Analysis of HPC results

Willem Vervoort, Michaela Dolk & Floris van Ogtrop

2020-02-04

```
# root dir
knitr::opts_knit$set(root.dir = #"C:/Users/rver4657/ownCloud/Virtual Experiments/VirtExp")
                        "D:/cloudstor/Virtual Experiments/VirtExp")
knitr::opts_chunk$set(echo = TRUE)
# LOAD REQUIRED PACKAGES # #####
library(pander)
library(tidyverse)
library(xts)
library(reshape2)
library(Rcpp)
library(hydromad)
```

This rmarkdown document and the resulting pdf are stored on github. All directories (apart from the root working directory) refer to the directories in this repository.

Introduction

This document is related to the manuscript "Disentangling climate change trends in Australian streamflow" (vervoort et al.). This is the sixth part of the series that analyses the results from the GR4J and SimHyd model fitting on the High Performance computer Artemis at the University of Sydney. The scripts related to the high performance computing and the HPC scripts are stored in the Rcode/HPC folder.

In particular, this part extracts the best parameters of the modelling, plots the performance distributions and extracts the residuals to be analysed in a further script using Mann Kendall (this is separated because this takes quite long to run). Finally a comparison between the non-parametric epsilon (sensitivity) and the model based epsilon is plotted for both gridded and non-gridded rainfall.

This is ultimately related to *Figure 3, Figure 4 and Figure 9* in the manuscript. *Figure 9* is generated in a different script, as this is the Mann Kendall analysis of the residuals of the models.

To recap, we have 4 different High Performance Computing results for each of the 13 catchments. In each of these the two models (GR4J and SimHyd) were fitted to 40 years of climate data (1970 - 2010). However, there are results for both station rainfall and gridded rainfall to make sure there is no difference between these. As we have seen in the non-parametric epsilon calculation, there is no major difference between the two rainfall data sets, and it is not necessarily clear, which one of these is a better data set.

The model fitting is based on the shuffled complex evolution optimisation in Hydromad as described in more detail in the paper. Essentially the model was fitted 10 times using the `FitBySCE()` function in the Hydromad package.

The HPC results are not stored on Github, as the files are too large, but are stored in the Cloudstor data directory.

1. Load basic data and define storage

This loads all the basic climate data and the catchment characteristics.
It also compiles the SimHyd model code.

```r
load("data/DailyDataIncludingGridded.Rdata")
load("data/ClimCh_project_MD.Rdata")
# correct the column name of maxT in GridRainAllDataout
colnames(GridRainAllDataout)[5] <- "MaxT"
# change to tibble
GridAlldata <- as_tibble(cbind(GridRainAllDataout,
                               Date=rep(time(flow_zoo),nrow(Stations))))
# now use spread
GridAlldata_wide <- spread(GridAlldata[,c("Station","gridRain","Date")],
                           key=Station, value=gridRain)
Gridrain_zoo <- zoo(GridAlldata_wide[,2:14],order.by=time(flow_zoo))

# compile SimHyd
root.dir = #"C:/Users/rver4657/ownCloud/Virtual Experiments/VirtExp"
  "D:/cloudstor/Virtual Experiments/VirtExp"
rcode_dir <- paste(root.dir,"RCode/Simhyd",sep="/")
source(paste(rcode_dir,"Simhyd.r",sep="/"))
```

Define the beginning and end date for the modelling. We are now using the whole series for the calibration

```r
start.date <- as.Date("1970-01-01")
end.date <- as.Date("2010-12-31")
```

As a first step define storage for the results of the modelling. These will get renamed for each of the individual modelling results

```r
sum_Res <- list()
Chiew_Res <- list()

mod_Res <- list()

Chiew <- data.frame(station=character(length=10),eta_p=numeric(length=10),
                    eta_e=numeric(length=10),pvalue_eta_p=numeric(length=10),
                    pvalue_eta_e=numeric(length=10))

Results <- data.frame(station=character(length=10),
                      Mod.r.sq=numeric(length=10),
                      Mod.bias=numeric(length=10))

Residuals <- list()

# some other auxillary data frames
pred_results <- data.frame(Pmin15ET0=numeric(length=nrow(flow_zoo)),
                           Pmin10ET0=numeric(length=nrow(flow_zoo)),
                           P0ET0=numeric(length=nrow(flow_zoo)),
                           Pplus10ET0=numeric(length=nrow(flow_zoo)),
                           Pmin15ETplus5=numeric(length=nrow(flow_zoo)),
                           Pmin10ETplus5=numeric(length=nrow(flow_zoo)),
                           P0ETplus5=numeric(length=nrow(flow_zoo)),
                           Pplus10ETplus5=numeric(length=nrow(flow_zoo)),
                           Pmin15ETplus10=numeric(length=nrow(flow_zoo)),
                           Pmin10ETplus10=numeric(length=nrow(flow_zoo)),
```

```
                                POETplus10=numeric(length=nrow(flow_zoo)),
                                Pplus10ETplus10=numeric(length=nrow(flow_zoo)))
```

2. GR4J model results with station rainfall

Extract the modelling results, rerun the model, do the Chiew (2006) amplification analysis and write away the results.

```
# find the list of files with GR4J results
filelist <- dir("../Projectdata/HPCResults", pattern = "GR4JCalibOutput")

for (i in seq_along(filelist)) {
  #i <- 1
  # load the rainfall, ET and flow data
  pred_data <- window(merge(flow_zoo[,i], rain_zoo[,i], maxT_zoo[,i]),
                      start=start.date, end=end.date)
  colnames(pred_data) <- c("Q","P","E")
  # # another storage data frame for the residuals
  resid_out <- data.frame(matrix(0,ncol=10,nrow=nrow(pred_data)))
  # load the relevant output
  load(paste("../Projectdata/HPCResults/",
             filelist[grep(Stations[i,1],filelist)],sep=""))
  # extract the model and update with the parameters
  Mod <- Output$mod

  mod_Res[[i]] <- Output$Store
  Chiew[,1] <- Stations[i,1]
  Results[,1] <- Stations[i,1]


  # run through all iterations
  for (j in 1:(nrow(Output$Store))) {
    # testing
    #j <- 1
    # update the model with the final fitted parameters, for each iteration
    Mod <- update(Mod, x1=Output$Store[j,8],x2=Output$Store[j,5],
                  x3=Output$Store[j,6],x4=Output$Store[j,7],
                  etmult=Output$Store[j,9],
                  return_state=F)


    # now predict the model output and use pred_data
    pred_mod <- predict(Mod,newdata=pred_data, all=T,na.rm=F)

    # store the residuals
    resid_out[,j] <- pred_data$Q-pred_mod

    Results[j,2:3] <- c(summary(Mod)$r.squared,summary(Mod)$rel.bias)

    # Now run the Chiew 2006 simulations on all the data
    mu <- cbind(rep(c(-15,-10,0,10),3),c(rep(0,4),rep(5,4),rep(10,4)))
    # Create the precipitation and ET data variations
```

```
    # station data
    RAIN <- rain_zoo
    test <- list()
    for (k in 1:nrow(mu)) {
      temp <- as.data.frame(cbind((1+mu[k,1]/100)*RAIN[,i],
                          (1+mu[k,2]/100)*maxT_zoo[,i]))
      test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                    list(year=format(time(flow_zoo),"%Y")),sum,na.rm=T)))
      test[[k]] <- test[[k]][,-3]
    }
    clim_adj <- do.call(rbind,test)

    # now run the different pred results
    for (k in 1:ncol(pred_results)) {
      # run the model over all data
      pred_data2 <- window(merge(flow_zoo[,i],
                              (1+mu[k,1]/100)*RAIN[,i],
                              (1+mu[k,2]/100)*maxT_zoo[,i]))
      colnames(pred_data2) <- c("Q","P","E")

      pred_results[,k] <- predict(Mod,newdata=pred_data2, all=T,na.rm=F)
    }
    # summarise the data annually
    pred_ann <- apply(pred_results,2,
             function(x) aggregate(x,list(year=format(time(flow_zoo),"%Y")),
                                         sum,na.rm=T))
    ann_flow <- rep(pred_ann[[1]][,2],6)
    pred_t <- do.call(rbind,pred_ann)
    # Now add the ET and precipitation data
    pred_ann <- data.frame(pred_t,rain=clim_adj[,2],maxT=clim_adj[,3])
    # summarise base rain and temp
    ann_rain <- rep(aggregate(RAIN[,i],list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo[,i],
                          list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)

    # Now calculate the difference
    pred_diff <- pred_ann
    pred_diff[,2] <- pred_diff[,2] - ann_flow
    pred_diff[,3] <- pred_diff[,3] - ann_rain
    pred_diff[,4] <- pred_diff[,4] - ann_maxT
    # Now fit a linear model (least squares (Chiew, 2006))
    fit <- lm(x~rain + maxT,data=pred_diff)
    # store the results
    Chiew[j,2:5] <- c(coef(fit)[2:3],summary(fit)$coefficients[2:3,4])


  }
  Chiew_Res[[i]] <- Chiew
  sum_Res[[i]] <- Results
  Residuals[[i]] <- resid_out
}
```

We can now temporary write away the results and the residuals and make some initial plots. Further plots will be generated later when comparing to the non-parametric epsilon (ϵ).

```
# write away the results
OutputTrends <- do.call(rbind,sum_Res)
write.csv(OutputTrends,
          file="../ProjectData/GR4JHPC_modelperformance.csv",
          row.names=F)

OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,file="../ProjectData/GR4JHPC_ChiewAnalysis.csv",
          row.names=F)
pander(aggregate(OutputChiew[,2:5],list(Station=OutputChiew$station),
                 mean),
       caption="Results GR4J epsilon fit with significance")
```

Table 1: Results GR4J epsilon fit with significance

Station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COCH	0.6663	-0.008773	4.827e-183	0.3177
COEN	0.7716	-0.02077	1.568e-209	3.916e-08
CORA	0.6879	-0.02098	3.674e-136	1.242e-05
COTT	0.4231	-0.01879	5.528e-96	0.0005491
DOMB	0.5284	-0.0336	1.074e-245	5.36e-23
ELIZ	0.6198	-0.03058	7.075e-144	7.467e-07
HELL	1.384	-0.1187	1.203e-254	3.98e-39
MURR	0.8892	-0.04354	5.086e-233	2.133e-24
NIVE	1.107	-0.06	2.71e-246	1.139e-09
RUTH	0.745	-0.0247	2.601e-101	9.224e-05
SCOT	0.4178	-0.01797	2.648e-173	1.499e-13
SOUT	1.289	-0.1166	3.829e-251	4.26e-40
YARR	0.07974	-0.004315	1.012e-111	3.152e-07

```
save(Residuals,file="../ProjectData/residuals/GR4JHPCresiduals.Rdata")

OutputMod_GR4J <- do.call(rbind,mod_Res)
save(OutputMod_GR4J,file="../ProjectData/GR4JHPCModelResults.Rdata")

OutputMod_GR4J <- OutputMod_GR4J[,c(1:4,ncol(OutputMod_GR4J))]

OutputMod_GR4J$model <- "GR4J Station Rainfall Data"

p <- ggplot(OutputMod_GR4J,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
                      size=5,aes(col=rel.bias))
print(p)
```

3. GR4J model results with gridded rainfall

Extract the modelling results

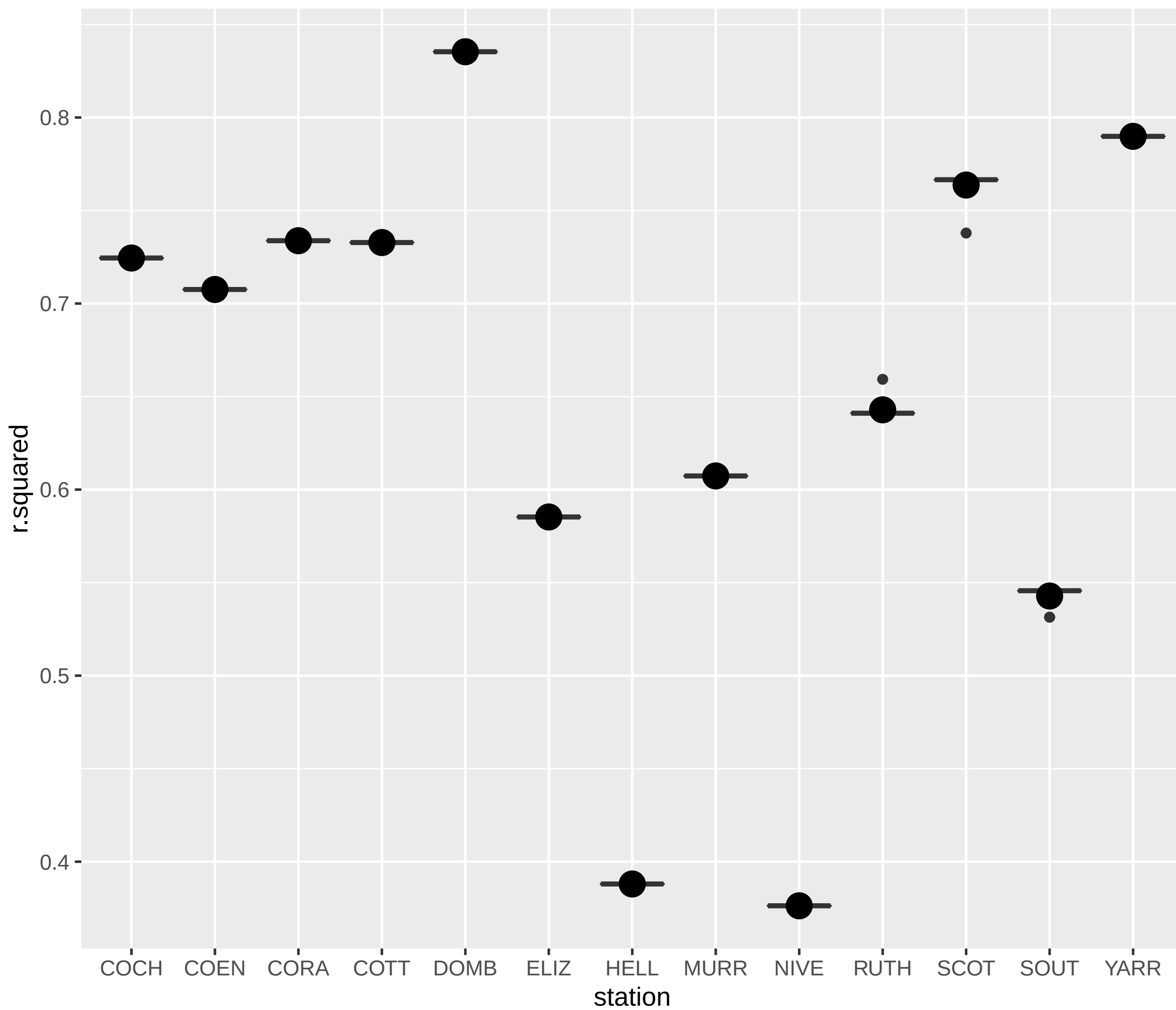

Figure 1: Calibration results for the GR4J model across 10 iterations using station rainfall data.

```r
# find the list of files with GR4J results
filelist <- dir("../Projectdata/HPCResults", pattern = "GR4JGridCalibOutput")

for (i in seq_along(filelist)) {
  # load the rainfall, ET and flow data
  pred_data <- window(merge(flow_zoo[,i], Gridrain_zoo[,i], maxT_zoo[,i]),
                     start=start.date, end=end.date)
  colnames(pred_data) <- c("Q","P","E")
  # # another storage data frame for the residuals
  resid_out <- data.frame(matrix(0,ncol=10,nrow=nrow(pred_data)))
  # load the relevant output
  load(paste("../Projectdata/HPCResults/",
             filelist[grep(Stations[i,1],filelist)],sep=""))
  # extract the model and update with the parameters
  Mod <- Output$mod

  mod_Res[[i]] <- Output$Store
  Chiew[,1] <- Stations[i,1]
  Results[,1] <- Stations[i,1]


  # run through all iterations
  for (j in 1:(nrow(Output$Store))) {
    # testing
    #j <- 1
    # update the model with the final fitted parameters, for each iteration
    Mod <- update(Mod, x1=Output$Store[j,8],x2=Output$Store[j,5],
                  x3=Output$Store[j,6],x4=Output$Store[j,7],
                  etmult=Output$Store[j,9],
                  return_state=F)


    # now predict the model output and use pred_data
    pred_mod <- predict(Mod,newdata=pred_data, all=T,na.rm=F)

    # store the residuals
    resid_out[,j] <- pred_data$Q-pred_mod

    Results[j,2:3] <- c(summary(Mod)$r.squared,summary(Mod)$rel.bias)

    # Now run the Chiew 2006 simulations on all the data
    mu <- cbind(rep(c(-15,-10,0,10),3),c(rep(0,4),rep(5,4),rep(10,4)))
    # Create the precipitation and ET data variations
    # station data
    RAIN <- Gridrain_zoo
    test <- list()
    for (k in 1:nrow(mu)) {
      temp <- as.data.frame(cbind((1+mu[k,1]/100)*RAIN[,i],
                            (1+mu[k,2]/100)*maxT_zoo[,i]))
      test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                      list(year=format(time(flow_zoo),"%Y")),sum,na.rm=T)))
      test[[k]] <- test[[k]][,-3]
    }
```

```r
    clim_adj <- do.call(rbind,test)

    # now run the different pred results
    for (k in 1:ncol(pred_results)) {
      # run the model over all data
      pred_data2 <- window(merge(flow_zoo[,i],
                                 (1+mu[k,1]/100)*RAIN[,i],
                                 (1+mu[k,2]/100)*maxT_zoo[,i]))
      colnames(pred_data2) <- c("Q","P","E")

      pred_results[,k] <- predict(Mod,newdata=pred_data2, all=T,na.rm=F)
    }
    # summarise the data annually
    pred_ann <- apply(pred_results,2,
             function(x) aggregate(x,list(year=format(time(flow_zoo),"%Y")),
                                       sum,na.rm=T))
    ann_flow <- rep(pred_ann[[1]][,2],6)
    pred_t <- do.call(rbind,pred_ann)
    # Now add the ET and precipitation data
    pred_ann <- data.frame(pred_t,rain=clim_adj[,2],maxT=clim_adj[,3])
    # summarise base rain and temp
    ann_rain <- rep(aggregate(RAIN[,i],list(year=format(time(flow_zoo),"%Y")),
                              sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo[,i],
                              list(year=format(time(flow_zoo),"%Y")),
                              sum,na.rm=T),6)

    # Now calculate the difference
    pred_diff <- pred_ann
    pred_diff[,2] <- pred_diff[,2] - ann_flow
    pred_diff[,3] <- pred_diff[,3] - ann_rain
    pred_diff[,4] <- pred_diff[,4] - ann_maxT
    # Now fit a linear model (least squares (Chiew, 2006))
    fit <- lm(x~rain + maxT,data=pred_diff)
    # store the results
    Chiew[j,2:5] <- c(coef(fit)[2:3],summary(fit)$coefficients[2:3,4])


  }
  Chiew_Res[[i]] <- Chiew
  sum_Res[[i]] <- Results
  Residuals[[i]] <- resid_out
}
```

We can now temporary write away the results and the residuals and make some initial plots. Further plots will be generated later when comparing to the non-parametric epsilon (ϵ).

```r
# write away the results
OutputTrends <- do.call(rbind,sum_Res)
write.csv(OutputTrends,
          file="../ProjectData/GR4JGridHPC_modelperformance.csv",
          row.names=F)
```

```
OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,file="../ProjectData/GR4JGridHPC_ChiewAnalysis.csv",
          row.names=F)
pander(aggregate(OutputChiew[,2:5],list(Station=OutputChiew$station),
                 mean),
       caption="Results GR4JGrid epsilon fit with significance")
```

Table 2: Results GR4JGrid epsilon fit with significance

Station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COCH	0.9266	-0.03637	7.914e-296	9.149e-36
COEN	0.8803	-0.02664	6.893e-231	1.071e-08
CORA	0.7226	-0.01728	1.453e-154	1.326e-05
COTT	0.9328	-0.08743	2.988e-221	3.677e-12
DOMB	0.6165	-0.0324	3.415e-208	6.215e-18
ELIZ	0.3101	-0.01586	6.743e-101	1.464e-11
HELL	0.9383	-0.07582	1.143e-237	4.782e-19
MURR	0.6653	-0.04132	2.645e-124	1.543e-14
NIVE	1.253	-0.1106	1.976e-323	2.607e-42
RUTH	1.032	-0.04624	3.219e-230	6.368e-12
SCOT	0.6662	-0.05352	2.652e-276	7.432e-28
SOUT	1.31	-0.07536	3.727e-292	2.98e-58
YARR	0.08925	-0.007062	1.063e-113	8.781e-12

```
save(Residuals,file="../ProjectData/residuals/GR4JGridHPCresiduals.Rdata")

OutputMod_GridGR4J <- do.call(rbind,mod_Res)
OutputMod_GridGR4J$station <- OutputMod_GR4J$station
save(OutputMod_GridGR4J,file="../ProjectData/GR4JgridHPCModelResults.Rdata")

OutputMod_GridGR4J <- OutputMod_GridGR4J[,c(1:4,ncol(OutputMod_GridGR4J))]
OutputMod_GridGR4J$model <- "GR4J Gridded Rainfall Data"

p <- ggplot(OutputMod_GridGR4J,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
                      size=5,aes(col=rel.bias))
print(p)
```

4. SimHyd model results with station rainfall

Extract the modelling results from the SimHyd model, rerun the model, do the Chiew (2006) amplification analysis and write away the results.

```
# find the list of files with SimHyd results
filelist <- dir("../Projectdata/HPCResults", pattern = "SimhydCalibOutput")
filelist <- filelist[-grep("gridSimhydCalibOutput",filelist)]

for (i in seq_along(Stations[,1])) {
  # load the rainfall, ET and flow data
```

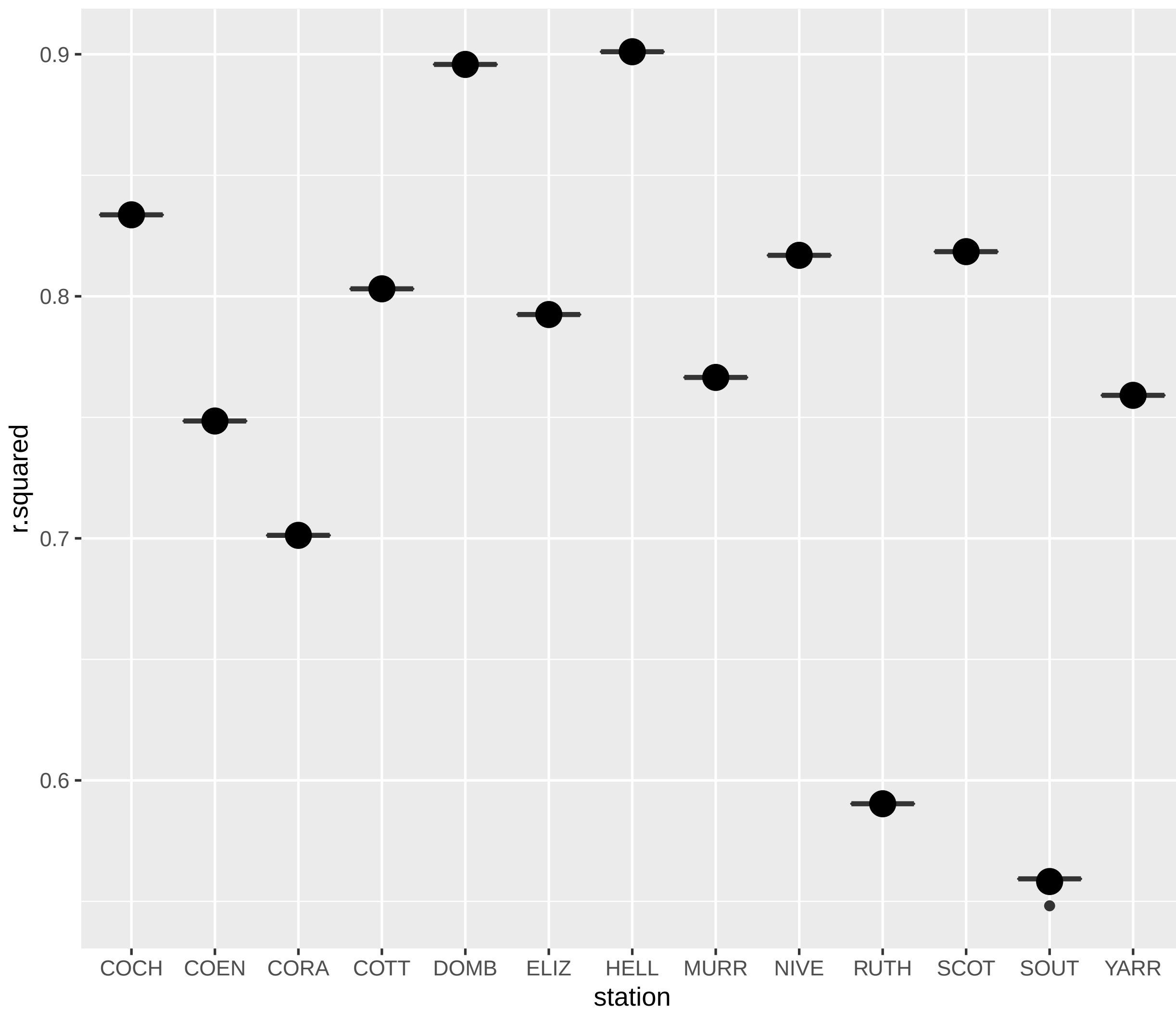

Figure 2: Calibration results for the GR4J model across 10 iterations with gridded rainfall

```r
pred_data <- window(merge(flow_zoo[,i], rain_zoo[,i], maxT_zoo[,i]),
                    start=start.date, end=end.date)
colnames(pred_data) <- c("Q","P","E")
# # another storage data frame for the residuals
resid_out <- data.frame(matrix(0,ncol=10,nrow=nrow(pred_data)))
# load the relevant output
load(paste("../Projectdata/HPCResults/",
          filelist[grep(paste(Stations[i,1],"SimhydCalibOutput",sep=""),
                        filelist)],sep=""))
# extract the model and update with the parameters
Mod <- Output$mod

mod_Res[[i]] <- Output$Store
Chiew[,1] <- Stations[i,1]
Results[,1] <- Stations[i,1]


# run through all iterations
for (j in 1:(nrow(Output$Store))) {
  # testing
  #j <- 1
  # update the model with the final fitted parameters, for each iteration
  Mod <- update(Mod, INSC=Output$Store[j,7],COEFF=Output$Store[j,8],
                  SQ=Output$Store[j,9],SMSC=Output$Store[j,10],
                  SUB=Output$Store[j,11],CRAK=Output$Store[j,12],
                  K=Output$Store[j,13],
                 etmult=Output$Store[j,14], DELAY=Output$Store[j,5],
                 X_m = Output$Store[j,6],
                return_state=F)


  # now predict the model output and use pred_data
  pred_mod <- predict(Mod,newdata=pred_data, all=T,na.rm=F)

  # store the residuals
  resid_out[,j] <- pred_data$Q-pred_mod

  Results[j,2:3] <- c(summary(Mod)$r.squared,summary(Mod)$rel.bias)

  # Now run the Chiew 2006 simulations on all the data
  mu <- cbind(rep(c(-15,-10,0,10),3),c(rep(0,4),rep(5,4),rep(10,4)))
  # Create the precipitation and ET data variations
  # station data
  RAIN <- rain_zoo
  test <- list()
  for (k in 1:nrow(mu)) {
    temp <- as.data.frame(cbind((1+mu[k,1]/100)*RAIN[,i],
                          (1+mu[k,2]/100)*maxT_zoo[,i]))
    test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                    list(year=format(time(flow_zoo),"%Y")),sum,na.rm=T)))
    test[[k]] <- test[[k]][,-3]
  }
  clim_adj <- do.call(rbind,test)
```

```
    # now run the different pred results
    for (k in 1:ncol(pred_results)) {
      # run the model over all data
      pred_data2 <- window(merge(flow_zoo[,i],
                                 (1+mu[k,1]/100)*RAIN[,i],
                                 (1+mu[k,2]/100)*maxT_zoo[,i]))
      colnames(pred_data2) <- c("Q","P","E")

      pred_results[,k] <- predict(Mod,newdata=pred_data2, all=T,na.rm=F)
    }
    # summarise the data annually
    pred_ann <- apply(pred_results,2,
              function(x) aggregate(x,list(year=format(time(flow_zoo),"%Y")),
                                    sum,na.rm=T))
    ann_flow <- rep(pred_ann[[1]][,2],6)
    pred_t <- do.call(rbind,pred_ann)
    # Now add the ET and precipitation data
    pred_ann <- data.frame(pred_t,rain=clim_adj[,2],maxT=clim_adj[,3])
    # summarise base rain and temp
    ann_rain <- rep(aggregate(RAIN[,i],list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo[,i],
                          list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)

    # Now calculate the difference
    pred_diff <- pred_ann
    pred_diff[,2] <- pred_diff[,2] - ann_flow
    pred_diff[,3] <- pred_diff[,3] - ann_rain
    pred_diff[,4] <- pred_diff[,4] - ann_maxT
    # Now fit a linear model (least squares (Chiew, 2006))
    fit <- lm(x~rain + maxT,data=pred_diff)
    # store the results
    Chiew[j,2:5] <- c(coef(fit)[2:3],summary(fit)$coefficients[2:3,4])


  }
  Chiew_Res[[i]] <- Chiew
  sum_Res[[i]] <- Results
  Residuals[[i]] <- resid_out
}
```

We can now temporary write away the results and the residuals and make some initial plots. Further plots will be generated later when comparing to the non-parametric epsilon (ϵ).

```
# write away the results
OutputTrends <- do.call(rbind,sum_Res)
write.csv(OutputTrends,
          file="../ProjectData/SimHydHPC_modelperformance.csv",
          row.names=F)

OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,file="../ProjectData/SimHydHPC_ChiewAnalysis.csv",
```

```
        row.names=F)
pander(aggregate(OutputChiew[,2:5],list(Station=OutputChiew$station),
                 mean),
       caption="Results SimHyd epsilon fit with significance")
```

Table 3: Results SimHyd epsilon fit with significance

Station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COCH	0.8155	-0.004441	7.187e-196	0.667
COEN	0.8741	-0.002738	2.229e-212	0.6631
CORA	-24.99	903.5	0.1157	0.5841
COTT	0.7543	-0.006282	1.287e-100	0.5227
DOMB	0.9136	-0.01482	2.816e-307	0.2901
ELIZ	0.8913	-0.0004408	8.569e-166	0.9541
HELL	0.8002	-0.007807	3.645e-239	0.3348
MURR	0.8747	-0.0006252	0.001443	0.7676
NIVE	0.873	-0.00468	7.671e-218	0.719
RUTH	0.832	-0.006284	1.046e-132	0.3661
SCOT	0.7944	-0.01498	9.523e-249	0.113
SOUT	0.6819	0.1928	0.000125	0.3262
YARR	0.9521	-0.01195	9.824e-294	0.3324

```
save(Residuals,file="../ProjectData/residuals/SimHydHPCresiduals.Rdata")

OutputMod_SimHyd <- do.call(rbind,mod_Res)
save(OutputMod_SimHyd,file="../ProjectData/SimHydHPCModelResults.Rdata")

OutputMod_SimHyd <- OutputMod_SimHyd[,c(1:4,ncol(OutputMod_SimHyd))]
OutputMod_SimHyd$model <- "SimHyd Station Rainfall Data"

p <- ggplot(OutputMod_SimHyd,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
                      size=5,aes(col=rel.bias))
print(p)
```

5. SimHyd model results with gridded rainfall

Extract the modelling results

```
# find the list of files with GridSimHyd results
filelist <- dir("../Projectdata/HPCResults",
                pattern = "gridSimhydCalibOutput")

for (i in seq_along(filelist)) {
    # load the rainfall, ET and flow data
  pred_data <- window(merge(flow_zoo[,i], Gridrain_zoo[,i], maxT_zoo[,i]),
                      start=start.date, end=end.date)
  colnames(pred_data) <- c("Q","P","E")
  # # another storage data frame for the residuals
```

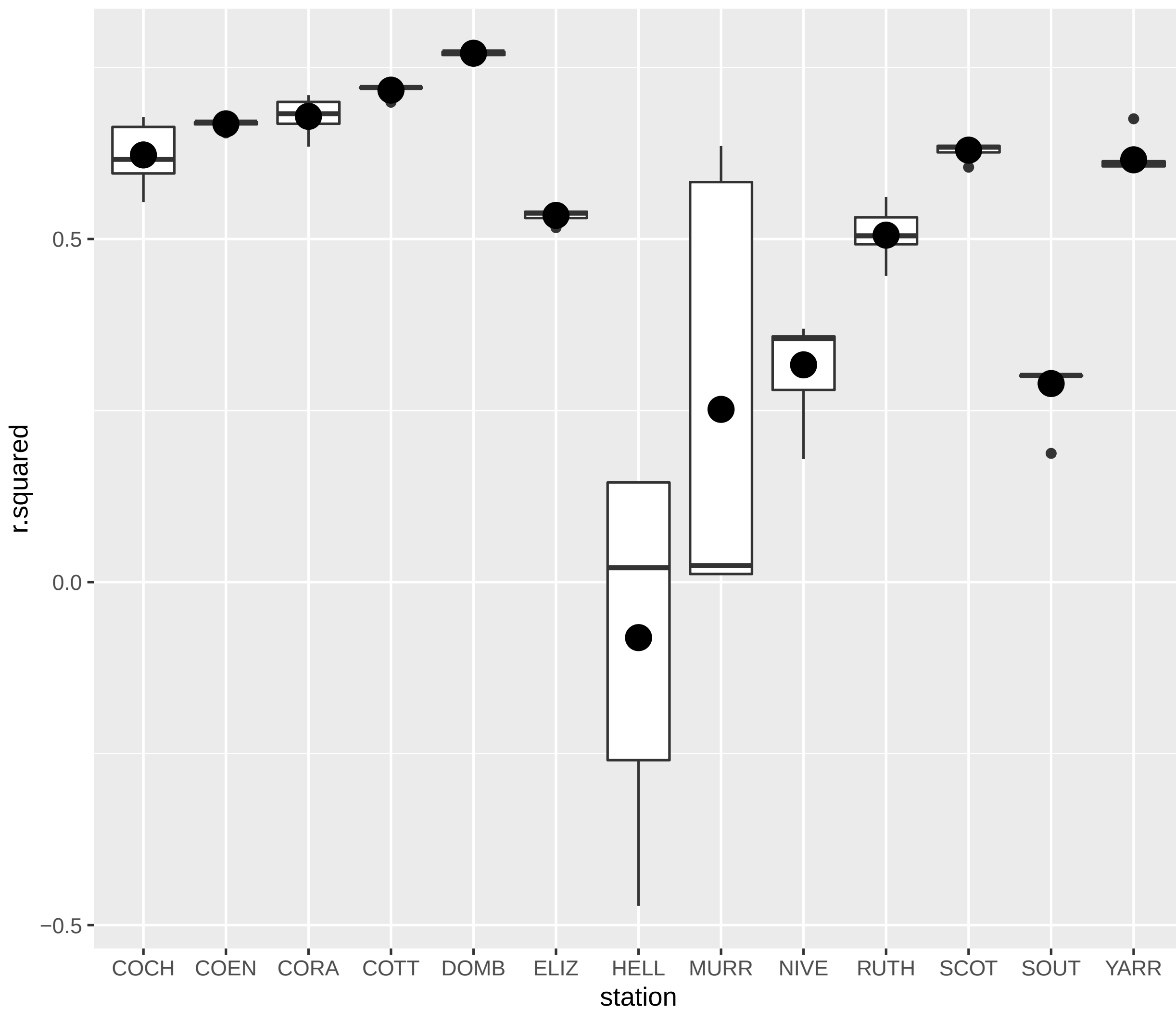

Figure 3: Calibration results for the SimHyd model across 10 iterations using station rainfall data.

```r
resid_out <- data.frame(matrix(0,ncol=10,nrow=nrow(pred_data)))
# load the relevant output
load(paste("../Projectdata/HPCResults/",
           filelist[grep(Stations[i,1],filelist)],sep=""))
# extract the model and update with the parameters
Mod <- Output$mod

mod_Res[[i]] <- Output$Store
Chiew[,1] <- Stations[i,1]
Results[,1] <- Stations[i,1]


# run through all iterations
for (j in 1:(nrow(Output$Store))) {
  # testing
  #j <- 1
  # update the model with the final fitted parameters, for each iteration
  Mod <- update(Mod, INSC=Output$Store[j,7],COEFF=Output$Store[j,8],
                  SQ=Output$Store[j,9],SMSC=Output$Store[j,10],
                  SUB=Output$Store[j,11],CRAK=Output$Store[j,12],
                  K=Output$Store[j,13],
                etmult=Output$Store[j,14], DELAY=Output$Store[j,5],
                X_m = Output$Store[j,6],
              return_state=F)


  # now predict the model output and use pred_data
  pred_mod <- predict(Mod,newdata=pred_data, all=T,na.rm=F)

  # store the residuals
  resid_out[,j] <- pred_data$Q-pred_mod

  Results[j,2:3] <- c(summary(Mod)$r.squared,summary(Mod)$rel.bias)

  # Now run the Chiew 2006 simulations on all the data
  mu <- cbind(rep(c(-15,-10,0,10),3),c(rep(0,4),rep(5,4),rep(10,4)))
  # Create the precipitation and ET data variations
  # station data
  RAIN <- Gridrain_zoo
  test <- list()
  for (k in 1:nrow(mu)) {
    temp <- as.data.frame(cbind((1+mu[k,1]/100)*RAIN[,i],
                          (1+mu[k,2]/100)*maxT_zoo[,i]))
    test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                    list(year=format(time(flow_zoo),"%Y")),sum,na.rm=T)))
    test[[k]] <- test[[k]][,-3]
  }
  clim_adj <- do.call(rbind,test)

  # now run the different pred results
  for (k in 1:ncol(pred_results)) {
    # run the model over all data
    pred_data2 <- window(merge(flow_zoo[,i],
```

```
                                  (1+mu[k,1]/100)*RAIN[,i],
                                  (1+mu[k,2]/100)*maxT_zoo[,i]))
      colnames(pred_data2) <- c("Q","P","E")

      pred_results[,k] <- predict(Mod,newdata=pred_data2, all=T,na.rm=F)
    }
    # summarise the data annually
    pred_ann <- apply(pred_results,2,
              function(x) aggregate(x,list(year=format(time(flow_zoo),"%Y")),
                                         sum,na.rm=T))
    ann_flow <- rep(pred_ann[[1]][,2],6)
    pred_t <- do.call(rbind,pred_ann)
    # Now add the ET and precipitation data
    pred_ann <- data.frame(pred_t,rain=clim_adj[,2],maxT=clim_adj[,3])
    # summarise base rain and temp
    ann_rain <- rep(aggregate(RAIN[,i],list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo[,i],
                          list(year=format(time(flow_zoo),"%Y")),
                          sum,na.rm=T),6)

    # Now calculate the difference
    pred_diff <- pred_ann
    pred_diff[,2] <- pred_diff[,2] - ann_flow
    pred_diff[,3] <- pred_diff[,3] - ann_rain
    pred_diff[,4] <- pred_diff[,4] - ann_maxT
    # Now fit a linear model (least squares (Chiew, 2006))
    fit <- lm(x~rain + maxT,data=pred_diff)
    # store the results
    Chiew[j,2:5] <- c(coef(fit)[2:3],summary(fit)$coefficients[2:3,4])


  }
  Chiew_Res[[i]] <- Chiew
  sum_Res[[i]] <- Results
  Residuals[[i]] <- resid_out
}
```

We can now temporary write away the results and the residuals and make some initial plots. Further plots will be generated later when comparing to the non-parametric epsilon (ϵ).

```
# write away the results
OutputTrends <- do.call(rbind,sum_Res)
write.csv(OutputTrends,
          file="../ProjectData/SimHydGridHPC_modelperformance.csv",
          row.names=F)

OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,
          file="../ProjectData/SimHydGridHPC_ChiewAnalysis.csv",
          row.names=F)
pander(aggregate(OutputChiew[,2:5],list(Station=OutputChiew$station),
                 mean),
```

```
          caption="Results SimHydGrid epsilon fit with significance")
```

Table 4: Results SimHydGrid epsilon fit with significance

Station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COCH	0.879	9.532e-05	3.759e-41	0.9802
COEN	0.8688	0.0004683	1.281e-236	0.915
CORA	0.7462	-0.002028	7.363e-167	0.7524
COTT	0.9022	-0.006645	1.989e-215	0.7274
DOMB	0.9349	-0.007156	3.94e-265	0.3763
ELIZ	0.8126	-2.098e-05	4.225e-216	0.9943
HELL	0.8072	-0.009166	3.266e-231	0.2749
MURR	0.8311	-0.00253	1.461e-147	0.6522
NIVE	0.9356	-0.0022	7.566e-316	0.6997
RUTH	0.7575	0.08575	3.662e-23	0.6742
SCOT	0.8866	-0.03023	2.586e-304	0.1981
SOUT	0.955	-0.002351	9.354e-285	0.6448
YARR	0.9302	-0.01677	3.128e-272	0.2938

```
saveRDS(Residuals,file="../ProjectData/residuals/SimHydGridHPCresiduals.RDS")

OutputMod_GridSimHyd <- do.call(rbind,mod_Res)
saveRDS(OutputMod_GridSimHyd,file="../ProjectData/SimHydHPCModelResults.RDS")

OutputMod_GridSimHyd <- OutputMod_GridSimHyd[,c(1:4,ncol(OutputMod_GridSimHyd))]
OutputMod_GridSimHyd$model <- "SimHyd Gridded Rainfall Data"

p <- ggplot(OutputMod_GridSimHyd,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
                      size=5,aes(col=rel.bias))
print(p)
```

6. Final plot comparing performance of all models

```
OutputMod <- rbind(OutputMod_GridGR4J,
                   OutputMod_GridSimHyd)
p <- ggplot(OutputMod,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
                      size=5,aes(colour=rel.bias))
p <- p + facet_wrap(~model) +
  xlab("Station") + ylab("Nash Sutcliffe Efficiency")
p <- p + theme(axis.text.x = element_text(size = rel(2), angle = 45,
                                          vjust=0.5),
               strip.text=element_text(size=rel(2), face = "bold"),
               axis.text.y = element_text(size = rel(2)),
               axis.title.x = element_text(size=rel(2), vjust=0),
               axis.title.y = element_text(size=rel(2), face = "bold.italic"))
```

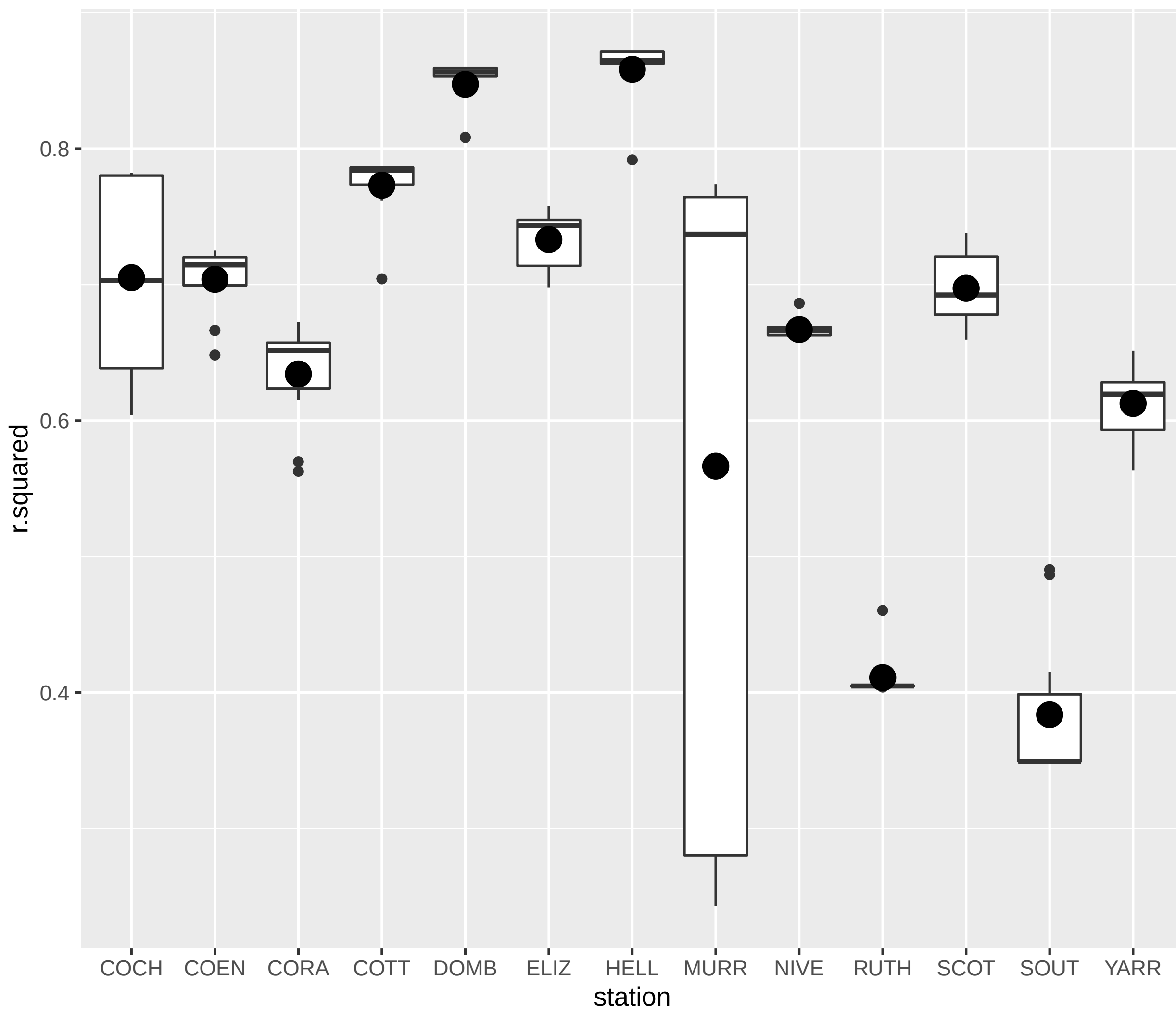

Figure 4: Calibration results for the SimHyd model across 10 iterations with gridded rainfall

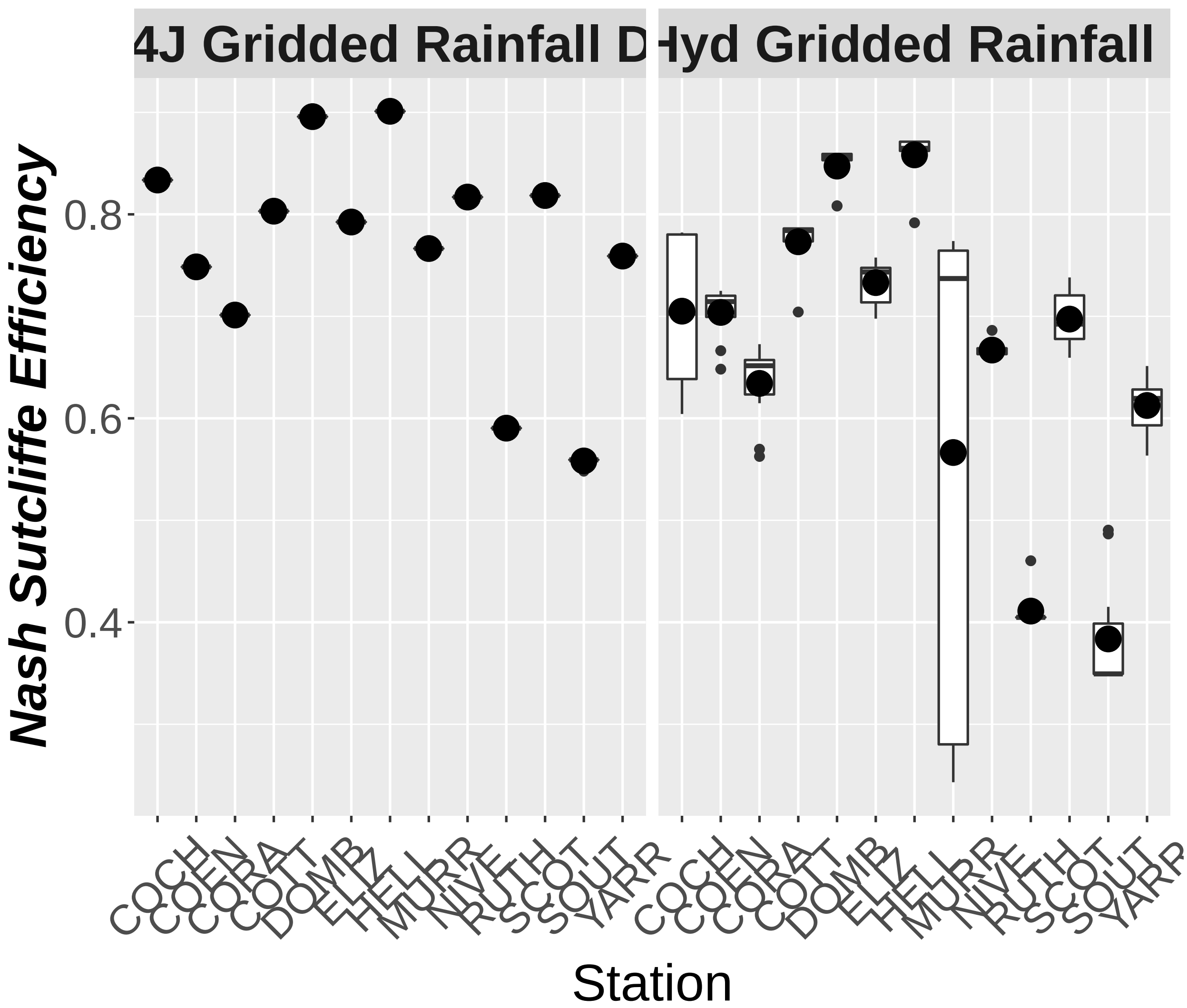

Figure 5: Comparing the performance of different models

```
print(p)
```

```
#labels <- c(GR4JGriddedData = "GR4J Gridded Rainfall Data", SimHydGridStationData = "SimHyd Gridded Ra
#p <- p + facet_grid(. ~ model, labeller=labeller(model = labels))
tiff("../Manuscript/Figure4_ModelCalStats.tif",
      width=16*480,height=12*480, res=600, compression="lzw")
print(p)
dev.off()
```

```
## pdf
##   2
```

```
# Write away the data
save(OutputMod,file="../ProjectData/ModelResults.Rdata")
```

```r
OutputMod2 <- rbind(OutputMod_GridGR4J,
                   OutputMod_GR4J,
                   OutputMod_GridSimHyd,
                    OutputMod_SimHyd)
save(OutputMod2,file="../ProjectData/ModelResults2.Rdata")
```