2]])]))
        simulation1[nrow(simulation1)+1,] = new_obs

#################################################################
        load(paste0("Engel\\Gibbs without Scale\\","GWOS_",p,"_",dist_eps,"_betap.RData"))
        beta_p = list()
        for (i in 1:d){
          beta_p[[i]]=unlist(lapply(beta_p_record, function(x){return(x[i,])}))[(burn_in+1):itr_total]
          plot(beta_p[[i]], type="l", main=paste("GWS_",p,"_",dist_eps,"_beta_",i), xlab="itr", ylab=pas
          abline(h=summary(rqfit)$coefficients[i],col="red",lwd=3)
        }
        new_obs = list(dist_eps, "GWOS", as.numeric(p),
                    as.numeric(mcse(beta_p[[1]])$se), as.numeric(length(beta_p[[1]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[1]])), as.numeric(sd(beta_p[[1]])),
                    as.numeric(beta_p[[1]][length(beta_p[[1]])]),
                    as.numeric(mcse(beta_p[[2]])$se), as.numeric(length(beta_p[[2]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[2]])), as.numeric(sd(beta_p[[2]])),
                    as.numeric(beta_p[[2]][length(beta_p[[2]])]))
        simulation1[nrow(simulation1)+1,] = new_obs

#################################################################
        load(paste0("Engel\\Double Exponential Prior\\","DEP_",p,"_",dist_eps,"_betap.RData"))
        beta_p = list()
        for (i in 1:d){
          beta_p[[i]]=unlist(lapply(beta_p_record, function(x){return(x[i,])}))[(burn_in+1):itr_total]
          plot(beta_p[[i]], type="l", main=paste("GWS_",p,"_",dist_eps,"_beta_",i), xlab="itr", ylab=pas
          abline(h=summary(rqfit)$coefficients[i],col="red",lwd=3)
        }
        new_obs = list(dist_eps, "DEP", as.numeric(p),
                    mcse(beta_p[[1]])$se, length(beta_p[[1]])/effectiveSize(beta_p[[1]]),
                    as.numeric(mean(beta_p[[1]])), as.numeric(sd(beta_p[[1]])),
                    as.numeric(beta_p[[1]][length(beta_p[[1]])]),
                    mcse(beta_p[[2]])$se, length(beta_p[[2]])/effectiveSize(beta_p[[2]]),
                    as.numeric(mean(beta_p[[2]])), as.numeric(sd(beta_p[[2]])),
```

```
                    as.numeric(beta_p[[2]][length(beta_p[[2]])]))
        simulation1[nrow(simulation1)+1,] = new_obs

#####################################################################
        load(paste0("Engel\\Tobit with Scale\\","TWS_",p,"_",dist_eps,"_betap.RData"))
        beta_p = list()
        for (i in 1:d){
          beta_p[[i]]=unlist(lapply(beta_p_record, function(x){return(x[i,])}))[(burn_in+1):itr_total]
          plot(beta_p[[i]], type="l", main=paste("GWS_",p,"_",dist_eps,"_beta_",i), xlab="itr", ylab=pas
          abline(h=summary(rqfit)$coefficients[i],col="red",lwd=3)
        }
        new_obs = list(dist_eps, "TWS", as.numeric(p),
                    as.numeric(mcse(beta_p[[1]])$se), as.numeric(length(beta_p[[1]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[1]])), as.numeric(sd(beta_p[[1]])),
                    as.numeric(beta_p[[1]][length(beta_p[[1]])]),
                    as.numeric(mcse(beta_p[[2]])$se), as.numeric(length(beta_p[[2]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[2]])), as.numeric(sd(beta_p[[2]])),
                    as.numeric(beta_p[[2]][length(beta_p[[2]])]))
        simulation1[nrow(simulation1)+1,] = new_obs

#####################################################################
        load(paste0("Engel\\Tobit without Scale\\","TWOS_",p,"_",dist_eps,"_betap.RData"))
        beta_p = list()
        for (i in 1:d){
          beta_p[[i]]=unlist(lapply(beta_p_record, function(x){return(x[i,])}))[(burn_in+1):itr_total]
          plot(beta_p[[i]], type="l", main=paste("GWS_",p,"_",dist_eps,"_beta_",i), xlab="itr", ylab=pas
          abline(h=summary(rqfit)$coefficients[i],col="red",lwd=3)
        }
        new_obs = list(dist_eps, "TWOS", as.numeric(p),
                    as.numeric(mcse(beta_p[[1]])$se), as.numeric(length(beta_p[[1]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[1]])), as.numeric(sd(beta_p[[1]])),
                    as.numeric(beta_p[[1]][length(beta_p[[1]])]),
                    as.numeric(mcse(beta_p[[2]])$se), as.numeric(length(beta_p[[2]])/effectiveSize(beta_
                    as.numeric(mean(beta_p[[2]])), as.numeric(sd(beta_p[[2]])),
                    as.numeric(beta_p[[2]][length(beta_p[[2]])]))
        simulation1[nrow(simulation1)+1,] = new_obs

    }
}
```


GWS_ 0.1 _ stdN _beta_ 1
GWS_ 0.1 _ stdN _beta_ 2
GWS_ 0.1 _ stdN _beta_ 1
GWS_ 0.1 _ stdN _beta_ 2
GWS_ 0.1 _ stdN _beta_ 1
GWS_ 0.1 _ stdN _beta_ 2
GWS_ 0.1 _ stdN _beta_ 1
GWS_ 0.1 _ stdN _beta_ 2
GWS_ 0.1 _ stdN _beta_ 1
GWS_ 0.1 _ stdN _beta_ 2
GWS_ 0.1 _ student _beta_ 1
GWS_ 0.1 _ student _beta_ 2
GWS_ 0.1 _ student _beta_ 1
GWS_ 0.1 _ student _beta_ 2
GWS_ 0.1 _ student _beta_ 1
GWS_ 0.1 _ student _beta_ 2
beta 1
beta 2
itr
0
2000
4000
6000
8000


GWS_ 0.1 _ student _beta_ 1
GWS_ 0.1 _ student _beta_ 2
GWS_ 0.5 _ stdN _beta_ 1
GWS_ 0.5 _ stdN _beta_ 2
GWS_ 0.5 _ stdN _beta_ 1
GWS_ 0.5 _ stdN _beta_ 2
GWS_ 0.5 _ stdN _beta_ 1
GWS_ 0.5 _ stdN _beta_ 2
GWS_ 0.5 _ stdN _beta_ 1
GWS_ 0.5 _ stdN _beta_ 2
GWS_ 0.5 _ stdN _beta_ 1
GWS_ 0.5 _ stdN _beta_ 2
GWS_ 0.5 _ student _beta_ 1
GWS_ 0.5 _ student _beta_ 2
GWS_ 0.5 _ student _beta_ 1
GWS_ 0.5 _ student _beta_ 2
GWS_ 0.5 _ student _beta_ 1
GWS_ 0.5 _ student _beta_ 2
beta 1
beta 2
itr
0
2000
4000
6000
8000


GWS_ 0.5 _ student _beta_ 1
GWS_ 0.5 _ student _beta_ 2
GWS_ 0.5 _ student _beta_ 1
GWS_ 0.5 _ student _beta_ 2
GWS_ 0.9 _ stdN _beta_ 1
GWS_ 0.9 _ stdN _beta_ 2
GWS_ 0.9 _ stdN _beta_ 1
GWS_ 0.9 _ stdN _beta_ 2
GWS_ 0.9 _ stdN _beta_ 1
GWS_ 0.9 _ stdN _beta_ 2
GWS_ 0.9 _ stdN _beta_ 1
GWS_ 0.9 _ stdN _beta_ 2
GWS_ 0.9 _ student _beta_ 1
GWS_ 0.9 _ student _beta_ 2
GWS_ 0.9 _ student _beta_ 1
GWS_ 0.9 _ student _beta_ 2
beta 1
beta 2
itr
0
2000
4000
6000
8000


GWS_ 0.9 _ student _beta_ 1
beta 1
−40
0
40
0
2000
4000
6000
8000
itr
GWS_ 0.9 _ student _beta_ 2
beta 2
−60
0
0
2000
4000
6000
8000
itr
GWS_ 0.9 _ student _beta_ 1
beta 1
−200
100
0
2000
4000
6000
8000
itr
GWS_ 0.9 _ student _beta_ 2
beta 2
0.4
1.0
0
2000
4000
6000
8000
itr
GWS_ 0.9 _ student _beta_ 1
beta 1
50
200
0
2000
4000
6000
8000
itr
GWS_ 0.9 _ student _beta_ 2
beta 2
0.35
0.55
0
2000
4000
6000
8000
itr

Clean

```
ord = c("stdN","heteroN","student")
simulation1$Distribution_Eps = factor(simulation1$Distribution_Eps,levels=ord)
simulation1 = simulation1[order(simulation1$Distribution_Eps),]
is.num <- sapply(simulation1, is.numeric)
simulation1[is.num] <- lapply(simulation1[is.num], round, 4)
simulation1
```

```
##    Distribution_Eps Method   p beta1_MCSE beta1_IF  beta1_mu beta1_sd
## 1              stdN    GWS 0.1     0.0024   1.0000  205.6222   0.2436
## 2              stdN   GWOS 0.1     0.1389   0.2230  183.1656  34.3504
## 3              stdN    DEP 0.1     0.0213   1.0000   -0.0078   2.1268
## 4              stdN    TWS 0.1     0.0024   1.0000  205.6222   0.2436
## 5              stdN   TWOS 0.1     0.1389   0.2230  183.1656  34.3504
## 11             stdN    GWS 0.5     0.0069   0.2313   61.5778   1.4369
## 12             stdN   GWOS 0.5     0.1430   0.2834  181.3803  27.0758
## 13             stdN    DEP 0.5     0.0205   1.0000    0.0075   2.0461
## 14             stdN    TWS 0.5     0.0069   0.2313   61.5778   1.4369
## 15             stdN   TWOS 0.5     0.1430   0.2834  181.3803  27.0758
## 21             stdN    GWS 0.9     0.3113   0.7344 -121.2961  35.7309
## 22             stdN   GWOS 0.9     0.1834   0.2138  187.8984  33.8837
## 23             stdN    DEP 0.9     0.0209   1.0000    0.0064   2.0857
## 24             stdN    TWS 0.9     0.3113   0.7344 -121.2961  35.7309
## 25             stdN   TWOS 0.9     0.1834   0.2138  187.8984  33.8837
## 6           student    GWS 0.1     0.0026   1.0727  205.6145   0.2586
## 7           student   GWOS 0.1     0.1554   0.2153  183.4250  33.9458
## 8           student    DEP 0.1     0.0196   1.0000    0.0091   1.9588
## 9           student    TWS 0.1     0.0026   1.0727  205.6145   0.2586
## 10          student   TWOS 0.1     0.1554   0.2153  183.4250  33.9458
## 16          student    GWS 0.5     0.0071   0.2223   61.5517   1.4918
## 17          student   GWOS 0.5     0.1349   0.2824  181.2703  26.9501
## 18          student    DEP 0.5     0.0296   1.0000   -0.0462   2.9588
## 19          student    TWS 0.5     0.0071   0.2223   61.5517   1.4918
## 20          student   TWOS 0.5     0.1349   0.2824  181.2703  26.9501
## 26          student    GWS 0.9     0.3201   0.8315 -119.6984  36.4621
## 27          student   GWOS 0.9     0.1278   0.2273  188.0345  33.5837
## 28          student    DEP 0.9     0.0213   1.0000   -0.0043   2.1340
## 29          student    TWS 0.9     0.3201   0.8315 -119.6984  36.4621
## 30          student   TWOS 0.9     0.1278   0.2273  188.0345  33.5837
##    beta1_last beta2_MCSE beta2_IF beta2_mu beta2_sd beta2_last
## 1    205.4694     0.0000   0.9454   0.1196   0.0005     0.1199
## 2    184.2547     0.0001   0.2288   0.4547   0.0349     0.4493
## 3      0.0579     0.0215   1.0765  -0.0046   2.0398    -0.6097
## 4    205.4694     0.0000   0.9454   0.1196   0.0005     0.1199
## 5    184.2547     0.0001   0.2288   0.4547   0.0349     0.4493
## 11    59.6592     0.0000   0.5973   0.5417   0.0012     0.5427
## 12   149.7276     0.0001   0.2880   0.4567   0.0279     0.4900
## 13    -2.0986     0.0259   1.0000   0.0380   2.5898    -0.5499
## 14    59.6592     0.0000   0.5973   0.5417   0.0012     0.5427
## 15   149.7276     0.0001   0.2880   0.4567   0.0279     0.4900
## 21  -105.4173     0.0004   1.0073   0.9711   0.0441     0.9383
## 22   207.5443     0.0002   0.2229   0.4535   0.0344     0.4319
## 23    -1.9056     0.0268   1.0000  -0.0065   2.6815     4.1610
```

```
## 24  -105.4173     0.0004   1.0073   0.9711   0.0441     0.9383
## 25   207.5443     0.0002   0.2229   0.4535   0.0344     0.4319
## 6    205.7859     0.0000   0.9956   0.1196   0.0006     0.1194
## 7    143.9234     0.0002   0.2239   0.4543   0.0343     0.4943
## 8     -0.0860     0.0210   1.0000  -0.0152   2.1021     0.8898
## 9    205.7859     0.0000   0.9956   0.1196   0.0006     0.1194
## 10   143.9234     0.0002   0.2239   0.4543   0.0343     0.4943
## 16    60.4416     0.0000   0.5559   0.5418   0.0013     0.5431
## 17   187.3763     0.0001   0.2894   0.4568   0.0277     0.4454
## 18    -0.5662     0.0197   1.0000  -0.0339   1.9749    -0.0959
## 19    60.4416     0.0000   0.5559   0.5418   0.0013     0.5431
## 20   187.3763     0.0001   0.2894   0.4568   0.0277     0.4454
## 26   -91.4538     0.0005   1.0557   0.9687   0.0456     0.9629
## 27   159.1309     0.0001   0.2349   0.4533   0.0342     0.4819
## 28     0.3276     0.0207   1.0000  -0.0122   2.0749    -1.6529
## 29   -91.4538     0.0005   1.0557   0.9687   0.0456     0.9629
## 30   159.1309     0.0001   0.2349   0.4533   0.0342     0.4819
```

quantreg

```
rqfit <- rq(foodexp ~ income, data = engel, tau = quantiles)
summary(rqfit)
```

```
##
## Call: rq(formula = foodexp ~ income, tau = quantiles, data = engel)
##
## tau: [1] 0.1
##
## Coefficients:
##             coefficients lower bd  upper bd
## (Intercept) 110.14157     79.88753 146.18875
## income        0.40177      0.34210   0.45079
##
## Call: rq(formula = foodexp ~ income, tau = quantiles, data = engel)
##
## tau: [1] 0.5
##
## Coefficients:
##             coefficients lower bd  upper bd
## (Intercept)  81.48225     53.25915 114.01156
## income        0.56018      0.48702   0.60199
##
## Call: rq(formula = foodexp ~ income, tau = quantiles, data = engel)
##
## tau: [1] 0.9
##
## Coefficients:
##             coefficients lower bd  upper bd
## (Intercept)  67.35087     37.11802 103.17399
## income        0.68630      0.64937   0.74223
```