test1

Peter Thejll

05/03/2019

R Markdown

Testing of Cannon's package MBC - straight from the manual

```
## Not run: Example from MBC manual
data(cccma)
set.seed(1) # Univariate quantile mapping
qdm.c <- cccma$gcm.c * 0
qdm.p <- cccma$gcm.p * 0
for (i in seq(ncol(cccma$gcm.c))) {
  fit.qdm <-
    QDM(
      o.c = cccma$rcm.c[, i],
      m.c = cccma$gcm.c[, i],
      m.p = cccma$gcm.p[, i],
      ratio = cccma$ratio.seq[i],
      trace = cccma$trace[i]
    )
  qdm.c[, i] <- fit.qdm$mhat.c
  qdm.p[, i] <- fit.qdm$mhat.p
}
# Multivariate MBCp bias correction
fit.mbcp <-
  MBCp(
    o.c = cccma$rcm.c,
    m.c = cccma$gcm.c,
    m.p = cccma$gcm.p,
    ratio.seq = cccma$ratio.seq,
    trace = cccma$trace
  )
```

```
## 1 0.1362384 2 0.01048991 3 0.001605281 4 0.0004467989 5 0.0001816461 6 9.473764e-05
```

```
mbcp.c <- fit.mbcp$mhat.c
mbcp.p <- fit.mbcp$mhat.p
# Multivariate MBCr bias correction
fit.mbcr <-
  MBCr(
    o.c = cccma$rcm.c,
    m.c = cccma$gcm.c,
    m.p = cccma$gcm.p,
    ratio.seq = cccma$ratio.seq,
    trace = cccma$trace
  )
```

```
## 1 0.1261986 0.1727507 2 0.1353311 0.01811302 3 0.20371 0.002430912 4 0.4616438 0.0003616098 5 0.8442
```

```
mbcr.c <- fit.mbcr$mhat.c
mbcr.p <- fit.mbcr$mhat.p
```

```r
# Multivariate MBCn bias correction
fit.mbcn <-
  MBCn(
    o.c = cccma$rcm.c,
    m.c = cccma$gcm.c,
    m.p = cccma$gcm.p,
    ratio.seq = cccma$ratio.seq,
    trace = cccma$trace
  )
```

```
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
```

```r
mbcn.c <- fit.mbcn$mhat.c
mbcn.p <- fit.mbcn$mhat.p
colnames(mbcn.c) <- colnames(mbcn.p) <- colnames(cccma$rcm.c)
# Correlation matrices (Pearson and Spearman)
# MBCp
dev.new()
par(mfrow = c(2, 2))
plot(
  c(cor(cccma$rcm.c)),
  c(cor(qdm.c)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCp' ,
  main = 'Pearson correlation\nMBCp calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c)), c(cor(mbcp.c)), col = 'red')
plot(
  c(cor(cccma$rcm.p)),
  c(cor(qdm.p)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCp' ,
  main = 'Pearson correlation\nMBCp evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p)), c(cor(mbcp.p)), col = 'red')
plot(
  c(cor(cccma$rcm.c, m = 's')),
  c(cor(qdm.c, m = 's')),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
```

```
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCp' ,
  main = 'Spearman correlation\nMBCp calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c, m = 's')), c(cor(mbcp.c, m = 's')), col = 'red')
plot(
  c(cor(cccma$rcm.p, m = 's')),
  c(cor(qdm.p, m = 's')),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCp' ,
  main = 'Spearman correlation\nMBCp evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p, m = 's')), c(cor(mbcp.p, m = 's')), col = 'red')
# MBCr
dev.new()
par(mfrow = c(2, 2))
plot(
  c(cor(cccma$rcm.c)),
  c(cor(qdm.c)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCr' ,
  main = 'Pearson correlation\nMBCr calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c)), c(cor(mbcr.c)), col = 'blue')
plot(
  c(cor(cccma$rcm.p)),
  c(cor(qdm.p)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCr' ,
  main = 'Pearson correlation\nMBCr evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p)), c(cor(mbcr.p)), col = 'blue')
plot(
```

```
  c(cor(cccma$rcm.c, m = 's')),
  c(cor(qdm.c, m = 's')),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCr' ,
  main = 'Spearman correlation\nMBCr calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c, m = 's')), c(cor(mbcr.c, m = 's')), col = 'blue')
plot(
  c(cor(cccma$rcm.p, m = 's')),
  c(cor(qdm.p, m = 's')),
  col = 'black',
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCr' ,
  main = 'Spearman correlation\nMBCr evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p, m = 's')), c(cor(mbcr.p, m = 's')), col = 'blue')
# MBCn
dev.new()
par(mfrow = c(2, 2))
plot(
  c(cor(cccma$rcm.c)),
  c(cor(qdm.c)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCn' ,
  main = 'Pearson correlation\nMBCn calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c)), c(cor(mbcn.c)), col = 'orange')
plot(
  c(cor(cccma$rcm.p)),
  c(cor(qdm.p)),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCn' ,
```

```
  main = 'Pearson correlation\nMBCn evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p)), c(cor(mbcn.p)), col = 'orange')
plot(
  c(cor(cccma$rcm.c, m = 's')),
  c(cor(qdm.c, m = 's')),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCn' ,
  main = 'Spearman correlation\nMBCn calibration'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.c, m = 's')), c(cor(mbcn.c, m = 's')), col = 'orange')
plot(
  c(cor(cccma$rcm.p, m = 's')),
  c(cor(qdm.p, m = 's')),
  col = 'black' ,
  pch = 19,
  xlim = c(-1, 1),
  ylim = c(-1, 1),
  xlab = 'CanRCM4',
  ylab = 'CanESM2 MBCn' ,
  main = 'Spearman correlation\nMBCn evaluation'
)
abline(0, 1)
grid()
points(c(cor(cccma$rcm.p, m = 's')), c(cor(mbcn.p, m = 's')), col = 'orange') # Pairwise scatterplots
dev.new()
pairs(cccma$gcm.c, main = 'CanESM2 calibration', col = '#0000001A')
dev.new()
pairs(cccma$rcm.c, main = 'CanRCM4 calibration', col = '#0000001A')
dev.new()
pairs(qdm.c, main = 'QDM calibration', col = '#0000001A')
dev.new()
pairs(mbcp.c, main = 'MBCp calibration', col = '#FF00001A')
dev.new()
pairs(mbcr.c, main = 'MBCr calibration', col = '#0000FF1A')
dev.new()

pairs(mbcn.c, main = 'MBCn calibration', col = '#FFA5001A')
# Energy distance skill score relative to univariate
escore.qdm <- escore(cccma$rcm.p, qdm.p, scale.x = TRUE)
escore.mbcp <- escore(cccma$rcm.p, mbcp.p, scale.x = TRUE)
escore.mbcr <- escore(cccma$rcm.p, mbcr.p, scale.x = TRUE)
escore.mbcn <- escore(cccma$rcm.p, mbcn.p, scale.x = TRUE)
cat('ESS (MBCp):', 1 - escore.mbcp / escore.qdm, '\n')
```

```
## ESS (MBCp): 0.7718066
```

```r
cat('ESS (MBCr):', 1 - escore.mbcr / escore.qdm, '\n')
```

```
## ESS (MBCr): 0.7024248
```

```r
cat('ESS (MBCn):', 1 - escore.mbcn / escore.qdm, '\n')
```

```
## ESS (MBCn): 0.9309846
```

```r
## End(Not run)
```