Testing Bias Correction

Adam Griffin; UKCEH

17/11/2020

In this document we analyse the difference between including and excluding the bias correction on gridded data making use of UKCP18 meteorological inputs.

Here the bias correction used is that of Guillod et al. (2018), which has been applied in a pooled approach(using all the RCM members to generate the bias correction factor together) and a seperate approach (seperate correction factors for each RCM)

Data Exploration

```
wd <- paste0("//nercwlsmb01/prj/ukscape2.2",
             "/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011")

D <- dir(wd, pattern="dmamflow_out.nc", recursive=T, full.names=T)
D_TBC <- D[grepl("T", sapply(D, function(x){strsplit(x,"/")[[1]][9]}))]
D_FBC <- D[grepl("F", sapply(D, function(x){strsplit(x,"/")[[1]][9]}))]

rcm_nos <- c(1,4,5,6,7,8,9,10,11,12,13,15)

CATAL <- catalogue()
CATAL <- CATAL[CATAL$`nrfa-mean-flow`,]

statlocs <- read_delim("./catcoords_istat_jstat_01000.txt", delim="\t", col_types=cols(
  gauge = col_double(),
  istat = col_double(),
  jstat = col_double()
))
statlocs$jstat <- statlocs$jstat - 250
```

First we will open the files and see what kind of data it is: predominantly netCDF files.

```
NCIN <- nc_open(D[1])
print(NCIN)
```

```
## File //nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
##
##      1 variables (excluding dimension variables):
##         float dmamflow[Easting,Northing,Time]   (Chunking: [700,1000,1])  (Compression: shuffle,level
##             units: m3 s-1
##             standard_name: dmamflow
##             long_name: Annual maxima of daily mean river flow
##             _FillValue: -999
##             missing_value: -999
```

```
##
##      3 dimensions:
##         Northing  Size:1000
##             standard_name: Northing
##             axis: Y
##             units: GB National Grid
##         Easting  Size:700
##             standard_name: Easting
##             axis: X
##             units: GB National Grid
##         Time  Size:30   *** is unlimited ***
##             standard_name: Time
##             axis: T
##             units: calendar_year as %Y
##             calendar: 360_day
```

```r
dmamax <- ncvar_get(NCIN, "dmamflow")
nor <- ncvar_get(NCIN, "Northing")
eas <- ncvar_get(NCIN, "Easting")

CATAL <- CATAL[CATAL$northing < 1000000 & CATAL$northing > 0 &
                 CATAL$easting < 700000 & CATAL$easting > 0 &
                 CATAL$id %in% statlocs$gauge,]


rn <- which((!is.na(dmamax[,,2])) & (dmamax[,,2] > -1), arr.ind=T)
rn_pos <- cbind(nor[rn[,2]], eas[rn[,1]])

dim(dmamax)
```

```
## [1]  700 1000   30
```

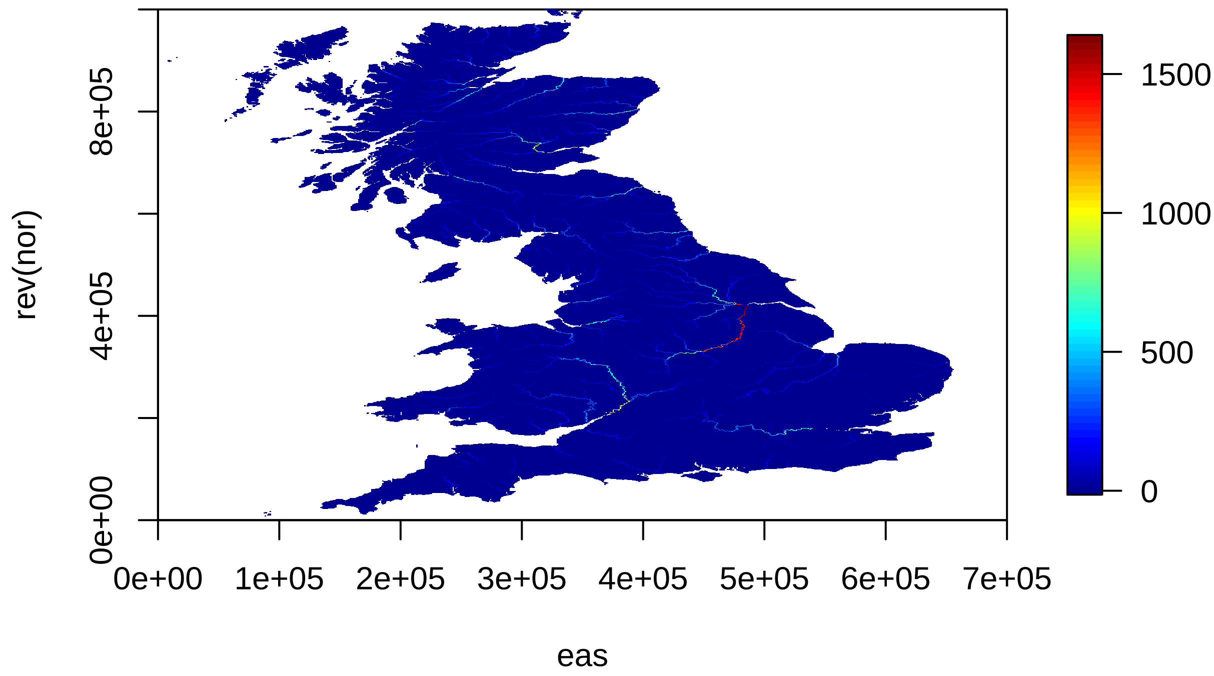
```r
str(dmamax)
```

```
##  num [1:700, 1:1000, 1:30] NA NA NA NA NA NA NA NA NA NA ...
```

```r
summary(dmamax)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's 
##       -1       -1       -1        6       -1     4559 14380141
```

```r
#dmamax[250:260, 80:90,2]
image.plot(x=eas, y=rev(nor), dmamax[,1000:1,2])
```




```
nc_close(NCIN)
```

Here we have time series of annual maxima based on daily flow for each gridpoint.

We're going to compare this to the daily mean flow for gauged stations in the UK. Note this is different from the standard AMAX calculations which use instantaneous peak flow to determine flood frequency curves. However, since the G2G outputs are at a daily resolution, this is the most comparable approach.

First we extract the AMAX values from the gauged daily flow from the NRFA API, and the corresponding amax from the relevant gridpoints in the RCM outputs.

```
am_summ_obs <- data.frame(station=numeric(),
                          year=numeric(),
                          value=numeric())
##### OBS #####
for(k in 1:nrow(CATAL)){
  if(k %% 100 == 0){print(k)}
  stn1 <- CATAL$id[k]

  gdf1 <- gdf(stn1)
  gdf_hyyr <- findHydrolYr(index(gdf1))
  gdf1 <- cbind(as.vector(gdf1), gdf_hyyr)
  colnames(gdf1)[1] <- "gdf"
  suppressMessages({
    gdf2 <- gdf1 %>%
          dplyr::filter(yr >= 1981 & yr <= 2010) %>%
          group_by(yr) %>%
          summarise(value=max(gdf, na.rm=T))
```

```
  })
  gdf2$station <- stn1
  am_summ_obs <- rbind(am_summ_obs, gdf2[,c(3,1,2)])
}
```

```
## [1] 100
## [1] 200
## [1] 300
## [1] 400
## [1] 500
## [1] 600
## [1] 700
## [1] 800
## [1] 900
## [1] 1000
## [1] 1100
## [1] 1200
## [1] 1300
```

```
colnames(am_summ_obs) <- c("station", "year", "value")

#wider version
AM <- dcast(am_summ_obs, station~year, value.var="value")

write_csv(am_summ_obs, "S:/Data/BiasCorrection/am_obs.csv")
```

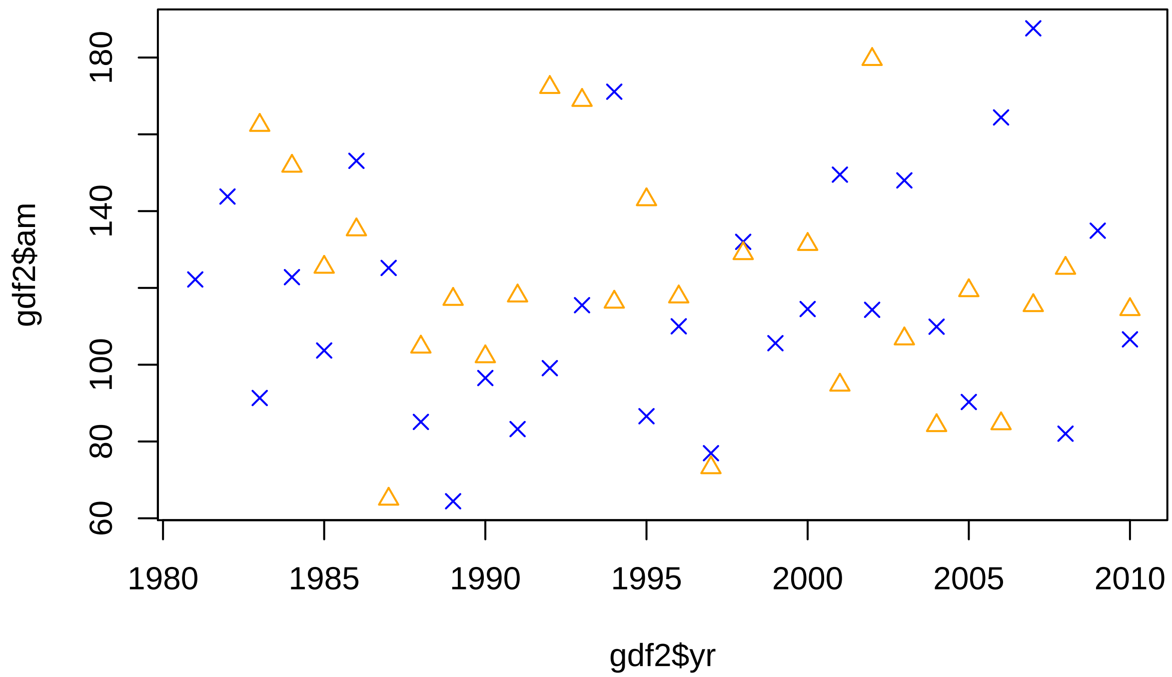
Here we extract the DM AMAX values for all the stations we can, any with a DMF time series on the NRFA. We take the nearest gridpoint to the station as a comparison, and take the annual maxima of the daily mean flow in each year (no accounting for missing data in the DMF at this point.)

```
am_summ_TBC <- data.frame(rcm=numeric(),
                          station=numeric(),
                          year=numeric(),
                          value=numeric())
am_summ_FBC <- data.frame(rcm=numeric(),
                          station=numeric(),
                          year=numeric(),
                          value=numeric())

#CATAL

rn1 <- matrix(NA, nrow=nrow(CATAL), ncol=2)
for(k in 1:nrow(CATAL)){
  #print(k)
  if(k%%50==0){print(k)}
  pos <- c(CATAL$northing[k], CATAL$easting[k])
  rn1[k, ] <- unlist(statlocs[statlocs$gauge==CATAL$id[k], ][2:3], use.names=F)
  #dista <- apply(rn_pos, 1, function(x){sqrt(sum((x-pos)^2))})
  #rn1[k,] <- rn[which.min(dista),]
}
```

```
## [1] 50
## [1] 100
## [1] 150
## [1] 200
```

```
## [1] 250
## [1] 300
## [1] 350
## [1] 400
## [1] 450
## [1] 500
## [1] 550
## [1] 600
## [1] 650
## [1] 700
## [1] 750
## [1] 800
## [1] 850
## [1] 900
## [1] 950
## [1] 1000
## [1] 1050
## [1] 1100
## [1] 1150
## [1] 1200
## [1] 1250
## [1] 1300
```

```r
### EXAMPLE ###
k <- 2; stn1 <- CATAL$id[k]

gdf1 <- gdf(stn1)
gdf_hyyr <- findHydrolYr(index(gdf1))
gdf1 <- cbind(as.vector(gdf1), gdf_hyyr)
colnames(gdf1)[1] <- "gdf"
gdf2 <- gdf1 %>% dplyr::filter(yr >= 1981 & yr <= 2010) %>%
  group_by(yr) %>% summarise(am=max(gdf, na.rm=T), am_doy=DOY[which.max(gdf)])

ukcp_am1 <- dmamax[rn1[k,1],rn1[k,2],]

plot(gdf2$yr, gdf2$am, col="blue", pch=4, ylim=range(c(gdf2$am, gdf2$ukcp_am), na.rm=T))
points(gdf2$yr, ukcp_am1, col="orange", pch=2)
```




```
parglo(lmoms(gdf2$am))
```

```
## $type
## [1] "glo"
##
## $para
##          xi       alpha       kappa
## 112.7583217  16.7388594  -0.1277475
##
## $source
## [1] "parglo"
```

```
parglo(lmoms(ukcp_am1))
```

```
## $type
## [1] "glo"
##
## $para
##          xi       alpha       kappa
## 122.8074406  23.5469803  -0.3240924
##
## $source
## [1] "parglo"
```

```
##### RCMS #####
for(j in seq_len(length(rcm_nos))){
  for(l in 1:2){
    if(l == 1){
```

```r
      NCIN <- nc_open(D_TBC[j])
      print(D_TBC[j])
    }else{
      NCIN <- nc_open(D_FBC[j])
      print(D_FBC[j])
    }
    dmamax <- ncvar_get(NCIN, "dmamflow")

    for(k in 1:nrow(CATAL)){
      stn1 <- CATAL$id[k]
      am_k <- dmamax[rn1[k,1],rn1[k,2],]
      df <-data.frame(rcm=rcm_nos[j],
                      station=stn1,
                      year=1981:2010,
                      value=am_k)

      if(l == 1){
        am_summ_TBC <- rbind(am_summ_TBC, df)
      }else{
        am_summ_FBC <- rbind(am_summ_FBC, df)
      }
    }
  }
}
```

```
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
## [1] "//nercwlsmb01/prj/ukscape2.2/run_gb_hmfg2g_ukcp18rcm12km/outputs/baseline_PEi_198012_201011/rcm
```

```r
readr::write_csv(x=am_summ_TBC, path="S:/Data/BiasCorrection/am_TBC.csv")
readr::write_csv(x=am_summ_FBC, path="S:/Data/BiasCorrection/am_FBC.csv")
```

AMAX analysis

We can't expect the AMAX values to match up between RCMs and the observations, but we can check average differences between the same RCM with and without bias correction (TBC and FBC, respectively), and we can compare GLO fits; we restrict the observed DM AMAX to hydrological years 1981-2010.

```
am_glo_obs <- data.frame(station=numeric(),
                         reclen=numeric(),
                         loc=numeric(),
                         sca=numeric(),
                         sha=numeric(),
                         QMED=numeric(),
                         Q20=numeric(),
                         Q50=numeric())

am_glo_TBC <- data.frame(rcm=numeric(),
                         station=numeric(),
                         reclen=numeric(),
                         loc=numeric(),
                         sca=numeric(),
                         sha=numeric(),
                         QMED=numeric(),
                         Q20=numeric(),
                         Q50=numeric())

am_glo_FBC <- data.frame(rcm=numeric(),
                         station=numeric(),
                         reclen=numeric(),
                         loc=numeric(),
                         sca=numeric(),
                         sha=numeric(),
                         QMED=numeric(),
                         Q20=numeric(),
                         Q50=numeric())

qt <- 1 - 1/c(2,20,50)
for(i in 1:nrow(CATAL)){
  if(i %% 100 == 0){print(CATAL$id[i])}
  try({
    am_temp <- am_summ_obs %>% dplyr::filter(station==CATAL$id[i])
    am_glo_obs[i, 1:5] <- c(CATAL$id[i],
                            length(unique(am_temp$year)),
                            parglo(lmoms(am_temp$value))$para)
    am_glo_obs[i,6:8]  <- quaglo(qt, parglo(lmoms(am_temp$value)))
  }, silent=T)
  for(j in 1:length(rcm_nos)){
    try({
    am_temp <- am_summ_TBC %>% dplyr::filter(station==CATAL$id[i] & rcm==rcm_nos[j])
    am_glo_TBC[(i-1)*length(rcm_nos) + j, 1:6] <- c(rcm_nos[j], CATAL$id[i],
                                                    length(unique(am_temp$year)),
                                                    parglo(lmoms(am_temp$value))$para)
    am_glo_TBC[(i-1)*length(rcm_nos) + j, 7:9]  <- quaglo(qt, parglo(lmoms(am_temp$value)))

    am_temp <- am_summ_FBC %>% dplyr::filter(station==CATAL$id[i] & rcm==rcm_nos[j])
```

```r
    am_glo_FBC[(i-1)*length(rcm_nos) + j, 1:6] <- c(rcm_nos[j], CATAL$id[i],
                                                       length(unique(am_temp$year)),
                                                       parglo(lmoms(am_temp$value))$para)
    am_glo_FBC[(i-1)*length(rcm_nos) + j, 7:9]  <- quaglo(qt, parglo(lmoms(am_temp$value)))
    }, silent=T)
  }
}
```

```
## [1] 14006
## [1] 21027
## [1] 27044
## [1] 28080
## [1] 33054
## [1] 38033
## [1] 39125
## [1] 43011
## [1] 52020
## [1] 55008
## [1] 67003
## [1] 73017
## [1] 85004
```

```r
write_csv(am_glo_obs, path="S:/Data/BiasCorrection/glo_obs.csv")
write_csv(am_glo_TBC, path="S:/Data/BiasCorrection/glo_TBC.csv")
write_csv(am_glo_FBC, path="S:/Data/BiasCorrection/glo_FBC.csv")
```

Here's a quick summary of the data: this very quick method does lead to some discrepencies, since I did not manually align the stations to the correct gridpoints in all cases. On the whole, both with an without bias correction (averaged across all ensemble members) show a negative bias (underestimating compared to observed).

```r
U_am <- unique(am_summ_obs$station)
U_obs <- unique(am_glo_obs$station) #1323
U_tbc <- unique(am_glo_TBC$station) #1511
U_fbc <- unique(am_glo_FBC$station)
CVID <- CATAL$id
V <- Reduce(intersect, list(U_am, U_obs, U_tbc, U_fbc, CVID)) #1322


am_glo_obs <- am_glo_obs %>% dplyr::filter(station %in% V)
am_glo_TBC <- am_glo_TBC %>% dplyr::filter(station %in% V)
am_glo_FBC <- am_glo_FBC %>% dplyr::filter(station %in% V)
am_summ_obs <- am_summ_obs %>% dplyr::filter(station %in% V)


CATAL0 <- CATAL[CATAL$id %in% V, ] #1402 rows
AM <- AM[AM$station %in% V,] #1402 rows
QM <- apply(AM[,-1],1,function(x){ifelse(!sum(is.na(x))<25,NA, median(x, na.rm=T))})
QM <- cbind(AM$station, QM)
dTBC <- am_glo_TBC %>% group_by(station) %>% summarise(qmed1 = mean(QMED, na.rm=T))
dFBC <- am_glo_FBC %>% group_by(station) %>% summarise(qmed1 = mean(QMED, na.rm=T))
```

```r
QT_wide <- am_glo_TBC %>% select(station, rcm, QMED)
QT_wide <- dcast(QT_wide, station~rcm, value.var="QMED")
QT_wide2 <- sapply(2:13, function(i){(QT_wide[,i] - QM[,2])/(QM[,2])})
```

```r
summary(QT_wide2)
```

```
##        V1                 V2                 V3                 V4          
##  Min.   :-0.94664   Min.   :-0.93150   Min.   :-0.93835   Min.   :-0.94005  
##  1st Qu.:-0.37859   1st Qu.:-0.37019   1st Qu.:-0.31325   1st Qu.:-0.41655  
##  Median :-0.21969   Median :-0.22010   Median :-0.13808   Median :-0.26464  
##  Mean   :-0.11548   Mean   :-0.09180   Mean   :-0.01213   Mean   :-0.14331  
##  3rd Qu.:-0.01747   3rd Qu.: 0.01011   3rd Qu.: 0.09094   3rd Qu.:-0.05001  
##  Max.   :10.21850   Max.   :10.31673   Max.   :12.74056   Max.   :10.45227  
##  NA's   :4          NA's   :4          NA's   :4          NA's   :4         
##        V5                 V6                 V7                  V8            
##  Min.   :-0.93195   Min.   :-0.94699   Min.   :-0.946536   Min.   :-0.950236  
##  1st Qu.:-0.34117   1st Qu.:-0.36385   1st Qu.:-0.376328   1st Qu.:-0.301999  
##  Median :-0.18309   Median :-0.20363   Median :-0.220150   Median :-0.123452  
##  Mean   :-0.05181   Mean   :-0.09450   Mean   :-0.101956   Mean   : 0.000273  
##  3rd Qu.: 0.03952   3rd Qu.: 0.01267   3rd Qu.:-0.009405   3rd Qu.: 0.113994  
##  Max.   :10.45797   Max.   : 9.81612   Max.   :10.441898   Max.   :11.779203  
##  NA's   :4          NA's   :4          NA's   :4           NA's   :4          
##        V9                V10                V11                V12          
##  Min.   :-0.92730   Min.   :-0.92913   Min.   :-0.95263   Min.   :-0.95034  
##  1st Qu.:-0.32536   1st Qu.:-0.38689   1st Qu.:-0.38462   1st Qu.:-0.41614  
##  Median :-0.17401   Median :-0.23767   Median :-0.23589   Median :-0.27794  
##  Mean   :-0.02375   Mean   :-0.09266   Mean   :-0.11456   Mean   :-0.16628  
##  3rd Qu.: 0.07151   3rd Qu.:-0.01821   3rd Qu.:-0.02072   3rd Qu.:-0.08499  
##  Max.   :11.86018   Max.   :11.63785   Max.   :10.49335   Max.   : 9.43126  
##  NA's   :4          NA's   :4          NA's   :4          NA's   :4         
```

```r
AA <- apply(QT_wide2, 2, function(x){table(sign(x))/length(x)})

QT_wide <- am_glo_FBC %>% select(station, rcm, QMED)
QT_wide <- dcast(QT_wide, station~rcm, value.var="QMED")
QT_wide2 <- sapply(2:13, function(i){(QT_wide[,i] - QM[,2])/(QM[,2])})
summary(QT_wide2)
```

```
##        V1                 V2                 V3                 V4          
##  Min.   :-0.92366   Min.   :-0.86529   Min.   :-0.93107   Min.   :-0.91061  
##  1st Qu.:-0.20701   1st Qu.:-0.16875   1st Qu.:-0.22136   1st Qu.:-0.25728  
##  Median : 0.02659   Median : 0.09495   Median :-0.01327   Median :-0.04814  
##  Mean   : 0.23323   Mean   : 0.41167   Mean   : 0.19107   Mean   : 0.17079  
##  3rd Qu.: 0.36028   3rd Qu.: 0.58287   3rd Qu.: 0.31252   3rd Qu.: 0.29812  
##  Max.   :15.48484   Max.   :19.94549   Max.   :18.71816   Max.   :16.92640  
##  NA's   :4          NA's   :4          NA's   :4          NA's   :4         
##        V5                 V6                 V7                V8          
##  Min.   :-0.93009   Min.   :-0.91968   Min.   :-0.9437   Min.   :-0.95671  
##  1st Qu.:-0.23254   1st Qu.:-0.20500   1st Qu.:-0.2711   1st Qu.:-0.30708  
##  Median :-0.01818   Median : 0.01245   Median :-0.0508   Median :-0.09919  
##  Mean   : 0.16552   Mean   : 0.21397   Mean   : 0.1012   Mean   : 0.03819  
##  3rd Qu.: 0.28466   3rd Qu.: 0.35784   3rd Qu.: 0.2317   3rd Qu.: 0.14320  
##  Max.   :13.86147   Max.   :14.81031   Max.   :13.4665   Max.   :12.94619  
##  NA's   :4          NA's   :4          NA's   :4         NA's   :4         
##        V9               V10                V11                V12          
##  Min.   :-0.8851   Min.   :-0.89184   Min.   :-0.93719   Min.   :-0.90578  
##  1st Qu.:-0.1146   1st Qu.:-0.17463   1st Qu.:-0.24723   1st Qu.:-0.19369  
##  Median : 0.1533   Median : 0.07069   Median :-0.02719   Median : 0.04611  
```

```
##  Mean   : 0.4346   Mean   : 0.30956   Mean   : 0.18207   Mean   : 0.26930
##  3rd Qu.: 0.5884   3rd Qu.: 0.43825   3rd Qu.: 0.32164   3rd Qu.: 0.42999
##  Max.   :19.9984   Max.   :17.85974   Max.   :15.95180   Max.   :15.91465
##  NA's   :4         NA's   :4          NA's   :4          NA's   :4
```

```
AA2 <- apply(QT_wide2, 2, function(x){table(sign(x))/length(x)})

print("****** QMED *******")
```

```
## [1] "****** QMED *******"
```

```
rbind(AA,AA2)
```

```
##         [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## -1 0.7592397 0.7381204 0.6810982 0.7877508 0.7138332 0.7349525 0.7550158
## 1  0.2365364 0.2576558 0.3146779 0.2080253 0.2819430 0.2608237 0.2407603
## -1 0.4635692 0.4223865 0.5100317 0.5438226 0.5153115 0.4825766 0.5607181
## 1  0.5322070 0.5733897 0.4857445 0.4519535 0.4804646 0.5131996 0.4350581
##         [,8]      [,9]     [,10]     [,11]     [,12]
## -1 0.6494192 0.7011616 0.7592397 0.7624076 0.7972545
## 1  0.3463569 0.2946146 0.2365364 0.2333685 0.1985216
## -1 0.6092925 0.3611404 0.4266103 0.5227033 0.4519535
## 1  0.3864836 0.6346357 0.5691658 0.4730729 0.5438226
```

```
rowSums(rbind(AA,AA2))/12
```

```
##        -1         1        -1         1
## 0.7366244 0.2591517 0.4891763 0.5065998
```

```
Q20_wide <- am_glo_TBC %>% select(station, rcm, Q20)
Q20_wide_obs <- data.frame(am_glo_obs %>% select(station, Q20))
Q20_wide <- dcast(Q20_wide, station~rcm, value.var="Q20")
Q20_wide2 <- sapply(2:13, function(i){(Q20_wide[,i] - Q20_wide_obs[,2]) / Q20_wide_obs[,2]})
summary(Q20_wide2)
```

```
##        V1                 V2                 V3                 V4
##  Min.   :-0.96222   Min.   :-0.96249   Min.   :-0.93777   Min.   :-0.9667
##  1st Qu.:-0.41886   1st Qu.:-0.44370   1st Qu.:-0.35171   1st Qu.:-0.4979
##  Median :-0.22220   Median :-0.26478   Median :-0.14762   Median :-0.3122
##  Mean   :-0.12243   Mean   :-0.15792   Mean   :-0.02619   Mean   :-0.1871
##  3rd Qu.: 0.05088   3rd Qu.:-0.02994   3rd Qu.: 0.16299   3rd Qu.:-0.0257
##  Max.   : 6.83307   Max.   : 6.42312   Max.   : 7.17555   Max.   : 7.6587
##        V5                 V6                 V7                 V8
##  Min.   :-0.95205   Min.   :-0.96323   Min.   :-0.96014   Min.   :-0.96175
##  1st Qu.:-0.35563   1st Qu.:-0.35311   1st Qu.:-0.38903   1st Qu.:-0.30642
##  Median :-0.16781   Median :-0.14469   Median :-0.20135   Median :-0.02625
##  Mean   :-0.04491   Mean   :-0.03049   Mean   :-0.06834   Mean   : 0.10320
##  3rd Qu.: 0.11443   3rd Qu.: 0.14195   3rd Qu.: 0.07633   3rd Qu.: 0.34481
##  Max.   : 7.17604   Max.   : 9.69557   Max.   :10.43611   Max.   : 9.29564
##        V9                V10                V11                V12
##  Min.   :-0.95171   Min.   :-0.95742   Min.   :-0.96175   Min.   :-0.95952
##  1st Qu.:-0.36484   1st Qu.:-0.46935   1st Qu.:-0.38816   1st Qu.:-0.43033
##  Median :-0.17225   Median :-0.31038   Median :-0.19051   Median :-0.24732
##  Mean   :-0.02469   Mean   :-0.16514   Mean   :-0.05791   Mean   :-0.11573
##  3rd Qu.: 0.12942   3rd Qu.:-0.04101   3rd Qu.: 0.10744   3rd Qu.: 0.03542
##  Max.   : 8.24402   Max.   : 6.31296   Max.   : 9.39443   Max.   : 7.87416
```

```r
AA <- apply(Q20_wide2, 2, function(x){table(sign(x))/length(x)})


Q20_wide <- am_glo_FBC %>% select(station, rcm, Q20)
Q20_wide_obs <- data.frame(am_glo_obs %>% select(station, Q20))
Q20_wide <- dcast(Q20_wide, station~rcm, value.var="Q20")
Q20_wide2 <- sapply(2:13, function(i){(Q20_wide[,i] - Q20_wide_obs[,2]) / Q20_wide_obs[,2]})
summary(Q20_wide2)
```

```
##        V1                 V2                 V3                 V4
##  Min.   :-0.94884   Min.   :-0.90114   Min.   :-0.92452   Min.   :-0.94232
##  1st Qu.:-0.28725   1st Qu.:-0.25738   1st Qu.:-0.28838   1st Qu.:-0.38014
##  Median :-0.01384   Median : 0.01801   Median :-0.06539   Median :-0.17301
##  Mean   : 0.17057   Mean   : 0.31102   Mean   : 0.15956   Mean   : 0.03698
##  3rd Qu.: 0.38359   3rd Qu.: 0.60738   3rd Qu.: 0.30090   3rd Qu.: 0.26205
##  Max.   :10.09916   Max.   :12.63720   Max.   :12.17561   Max.   :10.04841
##        V5                 V6                 V7                 V8
##  Min.   :-0.94492   Min.   :-0.94428   Min.   :-0.95597   Min.   :-0.970283
##  1st Qu.:-0.30651   1st Qu.:-0.28131   1st Qu.:-0.32308   1st Qu.:-0.379567
##  Median :-0.10104   Median :-0.03365   Median :-0.09147   Median :-0.146327
##  Mean   : 0.07446   Mean   : 0.15683   Mean   : 0.10193   Mean   : 0.001787
##  3rd Qu.: 0.24324   3rd Qu.: 0.37444   3rd Qu.: 0.30667   3rd Qu.: 0.191995
##  Max.   :10.19650   Max.   :12.02493   Max.   :12.79062   Max.   : 7.362563
##        V9                V10                V11                V12
##  Min.   :-0.9208   Min.   :-0.92965   Min.   :-0.94109   Min.   :-0.91390
##  1st Qu.:-0.2069   1st Qu.:-0.29130   1st Qu.:-0.34578   1st Qu.:-0.28056
##  Median : 0.0488   Median :-0.05196   Median :-0.08296   Median :-0.04825
##  Mean   : 0.3292   Mean   : 0.21550   Mean   : 0.12746   Mean   : 0.19684
##  3rd Qu.: 0.5538   3rd Qu.: 0.43324   3rd Qu.: 0.34575   3rd Qu.: 0.43518
##  Max.   :10.6131   Max.   :11.45225   Max.   :11.38101   Max.   : 9.70434
```

```r
AA2 <- apply(Q20_wide2, 2, function(x){table(sign(x))/length(x)})


print("******* Q20 *******")
```

```
## [1] "******* Q20 *******"
```

```r
rbind(AA,AA2)
```

```
##         [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## -1 0.7127772 0.7697994 0.6536431 0.7708553 0.6747624 0.6409715 0.7011616
## 1  0.2872228 0.2302006 0.3463569 0.2291447 0.3252376 0.3590285 0.2988384
## -1 0.5142555 0.4899683 0.5638860 0.6346357 0.5913411 0.5248152 0.5755016
## 1  0.4857445 0.5100317 0.4361140 0.3653643 0.4086589 0.4751848 0.4244984
##         [,8]      [,9]     [,10]     [,11]     [,12]
## -1 0.5258712 0.6515312 0.7761352 0.6652587 0.7243928
## 1  0.4741288 0.3484688 0.2238648 0.3347413 0.2756072
## -1 0.6293559 0.4593453 0.5417107 0.5649419 0.5248152
## 1  0.3706441 0.5406547 0.4582893 0.4350581 0.4751848
```

```r
rowSums(rbind(AA,AA2))/12
```

```
##        -1         1        -1         1
## 0.6889300 0.3110700 0.5512144 0.4487856
```

```r
Q50_wide <- am_glo_TBC %>% select(station, rcm, Q50)
Q50_wide_obs <- data.frame(am_glo_obs %>% select(station, Q50))
Q50_wide <- dcast(Q50_wide, station~rcm, value.var="Q50")
Q50_wide2 <- sapply(2:13, function(i){(Q50_wide[,i] - Q50_wide_obs[,2]) / Q50_wide_obs[,2]})
summary(Q50_wide2)
```

```
##        V1                  V2                  V3                  V4
##  Min.   :-0.96627   Min.   :-0.970235   Min.   :-0.934089   Min.   :-0.97330
##  1st Qu.:-0.44094   1st Qu.:-0.458984   1st Qu.:-0.369358   1st Qu.:-0.51344
##  Median :-0.21293   Median :-0.267688   Median :-0.148895   Median :-0.32276
##  Mean   :-0.08158   Mean   :-0.146377   Mean   : 0.000181   Mean   :-0.15247
##  3rd Qu.: 0.14155   3rd Qu.: 0.004472   3rd Qu.: 0.219248   3rd Qu.: 0.03249
##  Max.   : 5.97722   Max.   : 5.247566   Max.   : 5.722077   Max.   : 7.00871
##        V5                  V6                  V7                  V8
##  Min.   :-0.95822   Min.   :-0.96845   Min.   :-0.96557   Min.   :-0.96558
##  1st Qu.:-0.36577   1st Qu.:-0.37688   1st Qu.:-0.40994   1st Qu.:-0.31741
##  Median :-0.15468   Median :-0.11491   Median :-0.19136   Median : 0.01485
##  Mean   :-0.01086   Mean   : 0.04096   Mean   :-0.01972   Mean   : 0.19062
##  3rd Qu.: 0.19598   3rd Qu.: 0.24613   3rd Qu.: 0.13709   3rd Qu.: 0.47680
##  Max.   : 5.97676   Max.   : 9.95576   Max.   :10.61676   Max.   : 8.27455
##        V9                 V10                 V11                  V12
##  Min.   :-0.95894   Min.   :-0.96519   Min.   :-0.964015   Min.   :-0.96293
##  1st Qu.:-0.37583   1st Qu.:-0.50491   1st Qu.:-0.398065   1st Qu.:-0.43630
##  Median :-0.15008   Median :-0.31683   Median :-0.164462   Median :-0.22717
##  Mean   : 0.02642   Mean   :-0.15446   Mean   : 0.003619   Mean   :-0.05475
##  3rd Qu.: 0.23029   3rd Qu.:-0.02543   3rd Qu.: 0.200787   3rd Qu.: 0.10726
##  Max.   : 7.16800   Max.   : 5.25066   Max.   : 8.928395   Max.   : 7.60631
```

```r
AA <- apply(Q50_wide2, 2, function(x){table(sign(x))/length(x)})

Q50_wide <- am_glo_FBC %>% select(station, rcm, Q50)
Q50_wide_obs <- data.frame(am_glo_obs %>% select(station, Q50))
Q50_wide <- dcast(Q50_wide, station~rcm, value.var="Q50")
Q50_wide2 <- sapply(2:13, function(i){(Q50_wide[,i] - Q50_wide_obs[,2]) / Q50_wide_obs[,2]})
summary(Q50_wide2)
```

```
##        V1                  V2                  V3                  V4
##  Min.   :-0.95653   Min.   :-0.916062   Min.   :-0.91623   Min.   :-0.95149
##  1st Qu.:-0.31988   1st Qu.:-0.289992   1st Qu.:-0.31925   1st Qu.:-0.40729
##  Median :-0.03022   Median :-0.002034   Median :-0.05591   Median :-0.18339
##  Mean   : 0.19994   Mean   : 0.336041   Mean   : 0.19543   Mean   : 0.04522
##  3rd Qu.: 0.43367   3rd Qu.: 0.624387   3rd Qu.: 0.35405   3rd Qu.: 0.27398
##  Max.   : 8.58673   Max.   :10.673626   Max.   :10.18528   Max.   : 8.37345
##        V5                  V6                  V7                 V8
##  Min.   :-0.94982   Min.   :-0.95320   Min.   :-0.9613   Min.   :-0.9745
##  1st Qu.:-0.34309   1st Qu.:-0.30155   1st Qu.:-0.3515   1st Qu.:-0.4209
##  Median :-0.12777   Median :-0.02832   Median :-0.1019   Median :-0.1511
##  Mean   : 0.07425   Mean   : 0.18783   Mean   : 0.1487   Mean   : 0.0242
##  3rd Qu.: 0.27521   3rd Qu.: 0.45465   3rd Qu.: 0.3741   3rd Qu.: 0.2415
##  Max.   : 9.02267   Max.   :11.75945   Max.   :13.3304   Max.   : 5.8726
##        V9                 V10                 V11                 V12
##  Min.   :-0.93390   Min.   :-0.94151   Min.   :-0.94184   Min.   :-0.91791
##  1st Qu.:-0.23412   1st Qu.:-0.33446   1st Qu.:-0.37792   1st Qu.:-0.31294
##  Median : 0.05186   Median :-0.06727   Median :-0.08633   Median :-0.05842
```

```
##  Mean   : 0.35345   Mean   : 0.23333   Mean   : 0.15291   Mean   : 0.21925
##  3rd Qu.: 0.61216   3rd Qu.: 0.48162   3rd Qu.: 0.40675   3rd Qu.: 0.48059
##  Max.   : 8.26791   Max.   : 9.83278   Max.   : 9.96584   Max.   : 8.11792
```

```
AA2 <- apply(Q50_wide2, 2, function(x){table(sign(x))/length(x)})


print("******* Q50 ********")
```

```
## [1] "******* Q50 ********"
```

```
rbind(AA,AA2)
```

```
##          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
## -1 0.6800422 0.7465681 0.6325238 0.7243928 0.6441394 0.5913411 0.6599789
## 1  0.3199578 0.2534319 0.3674762 0.2756072 0.3558606 0.4086589 0.3400211
## -1 0.5174234 0.5015839 0.5480465 0.6356917 0.5997888 0.5269271 0.5744456
## 1  0.4825766 0.4984161 0.4519535 0.3643083 0.4002112 0.4730729 0.4255544
##          [,8]      [,9]     [,10]     [,11]     [,12]
## -1 0.4920803 0.6187962 0.7560718 0.6166843 0.7043295
## 1  0.5079197 0.3812038 0.2439282 0.3833157 0.2956705
## -1 0.6293559 0.4667371 0.5522703 0.5649419 0.5417107
## 1  0.3706441 0.5332629 0.4477297 0.4350581 0.4582893
```

```
rowSums(rbind(AA,AA2))/12
```

```
##        -1         1        -1         1
## 0.6555790 0.3444210 0.5549102 0.4450898
```

```
summary(am_glo_obs$QMED - am_glo_TBC$QMED)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -950.557  -25.275    3.213    8.861   40.594  845.365
```

```
summary(am_glo_obs$QMED - am_glo_FBC$QMED)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -959.9904  -33.9160   -0.5259   -2.1319   35.6937  844.5593
```

```
summary(am_glo_TBC$QMED - am_glo_FBC$QMED)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -460.681  -10.404   -3.664  -10.993   -1.144  269.076
```

```
sum(!is.na(am_glo_obs$QMED))
```

```
## [1] 947
```

```
sum(!is.na(am_glo_TBC$QMED)) # /12  = 1322
```

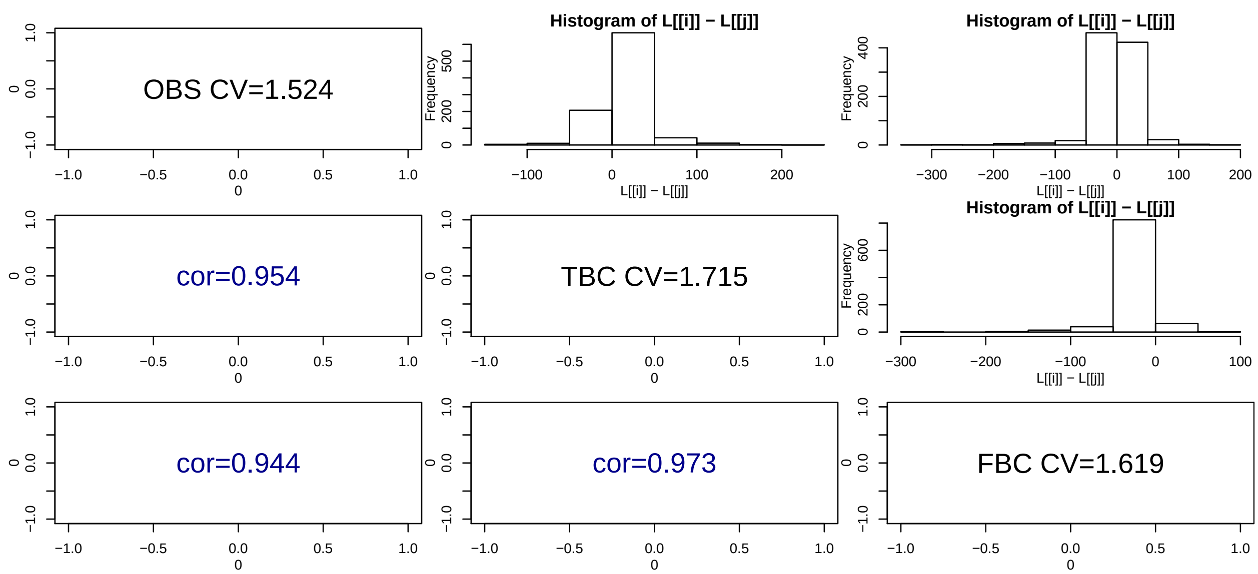
```
## [1] 11364
```

```
sum(!is.na(am_glo_FBC$QMED)) # /12  = 1322
```

```
## [1] 11364
```

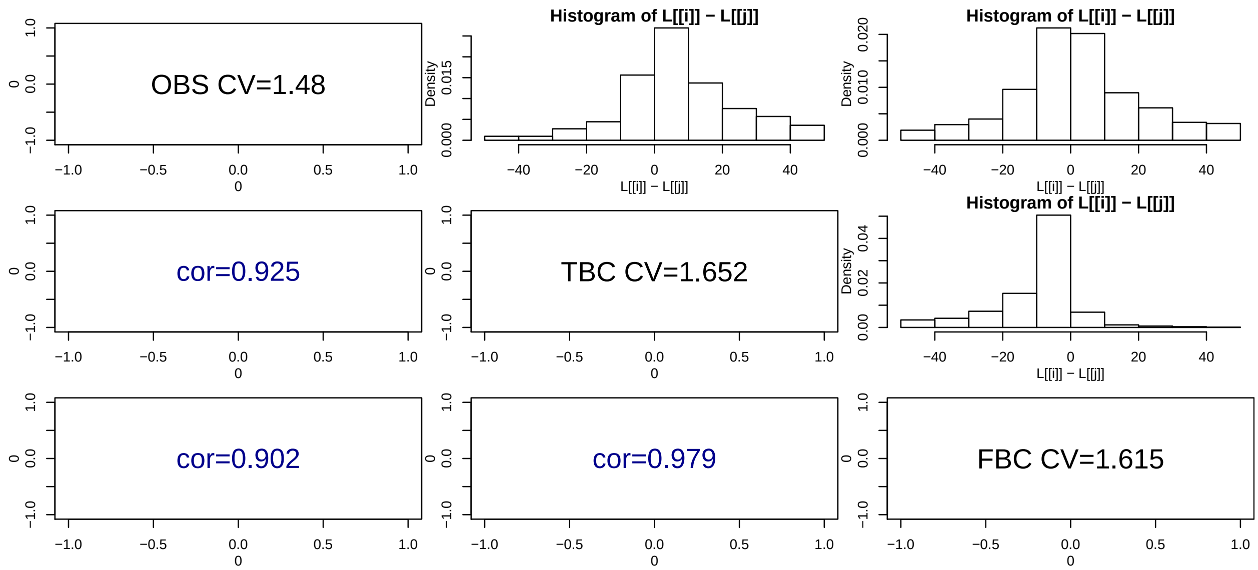
Plots

```r
qmed_TBC <- am_glo_TBC %>% dplyr::filter(rcm==1) %>% dplyr::select(QMED)
qmed_FBC <- am_glo_FBC %>% dplyr::filter(rcm==1) %>% dplyr::select(QMED)
qmed_obs <- am_glo_obs %>% dplyr::select(QMED)

nams <- c("OBS", "TBC", "FBC")
L <- list(unlist(qmed_obs, use.names=F),
          unlist(qmed_TBC, use.names=F),
          unlist(qmed_FBC, use.names=F))

par(mar=c(3,3,1,0), mgp=c(2,1,0), mfrow=c(3,3))

for(i in 1:3){
  for(j in 1:3){
    if(i==j){
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0(nams[i], " CV=",
                             round(sd(L[[i]], na.rm=T)/mean(L[[i]], na.rm=T),3)),
           cex=2)     #check sd, mean and cor
    }else if(i < j){
      hist(L[[i]] - L[[j]], breaks=10)
    }else{
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0("cor=",round(cor(L[[i]],L[[j]]),3)), cex=2, col="darkblue")
    }
  }
}
```



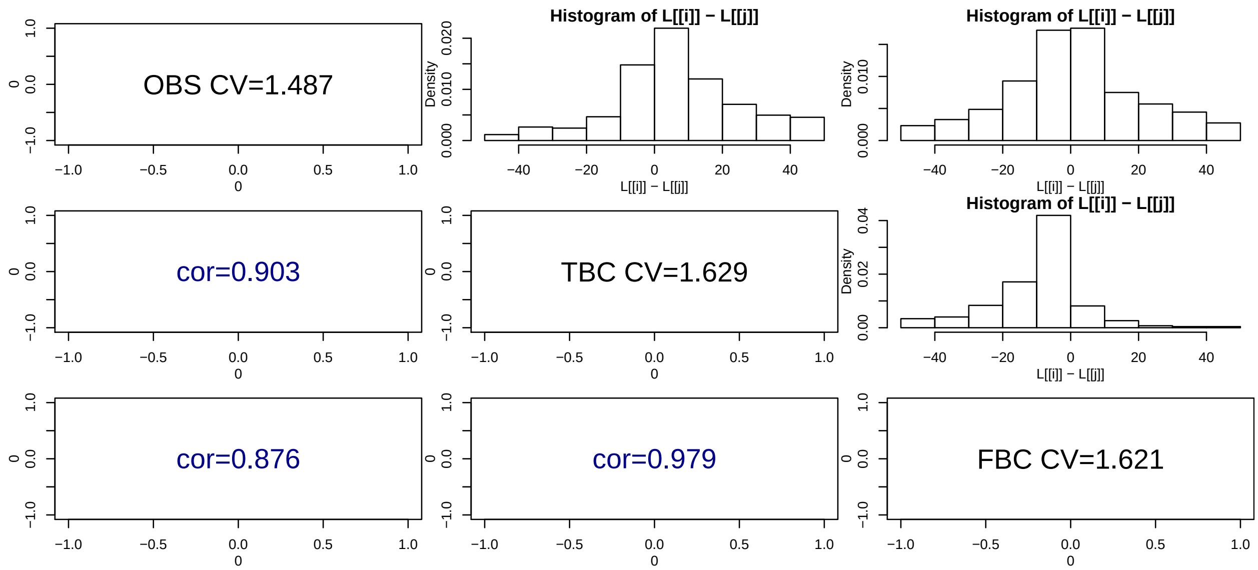

```r
q20_TBC <- am_glo_TBC %>% dplyr::filter(rcm==1) %>% dplyr::select(Q20)
q20_FBC <- am_glo_FBC %>% dplyr::filter(rcm==1) %>% dplyr::select(Q20)
q20_obs <- am_glo_obs %>% dplyr::select(Q20)
L <- list(unlist(q20_obs, use.names=F),
          unlist(q20_TBC, use.names=F),
          unlist(q20_FBC, use.names=F))

par(mar=c(3,3,1,0), mgp=c(2,1,0), mfrow=c(3,3))
```

```
for(i in 1:3){
  for(j in 1:3){
    if(i==j){
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0(nams[i],
                             " CV=",
                             round(sd(L[[i]], na.rm=T)/mean(L[[i]], na.rm=T),3)),
           cex=2)     #check sd, mean and cor
    }else if(i < j){
      hist(L[[i]] - L[[j]], breaks=c(-Inf,-50,-40,-30,-20,-10,0,10,20,30,40,50,Inf), xlim=c(-50,50))
    }else{
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0("cor=",round(cor(L[[i]],L[[j]]),digits=3)), cex=2, col="darkblue")
    }
  }
}
```




```
q50_TBC <- am_glo_TBC %>% dplyr::filter(rcm==1) %>% dplyr::select(Q50)
q50_FBC <- am_glo_FBC %>% dplyr::filter(rcm==1) %>% dplyr::select(Q50)
q50_obs <- am_glo_obs %>% dplyr::select(Q50)
L <- list(unlist(q50_obs, use.names=F),
          unlist(q50_TBC, use.names=F),
          unlist(q50_FBC, use.names=F))

par(mar=c(3,3,1,0), mgp=c(2,1,0), mfrow=c(3,3))

for(i in 1:3){
  for(j in 1:3){
    if(i==j){
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0(nams[i], " CV=",
                             round(sd(L[[i]], na.rm=T)/mean(L[[i]], na.rm=T),digits=3)),
           cex=2)     #check sd, mean and cor
    }else if(i < j){
      hist(L[[i]] - L[[j]], breaks=c(-Inf,-50,-40,-30,-20,-10,0,10,20,30,40,50,Inf), xlim=c(-50,50))
```

```
    }else{
      plot(0,0, xlim=c(-1,1), ylim=c(-1,1), pch=NA)
      text(0,0,labels=paste0("cor=",round(cor(L[[i]],L[[j]]),digits=3)), cex=2, col="darkblue")
    }
  }
}
```

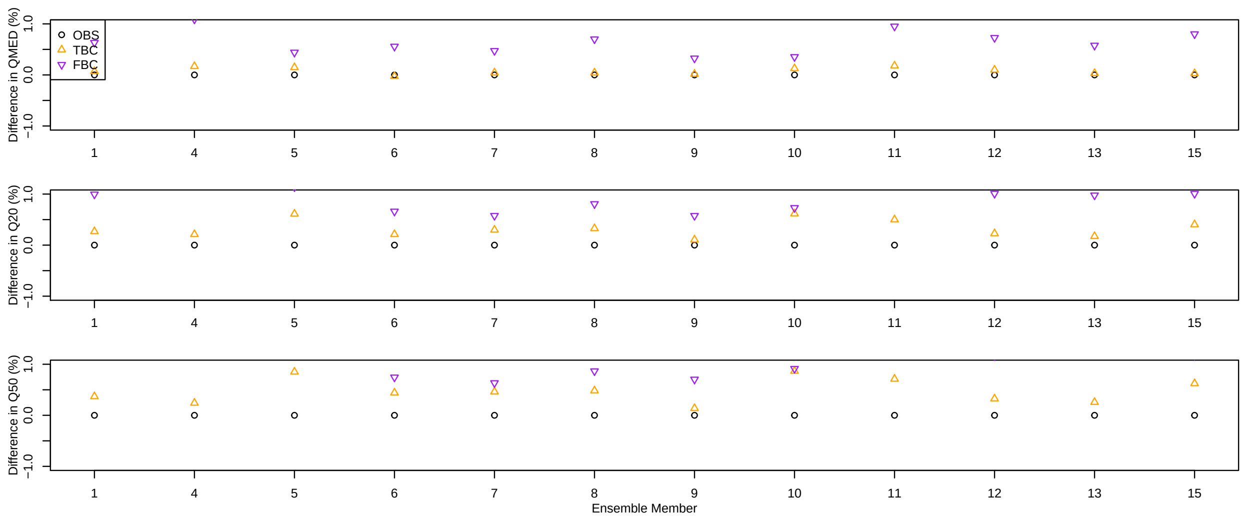


This plot shows correlation between observed, with (TBC) and without (FBC) bias correction, along with CV for the three sets of time-series for ensemble member 01. Additionally is a small histogram of differences is shown, capped around +/-40.

We use 76005 on the Eden as an example. We compare the DM AMAX values for QMED, Q20 and Q50 (fitted to GLO via L-moments methods) across all 12 ensemble members.

```
stn0 <- 39001

tbc_line <- am_glo_TBC %>% dplyr::filter(station==stn0) %>% select(QMED) %>% unlist(.)
fbc_line <- am_glo_FBC %>% dplyr::filter(station==stn0) %>% select(QMED) %>% unlist(.)
obs_line <- am_glo_obs %>% dplyr::filter(station==stn0) %>% select(QMED) %>% unlist(.)

tbc_line20 <- am_glo_TBC %>% dplyr::filter(station==stn0) %>% select(Q20) %>% unlist(.)
fbc_line20 <- am_glo_FBC %>% dplyr::filter(station==stn0) %>% select(Q20) %>% unlist(.)
obs_line20 <- am_glo_obs %>% dplyr::filter(station==stn0) %>% select(Q20) %>% unlist(.)

tbc_line50 <- am_glo_TBC %>% dplyr::filter(station==stn0) %>% select(Q50) %>% unlist(.)
fbc_line50 <- am_glo_FBC %>% dplyr::filter(station==stn0) %>% select(Q50) %>% unlist(.)
obs_line50 <- am_glo_obs %>% dplyr::filter(station==stn0) %>% select(Q50) %>% unlist(.)

# Plot of relative difference in QMED, Q20 and Q50 on average between RCMs

#png("./kingston.png", width=160, height=100, units='mm', res=300, pointsize=11)
par(mar=c(3,3,1,1), mgp=c(2,1,0), mfrow=c(3,1))

plot(1:length(rcm_nos), rep(0,length(rcm_nos)), pch=1, col="black",
     xaxt='n', xlab="", ylab="Difference in QMED (%)", ylim=c(-1,1))
points(1:length(rcm_nos), (tbc_line-obs_line)/obs_line, col="orange", pch=2)
points(1:length(rcm_nos), (fbc_line-obs_line)/obs_line, col="purple", pch=6)
```

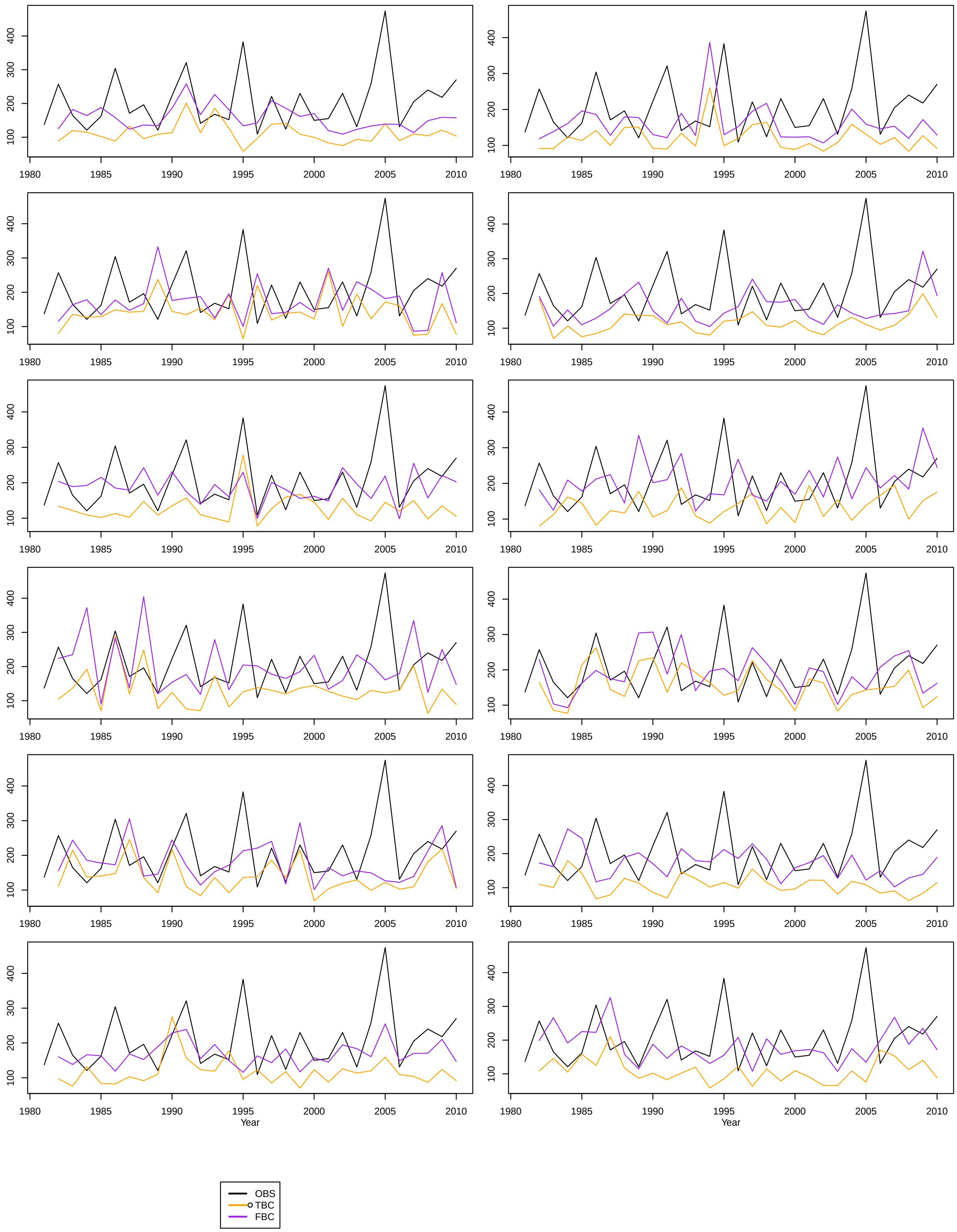
```r
axis(1, at=1:length(rcm_nos), labels=rcm_nos)
legend("topleft", legend=c("OBS", "TBC", "FBC"), col=c("black", "orange", "purple"), pch=c(1,2,6))

plot(1:length(rcm_nos), rep(0,length(rcm_nos)), pch=1, col="black",
     xaxt='n', xlab="", ylab="Difference in Q20 (%)", ylim=c(-1,1))
points(1:length(rcm_nos), (tbc_line20-obs_line20)/obs_line20, col="orange", pch=2)
points(1:length(rcm_nos), (fbc_line20-obs_line20)/obs_line20, col="purple", pch=6)
axis(1, at=1:length(rcm_nos), labels=rcm_nos)

plot(1:length(rcm_nos), rep(0,length(rcm_nos)), pch=1, col="black",
     xaxt='n', xlab="Ensemble Member", ylab="Difference in Q50 (%)", ylim=c(-1,1))
points(1:length(rcm_nos), (tbc_line50-obs_line50)/obs_line50, col="orange", pch=2)
points(1:length(rcm_nos), (fbc_line50-obs_line50)/obs_line50, col="purple", pch=6)
axis(1, at=1:length(rcm_nos), labels=rcm_nos)
```




```r
#dev.off()
```

This plot shows percentage difference in QMED, Q20 and Q50, with observed as a black circle (always zero), with bias correction in orange, and without in purple. This again highlights the negative bias in nearly all the ensemble members, but that there are differences between them. On the whole, the use of bias correction actually increases underestimation compared to mean daily flow.

```r
am_obs1 <- am_summ_obs %>% dplyr::filter(station==76005) %>% select(year, value)
# Comparison of like-for-like time series
par(mar=c(3,3,0.1,0.1), mgp=c(2,1,0), mfrow=c(7,2))
for(i in 1:length(rcm_nos)){
  am_tbc1 <- am_summ_TBC %>% dplyr::filter(station==76005 & rcm==rcm_nos[i]) %>% select(year, value)
  am_fbc1 <- am_summ_FBC %>% dplyr::filter(station==76005 & rcm==rcm_nos[i]) %>% select(year, value)

  plot(am_obs1$year, am_obs1$value, col=1, type='l', xlab=ifelse(i>10,"Year",""),
       ylab="", ylim=range(am_tbc1$value, am_fbc1$value, am_obs1$value, na.rm=T))
  lines(am_tbc1$year, am_tbc1$value, col="orange")
  lines(am_fbc1$year, am_fbc1$value, col="purple")
}

plot(0,0, xlim=c(-1,1), ylim=c(-1,1), xaxt="n", yaxt="n", axes=F, xlab="", ylab="")
legend("center", legend=c("OBS","TBC", "FBC"), col=c("black", "orange", "purple"),
       lwd=2, border=NA)
```


OBS
TBC
FBC
Year

Here are the time series for the twelve different ensemble members for station 76005 (Eden). One can see that on average, the time series match reasonably, the 1995 and 2005 calues are much higher in observed than any value in the modelled values. This may explain the underestimation in QT.

```
par(mar=c(0.5,0.5,0.5,0.5), mgp=c(2,1,0))
uk_outline <- readOGR("C:/Users/adagri/Documents/ResilRiskInds_C/interim_data/uk_outline")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "C:\Users\adagri\Documents\ResilRiskInds_C\interim_data\uk_outline", layer: "uk_outline_1000m
## with 1 features
## It has 71 fields
```

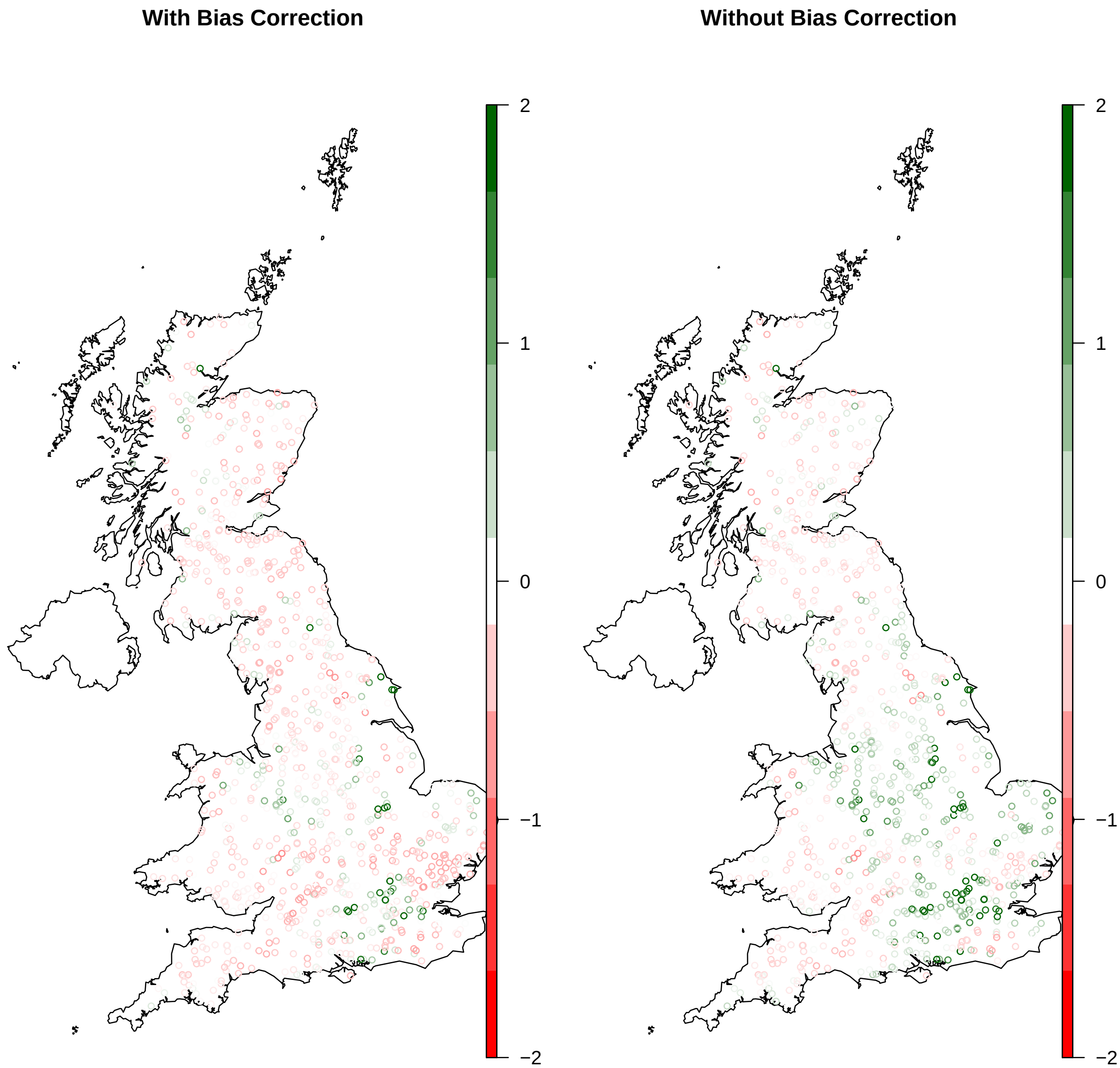
```
qmed_TBC <- am_glo_TBC %>% group_by(station) %>% summarise(qmed_av=mean(QMED))
qmed_FBC <- am_glo_FBC %>% group_by(station) %>% summarise(qmed_av=mean(QMED))
qmed_obs <- am_glo_obs %>% dplyr::select(station, QMED)

posUK <- CATAL %>% dplyr::filter(id %in% qmed_TBC$station) %>% dplyr::select(id, easting, northing)

Qpc <- (qmed_TBC$qmed_av - qmed_obs$QMED)/qmed_obs$QMED#
summary(Qpc)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.933789 -0.360077 -0.211206 -0.089853  0.008117 10.822835
```

```
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2

Qpc2 <- (Qpc + 2)/4

r2g <- colorRamp(c("red", "white", "darkgreen"))
r2g2 <- colorRampPalette(c("red", "white", "darkgreen"))(11)

par(mar=c(1,1,3,1), mfrow=c(1,2), mgp=c(2,1,0))
plot(uk_outline, main="With Bias Correction")
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2, legend.only=T)

Qpc <- (qmed_FBC$qmed_av - qmed_obs$QMED)/qmed_obs$QMED
summary(Qpc)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.90426 -0.20995  0.01057  0.21782  0.36963 16.35157
```

```
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2

Qpc2 <- (Qpc + 2)/4

plot(uk_outline,  main="Without Bias Correction")
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2, legend.only=T, legend.lab="Pe
```




```
##### Q20


qmed_TBC <- am_glo_TBC %>% group_by(station) %>% summarise(qmed_av=mean(Q20))
qmed_FBC <- am_glo_FBC %>% group_by(station) %>% summarise(qmed_av=mean(Q20))
qmed_obs <- am_glo_obs %>% dplyr::select(Q20)

Qpc <- (qmed_TBC$qmed_av - qmed_obs$Q20)/qmed_obs$Q20
summary(Qpc)
```

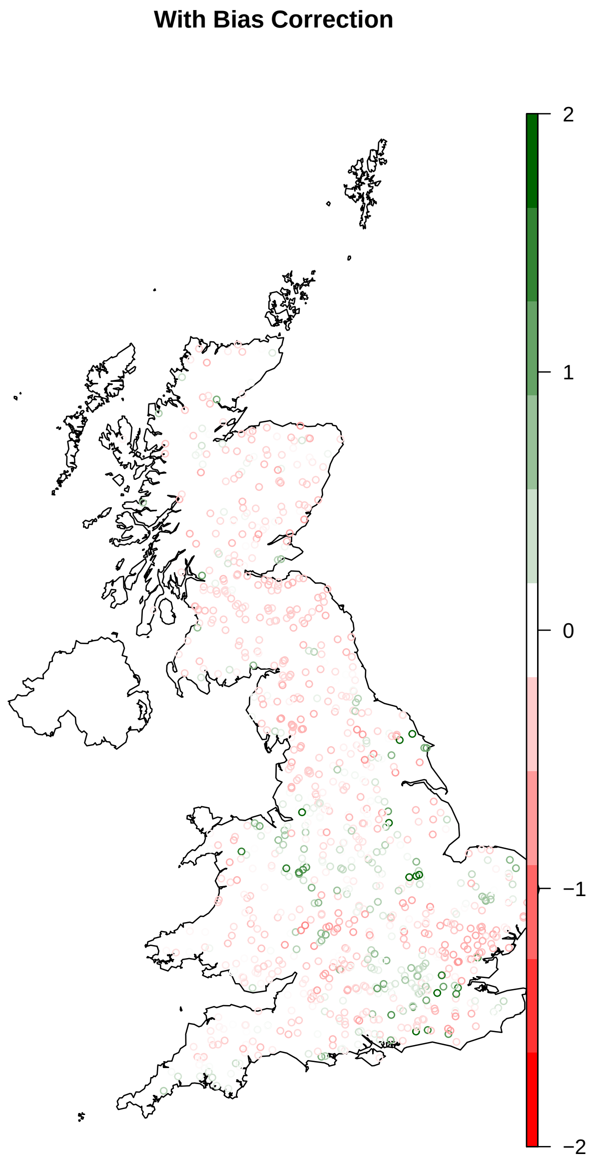
```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.95806 -0.37585 -0.20603 -0.07481  0.07897  8.04328
```

```
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2
```

```r
Qpc2 <- (Qpc + 2)/4

par(mfrow=c(1,2))
plot(uk_outline, main="With Bias Correction")
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2, legend.only=T)

Qpc <- (qmed_FBC$qmed_av - qmed_obs$Q20)/qmed_obs$Q20
summary(Qpc)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.93648 -0.27805 -0.05593  0.15684  0.37621 10.87381
```

```r
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2
Qpc2 <- (Qpc + 2)/4

plot(uk_outline, main="Without Bias Correction")
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2, legend.only=T, legend.lab="Pe
```




```
##### Q50

qmed_TBC <- am_glo_TBC %>% group_by(station) %>% summarise(qmed_av=mean(Q50))
qmed_FBC <- am_glo_FBC %>% group_by(station) %>% summarise(qmed_av=mean(Q50))
qmed_obs <- am_glo_obs %>% dplyr::select(Q50)

Qpc <- (qmed_TBC$qmed_av - qmed_obs$Q50)/qmed_obs$Q50
summary(Qpc)
```

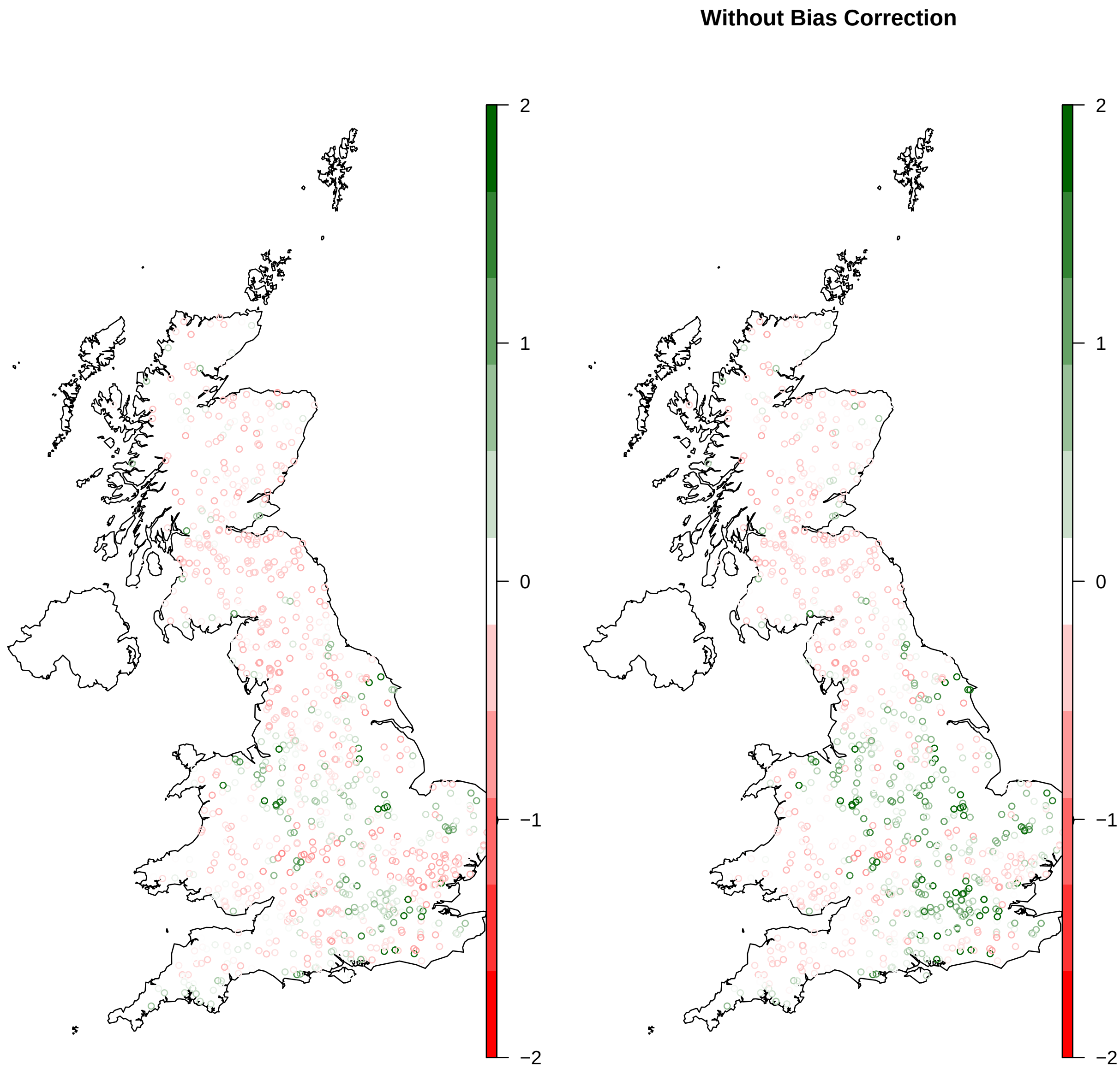
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.96273 -0.38708 -0.17732 -0.02987  0.15766  7.24737
```

```
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2
Qpc2 <- (Qpc + 2)/4
```

```r
par(mfrow=c(1,2))
plot(uk_outline)
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2, legend.only=T)

Qpc <- (qmed_FBC$qmed_av - qmed_obs$Q50)/qmed_obs$Q50
summary(Qpc)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.94286 -0.31031 -0.05623  0.18088  0.43093  9.47444
```

```r
Qpc[Qpc > 2] <- 2
Qpc[Qpc < -2] <- -2
Qpc2 <- (Qpc + 2)/4

plot(uk_outline, main="Without Bias Correction")
points(posUK$easting, posUK$northing, cex=0.7, col=rgb(r2g(Qpc2), maxColorValue=256))
image.plot(Qpc2, add=T, breaks=seq(from=-2,to=2,length.out=12), col=r2g2,
           legend.only=T, legend.lab="Percentage change in Q50")
```




```
# easting and northing
```

Finally, these six maps are presented to look for any spatial patterns in the difference between the two methods of modelling. Overall, the patterns are very similar in the three cases (QMED, Q20 and Q50 as before). One can see a pattern of overestimation around London and other populated areas of GB, and the greater negative bias.

Conclusions

Given this limited investigation, it seems that the bias correction will give less accurate flood frequency estimates compared to observed records.

References

Guillod, B.P., Jones, R.G., Dadson, S.J., Coxon, G., Bussi, G., Freer, J. & Allen, M.R. (2018). A large set of potential past, present and future hydro-meteorological time series for the UK. Hydrology and Earth System Sciences, 22(1), 611-634.