Analysis of Monthly HPC results

Willem Vervoort, Michaela Dolk & Floris van Ogtrop

2017-05-21

```
# root dir
knitr::opts_knit$set(root.dir = "C:/Users/rver4657/ownCloud/Virtual Experiments/VirtExp")
knitr::opts_chunk$set(echo = TRUE)
# LOAD REQUIRED PACKAGES # #####
library(pander)
library(tidyverse)
library(xts)
library(zoo)
library(ggplot2)
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.), submitted to Journal of Hydrology. This is the 9^{th} part of the series that analyses the monthly results from the GR4J and SimHyd model fitting on the High Performance computer Artemis at the University of Sydney.

This is a coomparison to part 6 of the series in which the daily data were analysed and also a comparison to the original Chiew (2006) paper that was based on monthly data.

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.

To recap, we have 2 different High Performance Computing monthly results for each of the 13 catchments. The two models (GR4J and SimHyd) were fitted to monhtly data for 10 years of climate data (1970 - 1980). Here we are only using the station data and not using the gridded data.

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 scripts related to the high performance computing and the HPC scripts are stored in the Rcode/HPC folder. However, in contrast to the daily modelling, this analysis cannot run use Viney's objective function as this does not work on the monthly data, so the monthly fit are (also following Chiew, 2006) based on fitting to the Nash Sutcliffe Efficiency (NSE), which in hydromad is `~hmadstat("r.squared")`, as explained in the hydromad helpfile for `hmadstat()` and on the website.

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.

```
load("../projectData/MonthlyDataOut.Rdata")
flow_zoo_m <- dataOut_Month[[1]]
rain_zoo_m <- dataOut_Month[[2]]
maxT_zoo_m <- dataOut_Month[[3]]
flow_rain_maxT_monthly <-dataOut_Month[[4]]
Stations <- read.csv("Data/CatchmentCharact.csv")

# compile SimHyd
rcode_dir <- paste(getwd(),"RCode/HPC",sep="/")
source(paste(rcode_dir,"Simhyd.r",sep="/"))
```

Define the beginning and end date for the modelling

```
start.date <- "Jan 1981"
end.date <- "Dec 2010"
```

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

```
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_m)),
                           Pmin10ET0=numeric(length=nrow(flow_zoo_m)),
                           P0ET0=numeric(length=nrow(flow_zoo_m)),
                           Pplus10ET0=numeric(length=nrow(flow_zoo_m)),
                           Pmin15ETplus5=numeric(length=nrow(flow_zoo_m)),
                           Pmin10ETplus5=numeric(length=nrow(flow_zoo_m)),
                           P0ETplus5=numeric(length=nrow(flow_zoo_m)),
                           Pplus10ETplus5=numeric(length=nrow(flow_zoo_m)),
                           Pmin15ETplus10=numeric(length=nrow(flow_zoo_m)),
                           Pmin10ETplus10=numeric(length=nrow(flow_zoo_m)),
                           P0ETplus10=numeric(length=nrow(flow_zoo_m)),
                           Pplus10ETplus10=numeric(length=nrow(flow_zoo_m)))
```

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.

```r
# find the list of files with GR4J results
filelist <- dir("../Projectdata/HPCResults/Monthly",
                pattern = "GR4JMonthCalibOutput")

for (i in seq_along(filelist)) {
  #i <- 1
  # load the rainfall, ET and flow data
  pred_data <- merge(flow_zoo_m[,i], rain_zoo_m[,i], maxT_zoo_m[,i])
  pred_data <- pred_data[time(pred_data) >= start.date &
                           time(pred_data) <= 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/monthly/",filelist[i],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_m
    test <- list()
    for (k in 1:nrow(mu)) {
```

```r
      temp <- as.data.frame(cbind((1+mu[k,1]/100)*RAIN[,i],
                            (1+mu[k,2]/100)*maxT_zoo_m[,i]))
      test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                        list(year=format(time(flow_zoo_m),"%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_m[,i],
                                (1+mu[k,1]/100)*RAIN[,i],
                                (1+mu[k,2]/100)*maxT_zoo_m[,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_m),"%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_m),"%Y")),
                             sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo_m[,i],
                             list(year=format(time(flow_zoo_m),"%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)
```

```r
write.csv(OutputTrends,
          file="../ProjectData/MonthlyGR4JHPC_modelperformance.csv",
          row.names=F)

OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,file="../ProjectData/MonthlyGR4JHPC_ChiewAnalysis.csv",
          row.names=F)

pander(OutputChiew, 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
COTT	0.8886	-0.07017	3.06e-91	5.038e-09
COTT	0.8888	-0.0702	3.062e-91	5.001e-09
COTT	0.8852	-0.07007	1.031e-91	4.082e-09
COTT	0.8886	-0.07018	3.066e-91	5.019e-09
COTT	0.889	-0.07022	3.039e-91	4.99e-09
COTT	0.8854	-0.07008	1.027e-91	4.08e-09
COTT	0.8854	-0.07008	1.019e-91	4.069e-09
COTT	0.8579	-0.06259	6.172e-93	4.058e-08
COTT	0.8888	-0.0702	3.046e-91	5.005e-09
COTT	0.8852	-0.07007	1.019e-91	4.068e-09
RUTH	0.9996	-0.06043	5.7e-126	8.531e-13
RUTH	0.9996	-0.06043	5.624e-126	8.519e-13
RUTH	0.9996	-0.06043	5.591e-126	8.494e-13
RUTH	0.9996	-0.06043	5.676e-126	8.521e-13
RUTH	0.9996	-0.06043	5.642e-126	8.511e-13
RUTH	0.9996	-0.06043	5.605e-126	8.505e-13
RUTH	0.9899	-0.06123	4.181e-127	1.8e-13
RUTH	0.9996	-0.06043	5.708e-126	8.515e-13
RUTH	0.9899	-0.06123	4.157e-127	1.795e-13
RUTH	0.9899	-0.06123	4.161e-127	1.795e-13
CORA	0.8366	-0.0449	9.585e-149	4.12e-16
CORA	0.8367	-0.0449	9.323e-149	4.124e-16
CORA	0.8367	-0.04489	9.407e-149	4.135e-16
CORA	0.8367	-0.04489	9.326e-149	4.127e-16
CORA	0.8367	-0.0449	9.314e-149	4.127e-16
CORA	0.8367	-0.04489	9.358e-149	4.128e-16
CORA	0.8367	-0.0449	9.348e-149	4.121e-16
CORA	0.8367	-0.04489	9.362e-149	4.132e-16
CORA	0.8367	-0.04489	9.365e-149	4.137e-16
CORA	0.8367	-0.0449	9.578e-149	4.13e-16
ELIZ	0.8289	-0.03269	1.458e-158	1.379e-05
ELIZ	0.8289	-0.03269	1.456e-158	1.38e-05
ELIZ	0.8289	-0.03269	1.458e-158	1.38e-05
ELIZ	0.8289	-0.03268	1.459e-158	1.383e-05
ELIZ	0.7753	-0.02126	4.864e-157	0.002619
ELIZ	0.7753	-0.02126	4.859e-157	0.00262
ELIZ	0.8433	-0.04482	1.737e-153	1.697e-08
ELIZ	0.8288	-0.03268	1.47e-158	1.386e-05
ELIZ	0.8288	-0.03268	1.467e-158	1.382e-05
ELIZ	0.8288	-0.03268	1.462e-158	1.382e-05

station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COCH	0.7957	-0.06048	2.787e-210	4.82e-11
COCH	0.7957	-0.06048	2.76e-210	4.83e-11
COCH	0.7957	-0.06048	2.76e-210	4.811e-11
COCH	0.7957	-0.06047	2.789e-210	4.847e-11
COCH	0.7956	-0.06046	2.849e-210	4.889e-11
COCH	0.7956	-0.06046	2.798e-210	4.874e-11
COCH	0.7957	-0.06048	2.735e-210	4.805e-11
COCH	0.7957	-0.06047	2.781e-210	4.835e-11
COCH	0.7958	-0.06049	2.701e-210	4.796e-11
COCH	0.7957	-0.06047	2.757e-210	4.837e-11
COEN	0.8812	-0.0443	2.937e-142	7.354e-12
COEN	0.8812	-0.0443	2.941e-142	7.355e-12
COEN	0.8812	-0.0443	2.987e-142	7.371e-12
COEN	0.8812	-0.0443	2.911e-142	7.345e-12
COEN	0.8812	-0.0443	2.956e-142	7.361e-12
COEN	0.8245	-0.04147	1.948e-165	4.123e-15
COEN	0.8812	-0.0443	2.914e-142	7.346e-12
COEN	0.8812	-0.0443	2.977e-142	7.368e-12
COEN	0.8812	-0.0443	2.94e-142	7.355e-12
COEN	0.8812	-0.0443	2.956e-142	7.36e-12
SCOT	0.9298	-0.03282	9.946e-274	2.985e-24
SCOT	0.9298	-0.03282	9.889e-274	2.99e-24
SCOT	0.9298	-0.03282	9.866e-274	2.986e-24
SCOT	0.9298	-0.03282	9.908e-274	2.986e-24
SCOT	0.9299	-0.03282	9.707e-274	2.983e-24
SCOT	0.9298	-0.03282	9.928e-274	2.988e-24
SCOT	0.9298	-0.03282	1.01e-273	2.991e-24
SCOT	1.099	-0.02716	5.511e-232	2.986e-09
SCOT	1.099	-0.02717	5.515e-232	2.942e-09
SCOT	0.9298	-0.03282	1e-273	2.986e-24
HELL	0.8109	-0.02639	6.729e-260	3.919e-08
HELL	0.8109	-0.02639	6.708e-260	3.93e-08
HELL	1.135	-0.1178	7.539e-280	3.711e-64
HELL	0.8109	-0.02639	6.726e-260	3.925e-08
HELL	0.8109	-0.02639	6.731e-260	3.918e-08
HELL	0.8109	-0.02639	6.716e-260	3.927e-08
HELL	0.8109	-0.02639	6.719e-260	3.918e-08
HELL	0.8109	-0.02639	6.73e-260	3.919e-08
HELL	0.8109	-0.0264	6.741e-260	3.913e-08
HELL	0.7602	-0.0187	2.804e-261	2.562e-05
NIVE	0.9359	-0.09052	1.519e-243	5.492e-25
NIVE	0.9358	-0.09051	1.537e-243	5.477e-25
NIVE	0.9359	-0.09053	1.484e-243	5.417e-25
NIVE	0.9359	-0.09052	1.508e-243	5.452e-25
NIVE	0.9359	-0.09053	1.488e-243	5.426e-25
NIVE	0.9358	-0.09052	1.489e-243	5.394e-25
NIVE	0.9358	-0.09052	1.506e-243	5.393e-25
NIVE	0.9358	-0.09052	1.507e-243	5.397e-25
NIVE	0.9358	-0.09051	1.537e-243	5.481e-25
NIVE	0.9357	-0.09051	1.53e-243	5.424e-25
MURR	1.046	-0.03622	2.056e-245	3.622e-15
MURR	1.046	-0.03622	2.023e-245	3.612e-15

station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
MURR	1.046	-0.03622	2.077e-245	3.628e-15
MURR	1.046	-0.03622	2.06e-245	3.624e-15
MURR	1.046	-0.03622	2.058e-245	3.623e-15
MURR	1.046	-0.03622	2.057e-245	3.622e-15
MURR	1.018	-0.06474	1.235e-230	7.883e-37
MURR	1.046	-0.03622	2.039e-245	3.616e-15
MURR	1.046	-0.03622	2.126e-245	3.633e-15
MURR	1.046	-0.03622	2.071e-245	3.626e-15
SOUT	0.6945	-0.04938	2.499e-190	6.334e-19
SOUT	0.6293	-0.04121	8.573e-189	3.177e-16
SOUT	0.6943	-0.04935	2.85e-190	6.552e-19
SOUT	0.6944	-0.04936	2.675e-190	6.459e-19
SOUT	0.6945	-0.04937	2.598e-190	6.424e-19
SOUT	0.6293	-0.04121	8.58e-189	3.179e-16
SOUT	0.6293	-0.04121	8.579e-189	3.177e-16
SOUT	0.6947	-0.04939	2.431e-190	6.299e-19
SOUT	0.6946	-0.04938	2.429e-190	6.292e-19
SOUT	0.6945	-0.04937	2.544e-190	6.369e-19
YARR	0.9493	-0.03858	1.046e-291	4.306e-25
YARR	0.9493	-0.03858	1.049e-291	4.308e-25
YARR	0.9493	-0.03858	1.043e-291	4.309e-25
YARR	0.9493	-0.03858	1.045e-291	4.308e-25
YARR	0.9493	-0.03858	1.05e-291	4.31e-25
YARR	0.9493	-0.03858	1.047e-291	4.307e-25
YARR	0.9494	-0.03858	1.034e-291	4.311e-25
YARR	0.9493	-0.03858	1.044e-291	4.308e-25
YARR	1.024	-0.03749	7.368e-306	9.784e-24
YARR	0.9493	-0.03858	1.043e-291	4.309e-25
DOMB	0.293	-0.03253	7.487e-140	3.087e-21
DOMB	0.293	-0.03254	7.138e-140	3.05e-21
DOMB	0.08562	-0.009898	2.838e-42	2.032e-05
DOMB	0.293	-0.03254	7.431e-140	3.082e-21
DOMB	0.293	-0.03254	7.415e-140	3.08e-21
DOMB	0.293	-0.03254	7.402e-140	3.075e-21
DOMB	0.293	-0.03254	7.3e-140	3.068e-21
DOMB	0.293	-0.03254	7.344e-140	3.073e-21
DOMB	0.08561	-0.009898	2.845e-42	2.032e-05
DOMB	0.293	-0.03254	7.425e-140	3.079e-21

```r
save(Residuals,file="../ProjectData/residuals/MonthlyGR4JHPCresiduals.Rdata")

OutputMod_GR4J <- do.call(rbind,mod_Res)
save(OutputMod_GR4J,file="../ProjectData/MonthlyGR4JHPCModelResults.Rdata")

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

OutputMod_GR4J$model <- "MonthlyGR4J"

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))
```

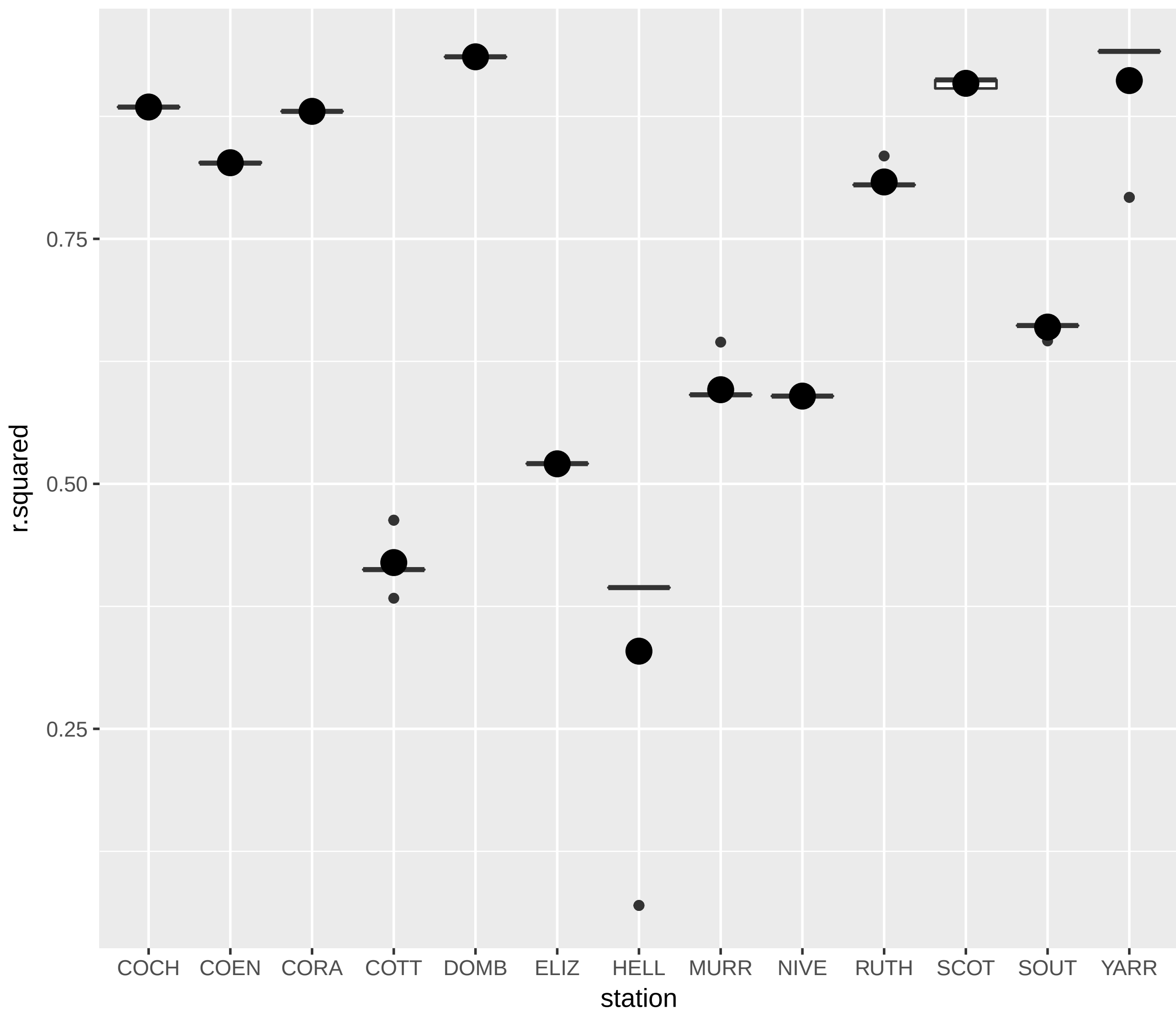

Figure 1: Calibration results for the GR4J model across 10 iterations using station rainfall data.

```
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/monthly/",
                pattern = "SimhydMonthCalibOutput")

for (i in seq_along(Stations[,1])) {
  # load the rainfall, ET and flow data
  pred_data <- merge(flow_zoo_m[,i], rain_zoo_m[,i], maxT_zoo_m[,i])
  pred_data <- pred_data[time(pred_data) >= start.date &
```

```r
                           time(pred_data) <= 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/monthly/",filelist[i],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_m
  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_m[,i]))
    test[[k]] <- do.call(cbind,apply(temp,2,function(x) aggregate(x,
                    list(year=format(time(flow_zoo_m),"%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)) {
```

```r
      # run the model over all data
      pred_data2 <- window(merge(flow_zoo_m[,i],
                           (1+mu[k,1]/100)*RAIN[,i],
                           (1+mu[k,2]/100)*maxT_zoo_m[,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_m),"%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_m),"%Y")),
                              sum,na.rm=T),6)
    ann_maxT <- rep(aggregate(maxT_zoo_m[,i],
                              list(year=format(time(flow_zoo_m),"%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/MonthlySimHydHPC_modelperformance.csv",
          row.names=F)

OutputChiew <- do.call(rbind,Chiew_Res)
write.csv(OutputChiew,
          file="../ProjectData/MonthlySimHydHPC_ChiewAnalysis.csv",
          row.names=F)
pander(OutputChiew, caption="Results Simhyd epsilon fit with significance")
```

Table 2: Results Simhyd epsilon fit with significance

station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COTT	0.8755	-0.005557	1.318e-106	0.5895
COTT	0.874	-0.005569	2.118e-106	0.5886
COTT	0.8757	-0.005558	1.321e-106	0.5895
COTT	0.8749	-0.005564	1.639e-106	0.589
COTT	0.875	-0.005565	1.616e-106	0.589
COTT	0.8739	-0.005569	2.134e-106	0.5886
COTT	0.875	-0.005561	1.529e-106	0.5892
COTT	0.8755	-0.005557	1.333e-106	0.5895
COTT	0.875	-0.005566	1.682e-106	0.589
COTT	0.8755	-0.005557	1.333e-106	0.5895
RUTH	0.9143	-0.004609	2.588e-148	0.4785
RUTH	0.9156	-0.004597	1.132e-148	0.4794
RUTH	0.9159	-0.004596	9.532e-149	0.4793
RUTH	0.9174	-0.004597	3.263e-149	0.4787
RUTH	0.9148	-0.004604	1.964e-148	0.479
RUTH	0.9111	-0.004614	1.17e-147	0.4784
RUTH	0.911	-0.004614	1.212e-147	0.4784
RUTH	0.9125	-0.004618	7.098e-148	0.4781
RUTH	0.9166	-0.004598	5.651e-149	0.4789
RUTH	0.9149	-0.004609	2.019e-148	0.4786
CORA	0.9454	-0.001055	1.013e-181	0.8311
CORA	0.9456	-0.001055	7.308e-182	0.8309
CORA	0.9456	-0.001056	7.134e-182	0.8308
CORA	0.9457	-0.001055	6.432e-182	0.8309
CORA	0.9459	-0.001055	5.628e-182	0.8308
CORA	0.9455	-0.001055	7.882e-182	0.831
CORA	0.9458	-0.001054	5.762e-182	0.831
CORA	0.946	-0.001056	3.241e-182	0.8305
CORA	0.9456	-0.001055	7.409e-182	0.8309
CORA	0.9456	-0.001055	7.637e-182	0.831
ELIZ	0.9319	-0.0005683	4.628e-167	0.943
ELIZ	0.9325	-0.00057	5.329e-167	0.9429
ELIZ	0.9324	-0.0005672	3.54e-167	0.9431
ELIZ	0.9319	-0.0005682	4.549e-167	0.9431
ELIZ	0.9318	-0.0005694	5.392e-167	0.9429
ELIZ	0.9319	-0.0005682	4.738e-167	0.9431
ELIZ	0.9318	-0.0005694	5.363e-167	0.9429
ELIZ	0.9317	-0.0005711	6.468e-167	0.9428
ELIZ	0.9319	-0.0005681	4.495e-167	0.9431
ELIZ	0.9318	-0.0005689	5.08e-167	0.943
COCH	0.9674	-0.004233	1.369e-245	0.6402
COCH	0.9675	-0.004224	1.273e-245	0.6408
COCH	0.9674	-0.004233	1.554e-245	0.6402
COCH	0.9675	-0.004232	1.326e-245	0.6402
COCH	0.9678	-0.00423	1.164e-245	0.6404
COCH	0.9675	-0.004223	1.21e-245	0.6409
COCH	0.9677	-0.004233	1.181e-245	0.6401
COCH	0.9678	-0.004229	1.11e-245	0.6404
COCH	0.9679	-0.004265	1.064e-245	0.6376
COCH	0.9674	-0.004225	1.28e-245	0.6407

station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
COEN	0.946	-0.001875	7.078e-204	0.6905
COEN	0.9516	-0.00197	8.883e-205	0.6759
COEN	0.9516	-0.00187	8.767e-205	0.6914
COEN	0.9517	-0.001949	6.974e-205	0.679
COEN	0.9516	-0.00189	7.916e-205	0.6883
COEN	0.9508	-0.003129	7.719e-205	0.5062
COEN	0.9516	-0.001877	9.27e-205	0.6904
COEN	0.9516	-0.001891	8.71e-205	0.6881
COEN	0.9516	-0.00187	8.883e-205	0.6915
COEN	0.948	-0.001873	2.626e-204	0.6906
SCOT	0.9784	-0.0007428	3.001e-295	0.7971
SCOT	0.9755	-0.0006679	2.253e-304	0.8082
SCOT	0.9777	-0.000724	2.08e-297	0.7999
SCOT	0.9784	-0.0007428	3.004e-295	0.7971
SCOT	0.9784	-0.0007428	3.004e-295	0.7971
SCOT	0.9784	-0.0007428	3.004e-295	0.7971
SCOT	0.9623	-0.0003329	1.433e-314	0.8973
SCOT	0.9785	-0.0007899	2.781e-295	0.7845
SCOT	0.9784	-0.0007428	3.004e-295	0.7971
SCOT	0.9784	-0.0007428	3.003e-295	0.7971
HELL	0.9723	-0.001869	1.363e-283	0.7099
HELL	0.9737	-0.001868	3.103e-280	0.7152
HELL	0.9727	-0.001871	5.848e-284	0.7093
HELL	0.9723	-0.001869	1.358e-283	0.7099
HELL	0.9727	-0.001905	8.272e-284	0.7043
HELL	0.9731	-0.001867	1.101e-281	0.713
HELL	0.9736	-0.001868	1.792e-280	0.7148
HELL	0.9727	-0.001872	5.751e-284	0.709
HELL	0.9727	-0.001956	6.254e-284	0.6967
HELL	0.9727	-0.001871	5.687e-284	0.7092
NIVE	0.9654	-0.0005155	8.887e-254	0.9493
NIVE	0.9634	-0.0005258	1.529e-253	0.9483
NIVE	0.9634	-0.0005258	1.536e-253	0.9483
NIVE	0.9634	-0.0005258	1.604e-253	0.9483
NIVE	0.9634	-0.0005258	1.505e-253	0.9483
NIVE	0.9634	-0.0005259	1.546e-253	0.9483
NIVE	0.9694	-0.0006486	1.548e-255	0.9359
NIVE	0.9644	-0.0005214	1.066e-253	0.9487
NIVE	0.9634	-0.0005258	1.526e-253	0.9483
NIVE	0.9636	-0.0005243	1.362e-253	0.9484
MURR	0.9678	-0.001292	1.033e-249	0.7487
MURR	0.9678	-0.001293	1.095e-249	0.7486
MURR	0.9682	-0.001291	6.845e-250	0.7487
MURR	0.9678	-0.001291	9.681e-250	0.7488
MURR	0.9679	-0.001293	1.16e-249	0.7487
MURR	0.9678	-0.001292	4.561e-250	0.7482
MURR	0.9681	-0.001291	7.661e-250	0.7488
MURR	0.9682	-0.003145	5.422e-250	0.4352
MURR	0.9676	-0.001291	1.11e-249	0.7489
MURR	0.9678	-0.001292	1e-249	0.7486
SOUT	0.9611	-0.002277	3.445e-259	0.6549
SOUT	0.9615	-0.002271	1.748e-259	0.6554

station	eta_p	eta_e	pvalue_eta_p	pvalue_eta_e
SOUT	0.9611	-0.002276	3.178e-259	0.655
SOUT	0.9611	-0.002276	3.166e-259	0.655
SOUT	0.9611	-0.002277	3.318e-259	0.6549
SOUT	0.9614	-0.002269	1.314e-259	0.6554
SOUT	0.9611	-0.002277	3.29e-259	0.6549
SOUT	0.9613	-0.002268	1.724e-259	0.6558
SOUT	0.9614	-0.002269	1.256e-259	0.6554
SOUT	0.9611	-0.002277	3.331e-259	0.6549
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
YARR	0.9831	-0.0009169	6.736e-297	0.7966
YARR	0.9757	-0.000919	2.239e-294	0.7972
YARR	0.9831	-0.0009169	6.735e-297	0.7966
YARR	0.9831	-0.0009169	6.733e-297	0.7966
DOMB	0.9735	-9.463e-06	0	0.9979
DOMB	0.9744	-9.021e-06	0	0.998
DOMB	0.9749	-9.811e-06	0	0.9978
DOMB	0.9744	-8.671e-06	0	0.9981
DOMB	0.9735	-9.81e-06	0	0.9978
DOMB	0.9749	-9.262e-06	0	0.9979
DOMB	0.975	-8.71e-06	0	0.9981
DOMB	0.9744	-8.15e-06	0	0.9982
DOMB	0.9745	-9.137e-06	0	0.998
DOMB	0.9748	-1.011e-05	0	0.9978

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

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

OutputMod_SimHyd <- OutputMod_SimHyd[,c(1:4,ncol(OutputMod_SimHyd))]
OutputMod_SimHyd$model <- "MonthlySimHydStationData"

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)
```

6. Final plot comparing performance of all models

```r
OutputMod <- rbind(OutputMod_GR4J, OutputMod_SimHyd)
p <- ggplot(OutputMod,aes(station,r.squared)) +  geom_boxplot()
p <- p + stat_summary(fun.y=mean, geom="point", shape=16,
```

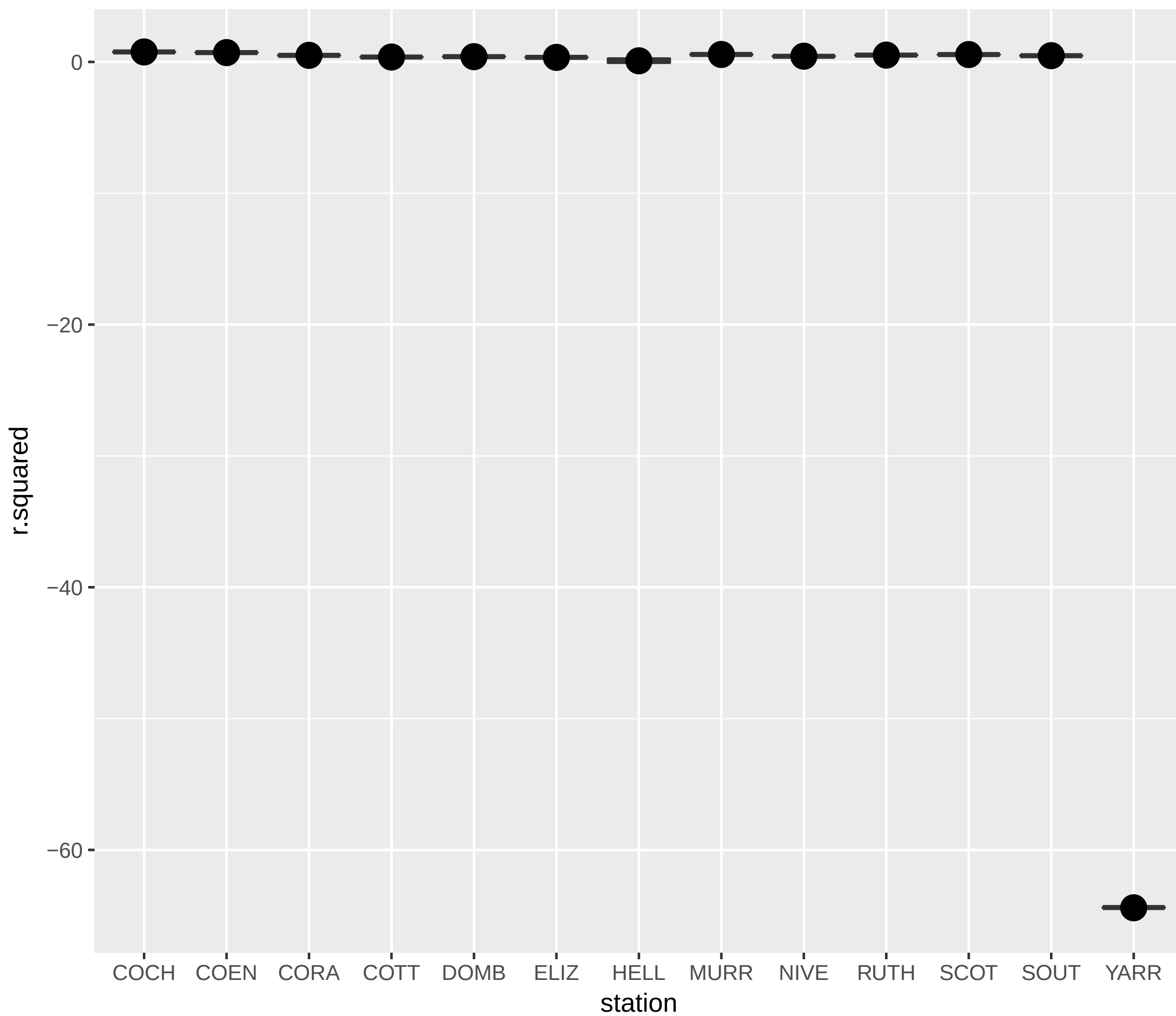

Figure 2: Calibration results for the SimHyd model across 10 iterations using station rainfall data.

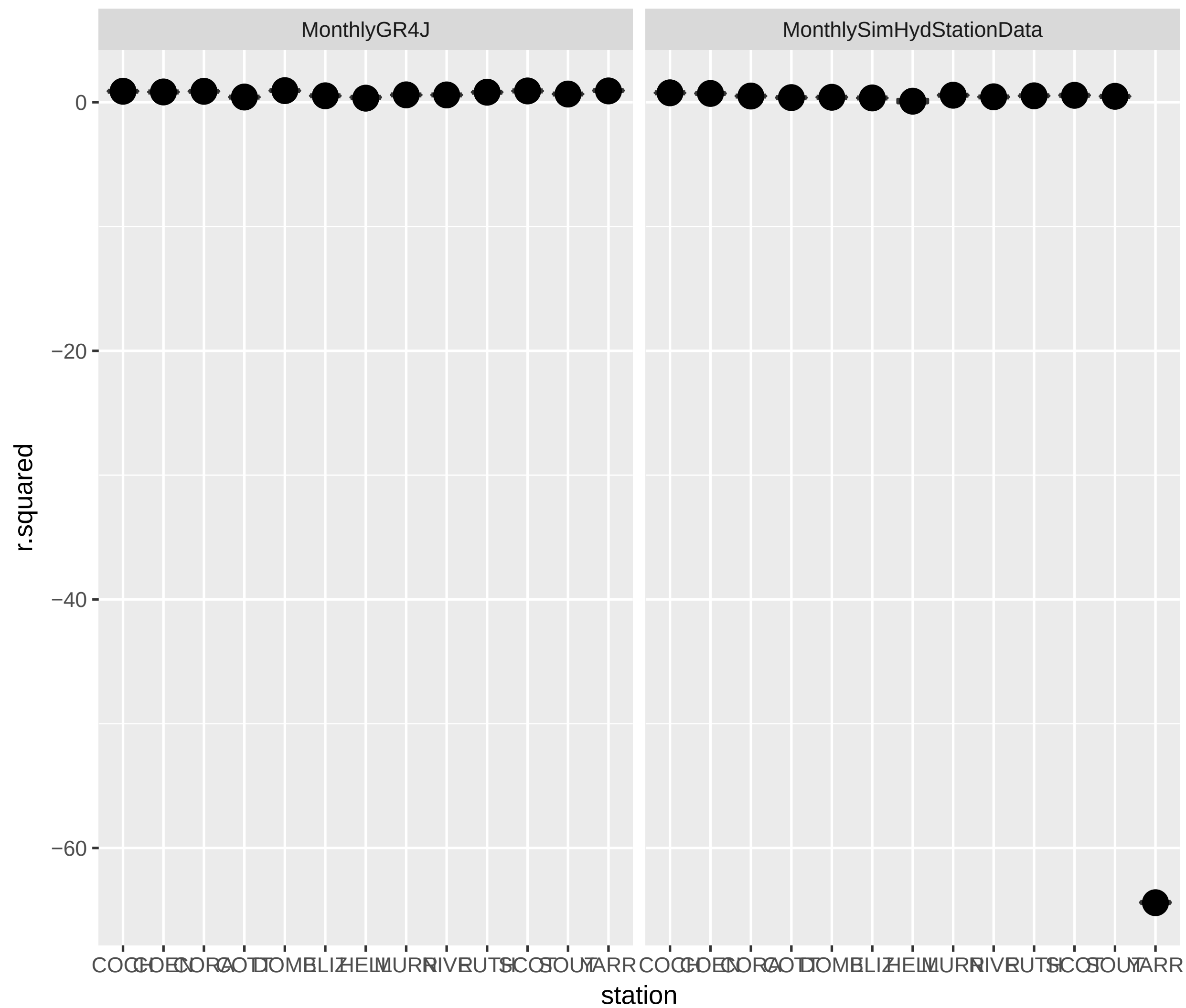

Figure 3: Comparing the performance of different models

```
                              size=5,aes(colour=rel.bias))
p <- p + facet_wrap(~model)
print(p)
```

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