Chapter 5: Prediction of annual runoff in ungauged basins - a EU example

Alberto Viglione

1 Introduction

This Tutorial has been developed by Alberto Viglione to illustrate the regional prediction of annual runoff in ungauged basins (see, McMahon et al., 2013). The idea is to predict the mean annual runoff in Europe based on a dataset of 763 catchments in EU where we have some basic information on runoff, precipitation, temperature and solar radiation.

First of all load the library:

```
library(PUBexamples)
```

Then the data:

```
help(data4chapter5and6)
```

```
data(data4chapter5and6)
head(CatchmentsEU, 15)
```

```
      code                             station                               river     lon     lat   elev   area country
1  6114500                   PONTE DE IONCAIS                   PONTE DE IONCAIS -7.5200 40.6100 330.00  606.0      PT
2  6115500                     M.DA GAMITINHA                     M.DA GAMITINHA -8.4000 38.0700  28.00 2721.0      PT
3  6118010                SAINT-JEAN-BREVELAY                              CLAIE -2.7033 47.8248  99.98  137.0      FR
4  6118015                   LOYAT (PONT D129)                              YVEL -2.3688 47.9938  88.26  315.0      FR
5  6118020       GRAND-FOUGERAY (LA BERNADAISE)                 RUISSEAUX D ARON -1.6908 47.7122  68.00  118.0      FR
6  6118030           PAIMPONT (PONT DU SECRET)                               AFF -2.1438 47.9816 119.01   30.2      FR
7  6118060                              GUENIN                              EVEL -2.9752 47.8999  95.16  316.0      FR
8  6118070                BANNALEC (PONT MEYA)                          STER-GOZ -3.7522 47.9068  85.08   69.7      FR
9  6118150                           TREZILIDE                           GUILLEC -4.0770 48.6150  91.10   43.0      FR
10 6118165                                 RAI                             RISLE  0.5806 48.7479 255.45  149.0      FR
11 6118175                          PLOUGONVEN                            JARLOT -3.8005 48.5656  73.86   44.0      FR
12 6118205               SAINT-OUEN-LA-ROUERIE                          LOISANCE -1.4363 48.4270 103.34   81.5      FR
13 6118210            MONFORT-SUR-MEU (LABBAYE)                               MEU -1.9448 48.1275  83.50  468.0      FR
14 6119010               BERENX (PONT DE BERENX)           BERENX (PONT DE BERENX) -0.8537 43.5078 100.44 2575.0      FR
15 6119020 OLORON-SAINTE-MARIE (OLORON-STE-CROIX) OLORON-SAINTE-MARIE (OLORON-STE-CROIX) -0.5971 43.1877 277.79  488.0      FR
```

```
head(meanQmon, 15)   # mean monthly discharge (m3/s)
```

```
          Jan   Feb   Mar    Apr    May    Jun   Jul   Aug   Sep   Oct   Nov   Dec
6114500 18.26 21.75 16.11  10.46   6.67   3.18  1.03  0.29  0.32  2.06  8.13 15.31
6115500 16.95 22.18 17.91   6.02   3.05   1.03  0.38  0.53  1.49  2.03  6.50 14.21
6118010  3.65  3.43  2.65   2.10   1.60   1.02  0.54  0.33  0.34  0.66  1.39  2.56
6118015  5.48  5.18  3.78   2.42   1.86   0.88  0.42  0.17  0.20  0.71  1.64  3.52
6118020  1.87  1.65  1.15   0.73   0.55   0.18  0.07  0.02  0.04  0.21  0.52  1.27
6118030  0.57  0.54  0.43   0.31   0.25   0.09  0.02  0.01  0.02  0.07  0.18  0.37
6118060  8.25  7.92  5.74   3.74   2.49   1.32  0.57  0.30  0.36  1.05  2.58  5.64
6118070  3.12  3.13  2.28   1.74   1.24   0.83  0.58  0.41  0.40  0.67  1.39  2.33
6118150  1.23  1.28  1.04   0.82   0.62   0.43  0.33  0.27  0.27  0.37  0.58  0.94
6118165  2.34  2.32  1.94   1.42   1.14   0.83  0.73  0.61  0.61  0.90  1.36  2.36
6118175  1.29  1.40  1.14   0.91   0.66   0.44  0.32  0.25  0.24  0.34  0.59  0.98
6118205  1.26  1.29  1.12   0.90   0.77   0.56  0.44  0.36  0.38  0.55  0.78  1.03
6118210  7.35  7.30  5.25   3.39   2.86   1.28  0.82  0.34  0.36  0.99  2.12  4.67
6119010 90.01 89.76 85.34 101.70 123.10 115.00 68.80 42.01 40.87 55.45 76.49 94.18
6119020 17.49 17.53 19.64  27.65  38.77  29.70 14.37  6.96  8.42 13.83 19.95 19.71
```

```
head(meanPmon, 15)   # mean monthly catchment precipitation (mm/d)
```

```
         Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
6114500 4.15 3.73 2.52 2.64 2.18 1.20 0.35 0.32 1.31 3.41 3.93 4.05
6115500 2.50 2.40 1.54 1.61 1.10 0.44 0.04 0.09 0.67 2.15 2.64 3.05
6118010 2.99 2.58 2.14 1.87 1.99 1.39 1.24 1.33 2.00 2.79 3.16 3.09
6118015 2.71 2.34 1.98 1.77 1.97 1.42 1.26 1.31 1.92 2.60 2.96 2.81
6118020 2.34 2.05 1.78 1.61 1.96 1.41 1.29 1.26 1.84 2.34 2.58 2.38
6118030 2.56 2.22 1.89 1.72 1.97 1.44 1.26 1.30 1.88 2.51 2.86 2.66
6118060 3.16 2.71 2.24 1.95 2.02 1.42 1.27 1.39 2.08 2.90 3.30 3.26
6118070 3.96 3.37 2.74 2.38 2.23 1.60 1.47 1.67 2.40 3.45 3.88 4.03
6118150 4.07 3.46 2.77 2.39 2.19 1.61 1.51 1.73   NA 3.46   NA 4.14
6118165 2.32 2.06 1.93 1.68 1.99 1.69 1.68 1.46   NA 2.31   NA 2.51
6118175 3.72 3.17 2.56 2.18 2.05 1.54 1.42 1.61 2.26 3.23   NA 3.86
6118205 2.33 2.05 1.78 1.61 1.92 1.45 1.31 1.27 1.82 2.34   NA 2.45
6118210 2.52 2.18 1.87 1.70 1.97 1.43 1.27 1.29 1.87 2.48   NA 2.61
6119010 3.52 3.23 2.91 3.72 3.58 2.85 1.82 2.19 2.77 3.54 4.00 3.69
6119020 3.79 3.41 3.00 3.82 3.70 2.99 1.83 2.17 2.89 3.93 4.36 3.98
```

```
head(meanTmon, 15)   # mean monthly catchment temperature (deg C)
```

```
          Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
6114500  6.16  7.16  9.29 10.84 14.05 18.40 21.45 21.45 18.80 14.18  9.49  6.76
6115500 10.66 11.47 13.26 14.71 17.44 20.92 23.60 23.86 21.97 18.40 14.18 11.38
6118010  6.23  6.43  8.07  9.90 12.94 15.78 17.74 17.94 16.17 13.16  9.42  6.97
6118015  5.79  6.09  7.87  9.78 12.94 15.87 17.90 18.02 16.06 12.88  8.96  6.45
```

```
6118020  5.40  5.94  7.99 10.11 13.45 16.58 18.67 18.66 16.35 12.85  8.57  5.91
6118030  5.63  5.97  7.81  9.77 12.97 15.94 17.99 18.08 16.03 12.78  8.79  6.26
6118060  6.36  6.49  8.04  9.82 12.80 15.59 17.52 17.77 16.08 13.17  9.53  7.13
6118070  6.63  6.61  7.89  9.41 12.16 14.78 16.65 16.96 15.54 12.91  9.58  7.45
6118150  6.82  6.80  8.02  9.45 12.12 14.68 16.54 16.83 15.43 12.87  9.61  7.60
6118165  3.64  4.25  6.51  8.84 12.29 15.35 17.50 17.57 15.02 11.46  6.99  4.17
6118175  6.58  6.55  7.85  9.38 12.14 14.75 16.66 16.95 15.49 12.86  9.52  7.39
6118205  5.17  5.61  7.60  9.66 12.95 15.99 18.08 18.14 15.92 12.53  8.38  5.75
6118210  5.62  5.98  7.84  9.81 13.03 16.01 18.06 18.14 16.07 12.79  8.79  6.24
6119010  2.59  3.29  5.22  7.15 10.89 14.55 17.36 17.35 14.55 10.66  5.82  3.19
6119020  0.02  0.58  2.37  4.15  8.05 12.00 15.20 15.19 12.08  7.95  3.17  0.66
```

```
head(meanSImon, 15)  # mean monthly catchment SI ratio
```

```
          Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
6114500 0.213 0.240 0.273 0.306 0.333 0.346 0.340 0.317 0.285 0.252 0.221 0.206
6115500 0.221 0.247 0.273 0.303 0.323 0.338 0.332 0.312 0.285 0.257 0.230 0.213
6118010 0.200 0.232 0.269 0.312 0.344 0.362 0.354 0.324 0.287 0.248 0.210 0.191
6118015 0.197 0.231 0.269 0.312 0.347 0.368 0.357 0.325 0.286 0.246 0.210 0.185
6118020 0.201 0.232 0.269 0.312 0.343 0.361 0.353 0.324 0.287 0.249 0.211 0.191
6118030 0.202 0.232 0.269 0.312 0.345 0.363 0.354 0.324 0.286 0.248 0.211 0.186
6118060 0.200 0.231 0.269 0.312 0.345 0.365 0.355 0.324 0.286 0.248 0.211 0.186
6118070 0.200 0.231 0.268 0.312 0.347 0.366 0.356 0.324 0.286 0.247 0.210 0.185
6118150 0.194 0.231 0.269 0.314 0.350 0.369 0.360 0.324 0.287 0.247 0.204 0.185
6118165 0.195 0.231 0.270 0.314 0.349 0.368 0.359 0.326 0.288 0.245 0.205 0.185
6118175 0.193 0.230 0.269 0.314 0.350 0.369 0.360 0.325 0.287 0.247 0.206 0.183
6118205 0.195 0.232 0.269 0.314 0.349 0.367 0.359 0.325 0.287 0.246 0.207 0.185
6118210 0.195 0.231 0.269 0.313 0.346 0.367 0.358 0.325 0.287 0.246 0.211 0.185
6119010 0.196 0.234 0.273 0.314 0.345 0.362 0.354 0.326 0.290 0.248 0.207 0.184
6119020 0.189 0.230 0.274 0.318 0.350 0.367 0.359 0.331 0.292 0.246 0.202 0.176
```

Since the objective of the exercise is to estimate the mean annual runoff, let's calculate it from the monthly values

```
MAQ <- apply(meanQmon, 1, mean)
 summary(MAQ)
```

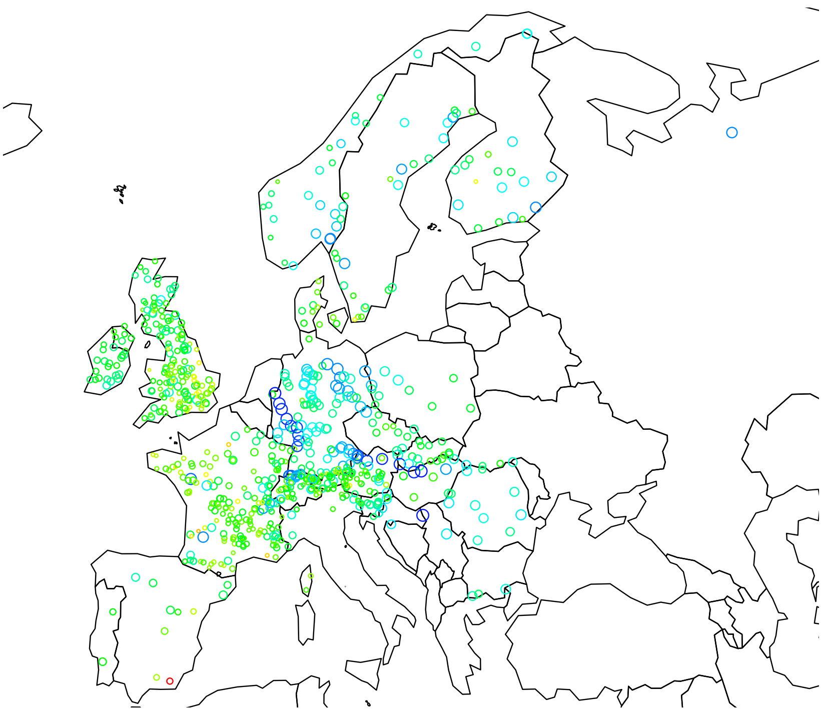
```
   Min.  1st Qu.   Median     Mean  3rd Qu.      Max.
 0.0158   3.0960  10.7000  97.3300  43.0500 2710.0000
```

```
 summary(log10(MAQ))
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-1.8000  0.4908  1.0290  1.1100  1.6340  3.4330
```

Plot the mean annual discharge on a map:

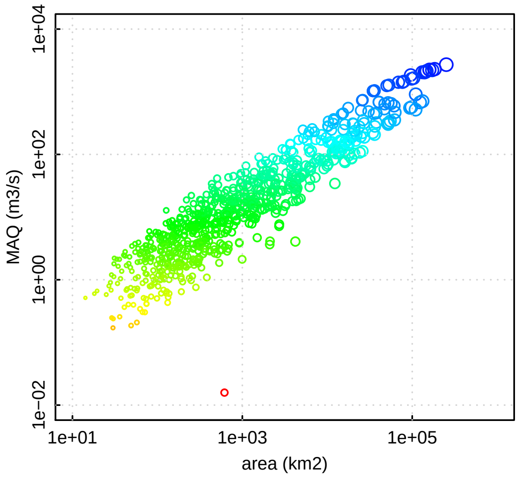
```
library(rworldmap)
newMap <- getMap(resolution="coarse")  # you can use resolution="low", which is better


# for colors
minimo <- min(round(log10(MAQ)*10))
massimo <- max(round(log10(MAQ)*10))
colori <- rainbow(massimo - minimo + 1, start=0, end=.65, alpha=1)
# high values of discharge are blue, low values are red
plot(newMap, xlim=range(CatchmentsEU$lon), ylim=range(CatchmentsEU$lat))
 points(CatchmentsEU$lon, CatchmentsEU$lat, pch=1,
        cex=0.3*log10(CatchmentsEU$area),
        col=colori[round(log10(MAQ)*10) - minimo + 1])
```



2 Catchment area as explanatory variable

Of course, the mean annual discharge (in m^3/s) has to do with area:

```
A <- CatchmentsEU$area
plot(A, MAQ, xlab="area (km2)", ylab="MAQ (m3/s)",
     log="xy", xlim=c(10, 1e6), ylim=c(1e-2, 1e4),
     cex=0.3*log10(A),
     col=colori[round(log10(MAQ)*10) - minimo + 1])
 grid()
```



With the only exception of the southern spanish catchment. Therefore a natural way of estimating the mean annual discharge is to relate it to area through a log-linear regression:

```
regr01 <- lm(log(MAQ) ~ log(A))
 summary(regr01)
```

```
Call:
lm(formula = log(MAQ) ~ log(A))

Residuals:
    Min      1Q  Median      3Q     Max
-6.4571 -0.4849  0.0247  0.5648  1.6417

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.39770    0.09470  -35.88   <2e-16 ***
log(A)       0.88882    0.01353   65.69   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7579 on 761 degrees of freedom
Multiple R-squared:  0.8501,	Adjusted R-squared:  0.8499
F-statistic:  4316 on 1 and 761 DF,  p-value: < 2.2e-16
```

```
exp(regr01$coefficients[1])
```

```
(Intercept)
 0.03345024
```

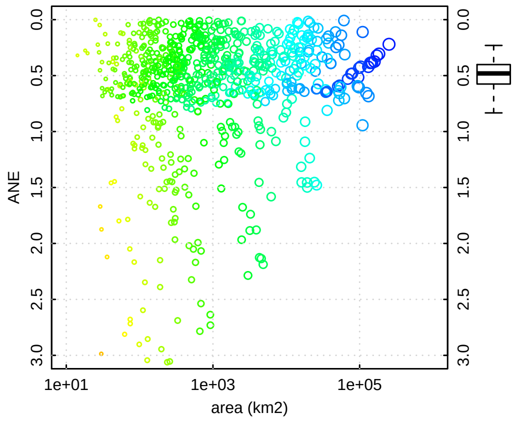
```
# MAQ ~ 0.033*A^0.89


# add regression to the previous plot
curve(0.033*x^0.89, add=TRUE)
```

which has a R^2 greater than 0.8... Our work is done!!!

But wait, what is the error of prediction at single locations, e.g., measured as Absolute Normalise Error (ANE):

```
regMAQ01 <- 0.033*A^0.89
ANE01 <- abs(regMAQ01 - MAQ)/MAQ


boxplot20 <- function(m, ...){
 # m has to be a data.frame or list
 bp <- boxplot(m, plot=FALSE)
 bp$stats <- sapply(m, function(x)
                   quantile(x, c(0.2,0.4, 0.5, 0.6, 0.8), na.rm=TRUE))
 bxp(bp, outline=FALSE, ...)
}
```

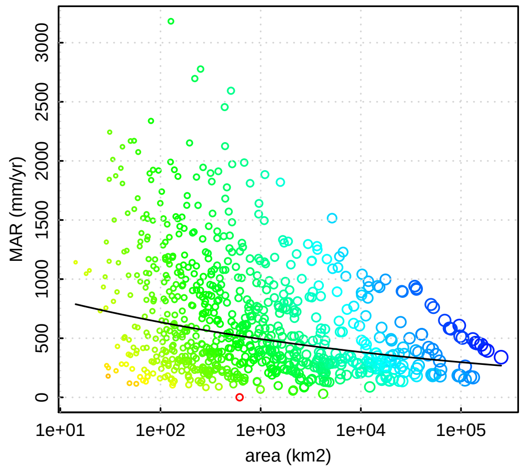
```
layout(matrix(1:2, nrow=1), widths=c(5,1))
plot(A, ANE01, xlab="area (km2)", ylab="ANE",
     log="x", xlim=c(10, 1e6), ylim=c(3, 0),
     cex=0.3*log10(A),
     col=colori[round(log10(MAQ)*10) - minimo + 1])
 grid()
 axis(4)
par(mar=c(3,0,2,0)+0.03)
boxplot20(as.data.frame(ANE01), ylim=c(3, 0), axes=FALSE)
```



The boxplot has been added to the right to be compared with the PUB book assessment Figure 5.27 at page 99. An error of 50% is common (it's the median).

But wait, does area explain everything? What if I consider runoff in mm/yr instead of m^3/s?

```
MAR <- 365.25*24*3.6*MAQ/A  # from m3/s to mm/yr
regr02 <- lm(log(MAR) ~ log(A))
 summary(regr02)

Call:
lm(formula = log(MAR) ~ log(A))

Residuals:
    Min      1Q  Median      3Q     Max
-6.4571 -0.4849  0.0247  0.5648  1.6417

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.96187    0.09470  73.515  < 2e-16 ***
log(A)      -0.11118    0.01353  -8.217 8.93e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7579 on 761 degrees of freedom
Multiple R-squared:  0.0815,	Adjusted R-squared:  0.08029
F-statistic: 67.53 on 1 and 761 DF,  p-value: 8.926e-16

 exp(regr02$coefficients[1])

(Intercept)
   1055.609

 # MAR ~ 1055*A^-0.11



 plot(A, MAR, xlab="area (km2)", ylab="MAR (mm/yr)",
      log="x",
      cex=0.3*log10(A),
      col=colori[round(log10(MAQ)*10) - minimo + 1])
 grid()
curve(1055*x^-0.11, add=TRUE)
```



Now R^2 is very low, we may do better than that.

3 Mean annual precipitation as explanatory variable

Get the mean annual precipitation from the monthly values:

```
MAP <- 365.25*apply(meanPmon, 1, mean, na.rm=TRUE)  # mm/yr
 summary(MAP)
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
404.8   703.5   909.5   979.1  1205.0  3157.0 
```

An additive regression between MAP and MAR may make sense:

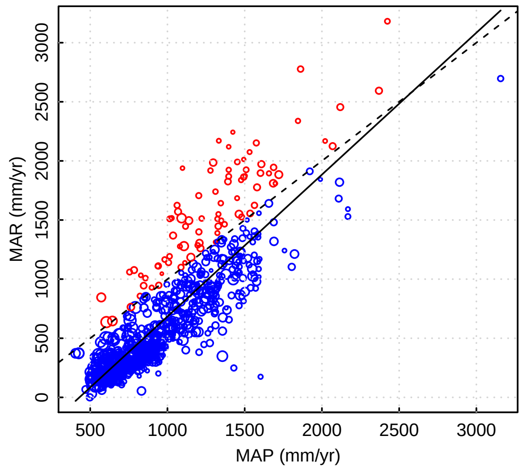
```
regr03 <- lm(MAR ~ MAP)
 summary(regr03)
```

```
Call:
lm(formula = MAR ~ MAP)

Residuals:
     Min       1Q   Median       3Q      Max 
-1237.39  -144.63   -42.63    99.16  1135.42 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -516.45294   28.09114  -18.39   <2e-16 ***
MAP            1.20300    0.02701   44.55   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 261.9 on 761 degrees of freedom
Multiple R-squared:  0.7228,	Adjusted R-squared:  0.7224 
F-statistic:  1984 on 1 and 761 DF,  p-value: < 2.2e-16
```

```
# MAR ~ 1.2*MAP - 516  (mm/yr)
```

Which has a decent R^2 of more than 0.7.

```
plot(MAP, MAR, xlab="MAP (mm/yr)", ylab="MAR (mm/yr)",
      cex=0.3*log10(A),
      col=c("blue","red")[(MAR > MAP) + 1])
 grid()
curve(1.2*x - 516, add=TRUE)
abline(0, 1, lty=2)
```



Notice that the dashed line is a limit above which no point should lie, because that would mean that there is more runoff that rainfall!!! Ops, why are there catchments like that? Where are these catchments?

```
plot(newMap, xlim=range(CatchmentsEU$lon), ylim=range(CatchmentsEU$lat))
 points(CatchmentsEU$lon, CatchmentsEU$lat, pch=1,
        cex=0.3*log10(CatchmentsEU$area),
        col=c("blue","red")[(MAR > MAP) + 1])
```

Many are in the mountains and many at high latitudes... Maybe snow is one explanation: it is difficult to measure precipitation correctly where it snows. Moreover, notice that precipitation data where interpolated spatially from maps produced from ECA data whose density is not homogeneus in space.

Anyway, let's think statistically, we do not care now of non plausible MAP-MAR relationships. Let's try some other regressions, e.g.:

```
regr04 <- lm(MAR ~ MAP + A)
 summary(regr04)
```

```
Call:
lm(formula = MAR ~ MAP + A)

Residuals:
     Min       1Q   Median       3Q      Max
-1237.40  -143.91   -42.84    98.26  1134.18

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.126e+02  2.920e+01 -17.552   <2e-16 ***
MAP          1.201e+00  2.746e-02  43.719   <2e-16 ***
A           -2.060e-04  4.216e-04  -0.489    0.625
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 262 on 760 degrees of freedom
Multiple R-squared:  0.7229,	Adjusted R-squared:  0.7222
F-statistic: 991.3 on 2 and 760 DF,  p-value: < 2.2e-16
```

```
regr05 <- lm(log(MAR) ~ log(MAP))
 summary(regr05)
```

```
Call:
lm(formula = log(MAR) ~ log(MAP))

Residuals:
    Min      1Q  Median      3Q     Max
-5.2351 -0.2207 -0.0331  0.2363  1.4466

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.94491    0.31632  -21.95   <2e-16 ***
log(MAP)     1.92786    0.04627   41.66   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4366 on 761 degrees of freedom
Multiple R-squared:  0.6952,	Adjusted R-squared:  0.6948
F-statistic:  1736 on 1 and 761 DF,  p-value: < 2.2e-16
```

```
regr06 <- lm(log(MAR) ~ log(MAP) + log(A))
 summary(regr06)
```

```
Call:
lm(formula = log(MAR) ~ log(MAP) + log(A))

Residuals:
    Min      1Q  Median      3Q     Max
-5.2339 -0.2209 -0.0332  0.2341  1.4463
```

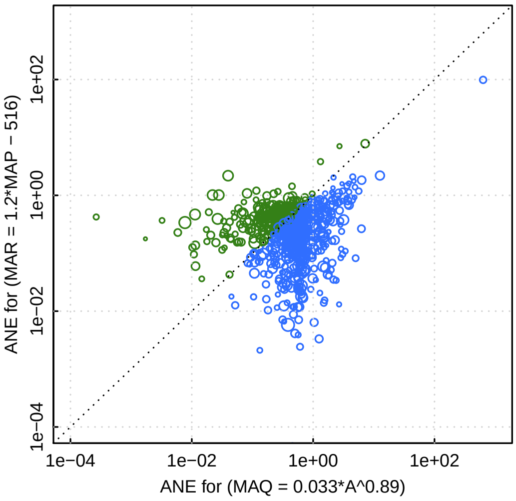
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.960360   0.360069  -19.33   <2e-16 ***
log(MAP)     1.929392   0.049323   39.12   <2e-16 ***
log(A)       0.000748   0.008307    0.09    0.928
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4369 on 760 degrees of freedom
Multiple R-squared:  0.6952,	Adjusted R-squared:  0.6944
F-statistic: 866.7 on 2 and 760 DF,  p-value: < 2.2e-16
```

Once we reason in terms of mm/yr, catchment area does not seem to be useful anymore (at least considering linear relationships).

What is the error of prediction at single locations?

```
regMAR03 <- 1.2*MAP - 516
ANE03 <- abs(regMAR03 - MAR)/MAR
```

```
plot(ANE01, ANE03, xlab="ANE for (MAQ = 0.033*A^0.89)", ylab="ANE for (MAR = 1.2*MAP - 516)",
     log="xy", xlim=c(1e-4,1e3), ylim=c(1e-4,1e3),
     cex=0.3*log10(A),
     col=c("#306EFF", "#348017")[(ANE03 > ANE01) + 1])
 abline(0, 1, lty=3)
 grid()
```



It seems that most of the cases (blue) the relationship with MAP beats the one with A. PS. for deciding colors I normally use this website.

```
mean(ANE01)
```

```
[1] 1.548154
```

```
mean(ANE03)
```

```
[1] 0.4836043
```

```
sum((ANE01 > ANE03))/length(ANE03)
```

```
[1] 0.7195282
```

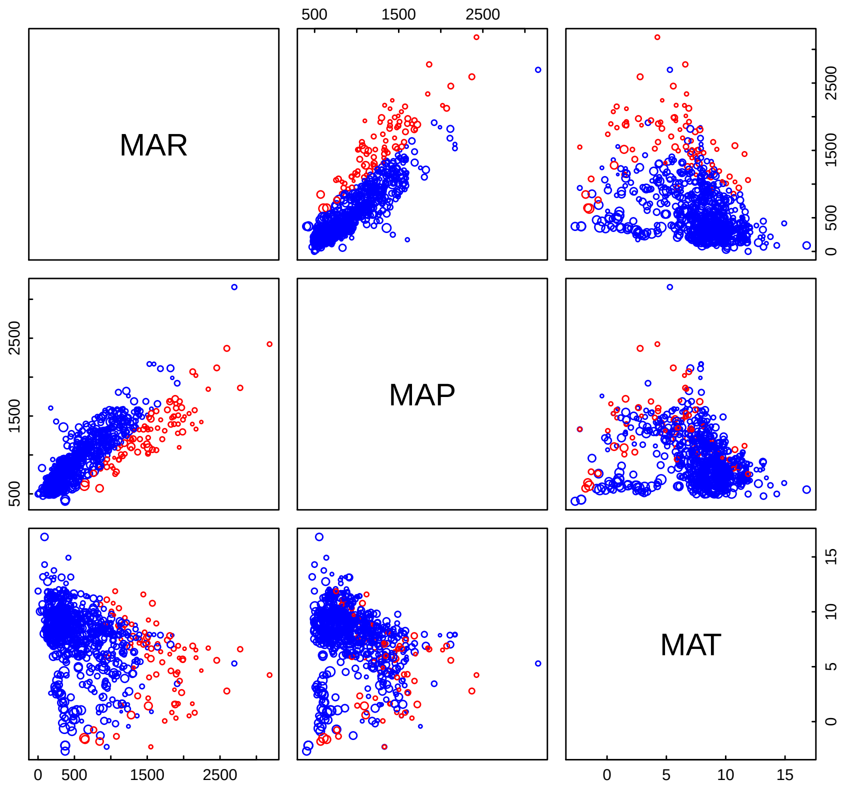
More than 70% of the times the relationship with MAP beats the one with A.

4 Mean annual temperature as explanatory variable

We have information also on temperature which is related to evaporation and therefore phisically meaningful as a covariate for explaning runoff. Moreover, if snow is responsible of some of the data errors, temperature may capture this as well.

```
MAT <- apply(meanTmon, 1, mean, na.rm=TRUE)  # degC
 summary(MAT)

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-2.688   6.269   8.182   7.541   9.525  16.820
```

```
plot(data.frame(MAR, MAP, MAT),
     col=c("blue","red")[(MAR > MAP) + 1],
     cex=0.3*log10(A))
```



Mmmm... also places with high temperature have the problem MAR > MAP.

Anyway, let's try the following:

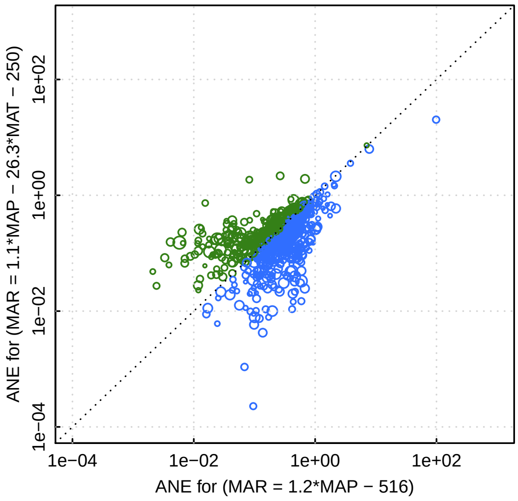
```
regr07 <- lm(MAR ~ MAP + MAT)
 summary(regr07)

Call:
lm(formula = MAR ~ MAP + MAT)

Residuals:
     Min       1Q   Median       3Q      Max
-1324.48  -135.42   -33.13    73.07  1097.74

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -250.2012    41.3844  -6.046 2.33e-09 ***
MAP            1.1338     0.0271  41.834  < 2e-16 ***
MAT          -26.3159     3.1107  -8.460  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 250.5 on 760 degrees of freedom
Multiple R-squared:  0.7467,	Adjusted R-squared:  0.746
F-statistic:  1120 on 2 and 760 DF,  p-value: < 2.2e-16

 # MAR ~ 1.1*MAP - 26.3*MAT - 250
```

which is slightly better that the regression with MAP alone, in terms of R^2.

What about the error of prediction at single locations?

```
regMAR07 <- 1.1*MAP - 26.3*MAT - 250
ANE07 <- abs(regMAR07 - MAR)/MAR
```

```
plot(ANE03, ANE07, xlab="ANE for (MAR = 1.2*MAP - 516)", ylab="ANE for (MAR = 1.1*MAP - 26.3*MAT - 250)",
     log="xy", xlim=c(1e-4,1e3), ylim=c(1e-4,1e3),
     cex=0.3*log10(A),
     col=c("#306EFF", "#348017")[(ANE07 > ANE03) + 1])
 abline(0, 1, lty=3)
 grid()
```



This time it is harder to say which relationship is best.

```
mean(ANE03)
```

```
[1] 0.4836043
```

```
mean(ANE07)
```

```
[1] 0.3128372
```

```
sum((ANE03 > ANE07))/length(ANE07)
```

```
[1] 0.5858453
```

Almost 60% of the times the relationship with MAP and MAT beats the one with MAP only.

5 Budyko

Hydrologists tell us that reasoning in terms of aridity index (potential evaporation over precipitation) and the Budyko diagram helps. Let's try.

In order to calculate the aridity index I need the catchment long term potential evaporation. The data include the SI ratio data for the catchments for every month. As in Parajka et al. (2003), the Blaney-Criddle method modified by Schroedter (1985) can be used to calculate the potential evapotranspiration, i.e., $\mathrm{EP} = -1.55+0.96\cdot(8.128+0.457\cdot\mathrm{T})\cdot\mathrm{SI}$ with the constrain that $\mathrm{EP} \geq 0$.

```
meanEPmon <- -1.55 + 0.96*(8.128 + 0.457*meanTmon)*meanSImon
 meanEPmon[meanEPmon < 0] <- 0                        # mean monthly potential evapotranspiration (mm/d)
PET <- 365.25*apply(meanEPmon, 1, mean, na.rm=TRUE)  # mm/yr
 summary(PET)
```

```
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
267.1   598.0   654.9   642.0   699.9   999.2
```

Let's calculate the aridity index and other interesting nondimensional coefficients:

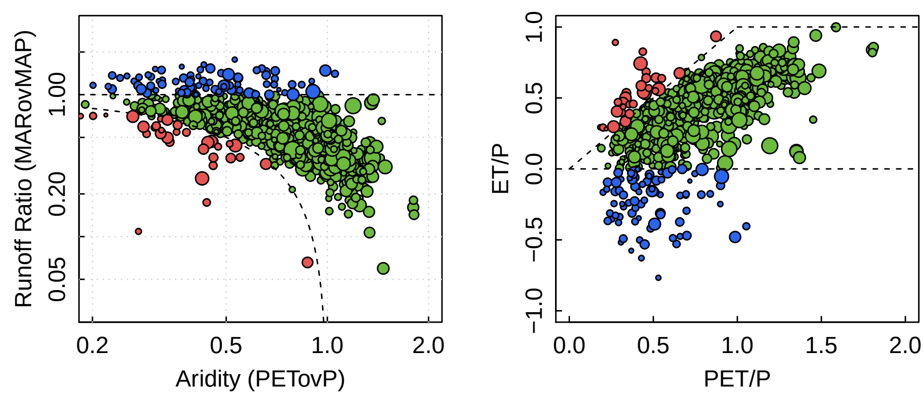
```
PETovP <- PET/MAP   # aridity index
 summary(PETovP)

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.1845  0.4976  0.7366  0.7517  0.9802  1.8100

MARovMAP <- MAR/MAP  # runoff ratio
 summary(MARovMAP)

    Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
0.001631 0.394400 0.552400 0.614200 0.781000 1.767000

ETovP <- (MAP - MAR)/MAP  # actual evaporation over precipitation
 summary(ETovP)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.7671  0.2190  0.4476  0.3858  0.6056  0.9984
```

and index some non phisically plausible sites:

```
MARgrMAP <- (MAR > MAP)  # runoff is greater than rainfall
ETgrPET <- ((MAP - MAR) > PET)  # actual evaporation greater than the potential one
```

And plot some interesting graph:

```
layout(matrix(1:2, ncol=2, byrow=TRUE))
# Figure 4 in Peel et al. (2010)
plot(c(.2,2), c(.03,3), type="n", log="xy",
     xlab="Aridity (PETovP)", ylab="Runoff Ratio (MARovMAP)")
 grid(equilogs=FALSE)
points(PETovP[!(MARgrMAP|ETgrPET)], MARovMAP[!(MARgrMAP|ETgrPET)],
       pch=21, bg="#6CBB3C", cex=0.3*log10(A[!(MARgrMAP|ETgrPET)]))
 abline(h=1, lty=2)
 curve(1 - x, add=TRUE, n=10001, lty=2)
points(PETovP[MARgrMAP], MARovMAP[MARgrMAP],
       pch=21, bg="#2B65EC",
       cex=0.3*log10(A[MARgrMAP]))
points(PETovP[ETgrPET], MARovMAP[ETgrPET],
       pch=21, bg="#E55451",
       cex=0.3*log10(A[ETgrPET]))
# Budyko
plot(PETovP[!(MARgrMAP|ETgrPET)], ETovP[!(MARgrMAP|ETgrPET)],
     xlim=c(0,2), ylim=c(-1,1), xlab="PET/P", ylab="ET/P",
     pch=21, bg="#6CBB3C", cex=0.3*log10(A[!(MARgrMAP|ETgrPET)]))
 segments(x0=c(0,1), x1=c(1,4), y0=c(0,1), y1=c(1,1), lty=2)
 segments(x0=0, x1=4, y0=0, lty=2)
points(PETovP[MARgrMAP], ETovP[MARgrMAP],
       pch=21, bg="#2B65EC",
       cex=0.3*log10(A[MARgrMAP]))
points(PETovP[ETgrPET], ETovP[ETgrPET],
       pch=21, bg="#E55451",
       cex=0.3*log10(A[ETgrPET]))
```



Which on a map look like this:

```
plot(newMap, xlim=range(CatchmentsEU$lon), ylim=range(CatchmentsEU$lat))
 points(CatchmentsEU$lon[!(MARgrMAP|ETgrPET)], CatchmentsEU$lat[!(MARgrMAP|ETgrPET)],
        pch=21, bg="#6CBB3C", cex=0.3*log10(A[!(MARgrMAP|ETgrPET)]))
 points(CatchmentsEU$lon[MARgrMAP], CatchmentsEU$lat[MARgrMAP],
       pch=21, bg="#2B65EC", cex=0.3*log10(A[MARgrMAP]))
 points(CatchmentsEU$lon[ETgrPET], CatchmentsEU$lat[ETgrPET],
       pch=21, bg="#E55451", cex=0.3*log10(A[ETgrPET]))
```

Interpretations?

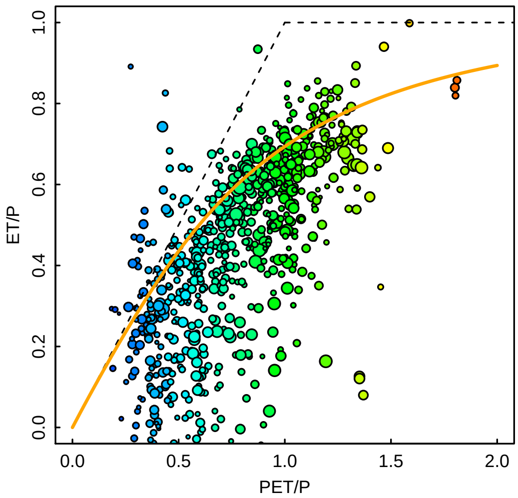
5.1 Non parametric Budyko

Anyway, how well would the runoff be estimated by the original non parametric Budyko formula?

$$F(\varphi) = \left\{\varphi[1 - \exp(-\varphi)] \tanh(\varphi^{-1})\right\}^{0.5}$$

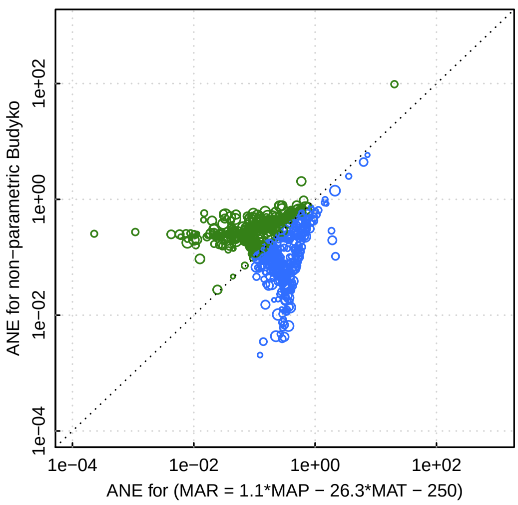
where φ is the aridity index (aka `PETovP`).

```
# I use colors that reflect the aridity
colori <- rev(rainbow(20, start=0, end=.65, alpha=1))
plot(PETovP, ETovP,
     xlim=c(0,2), ylim=c(0,1), xlab="PET/P", ylab="ET/P",
     pch=21, bg=colori[round(10*PETovP)], cex=0.3*log10(A))
 segments(x0=c(0,1), x1=c(1,4), y0=c(0,1), y1=c(1,1), lty=2)
 curve(sqrt(x*(1 - exp(-x))*tanh(1/x)), add=TRUE, lwd=2, col="#FFA500")
```




```
bdkETovP01 <- sqrt(PETovP*(1 - exp(-PETovP))*tanh(1/PETovP))  # estimated EP/P
bdkMAR01 <- MAP - bdkETovP01*MAP
ANEbdk01 <- abs(bdkMAR01 - MAR)/MAR
```

```
plot(ANE07, ANEbdk01, xlab="ANE for (MAR = 1.1*MAP - 26.3*MAT - 250)", ylab="ANE for non-parametric Budyko",
     log="xy", xlim=c(1e-4,1e3), ylim=c(1e-4,1e3),
     cex=0.3*log10(A),
     col=c("#306EFF", "#348017")[(ANEbdk01 > ANE07) + 1])
 abline(0, 1, lty=3)
 grid()
```



This time it is harder to say which relationship is best.

```
mean(ANE07)
```

```
[1] 0.3128372
```

```
mean(ANEbdk01)
```

```
[1] 0.4217168
```

```
sum((ANE07 > ANEbdk01))/length(ANE07)
```

```
[1] 0.3984273
```

About 40% of the times the non-parametric Budyko outperforms the parametric (fitted) regression with MAP and MAT. Not bad. Where does it do this?

```
plot(newMap, xlim=range(CatchmentsEU$lon), ylim=range(CatchmentsEU$lat))
 points(CatchmentsEU$lon, CatchmentsEU$lat, pch=1,
        cex=0.3*log10(CatchmentsEU$area),
        col=c("#306EFF", "#348017")[(ANEbdk01 > ANE07) + 1])
```

Motivations?

5.2 Parametric Budyko

How well would the runoff be estimated by the parametric generalised Turk-Pike formula?

$$F(\varphi) = \left[1 + \varphi^{-\nu}\right]^{-1/\nu}$$

Find ν minimising the sum of square-residuals calculated as:

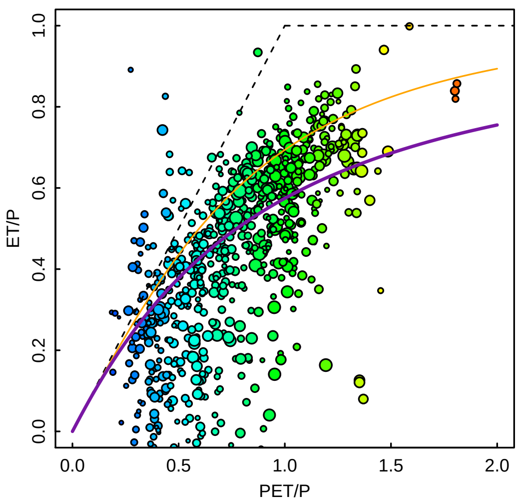
```
sumAbsRes <- function (nu, aridity, epOVERp) {
 estim <- (1 + aridity^(-nu))^(-1/nu)
 output <- sum(abs(estim - epOVERp))
 return(output)
}
```

```
optimize(f=sumAbsRes, interval=c(.1, 10), aridity=PETovP, epOVERp=ETovP)
```

```
$minimum
[1] 1.250115

$objective
[1] 131.697
```

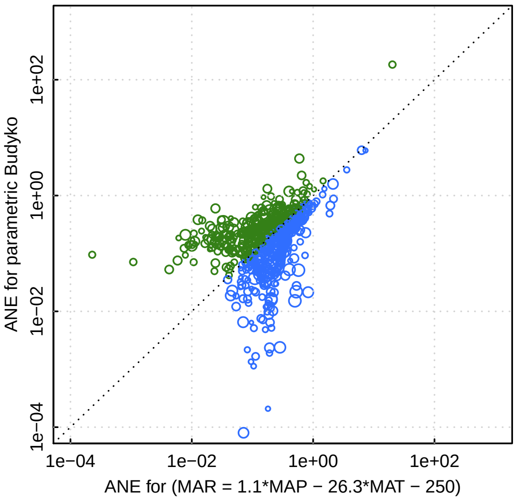
```
# nu = 1.25
```

```
plot(PETovP, ETovP,
     xlim=c(0,2), ylim=c(0,1), xlab="PET/P", ylab="ET/P",
     pch=21, bg=colori[round(10*PETovP)], cex=0.3*log10(A))
 segments(x0=c(0,1), x1=c(1,4), y0=c(0,1), y1=c(1,1), lty=2)
 curve(sqrt(x*(1 - exp(-x))*tanh(1/x)), add=TRUE, lwd=1, col="#FFA500")
 curve((1 + x^(-1.25))^(-1/1.25), add=TRUE, lwd=2, col="#7917A3")
```



```
bdkETovP02 <- (1 + PETovP^(-1.25))^(-1/1.25)  # estimated EP/P
bdkMAR02 <- MAP - bdkETovP02*MAP
ANEbdk02 <- abs(bdkMAR02 - MAR)/MAR
```

```
plot(ANE07, ANEbdk02, xlab="ANE for (MAR = 1.1*MAP - 26.3*MAT - 250)", ylab="ANE for parametric Budyko",
     log="xy", xlim=c(1e-4,1e3), ylim=c(1e-4,1e3),
     cex=0.3*log10(A),
     col=c("#306EFF", "#348017")[(ANEbdk02 > ANE07) + 1])
 abline(0, 1, lty=3)
 grid()
```



This time it is harder to say which relationship is best.

```
mean(ANE07)
```

```
[1] 0.3128372
```

```
mean(ANEbdk02)
```

```
[1] 0.5361023
```

```
sum((ANE07 > ANEbdk02))/length(ANE07)
```

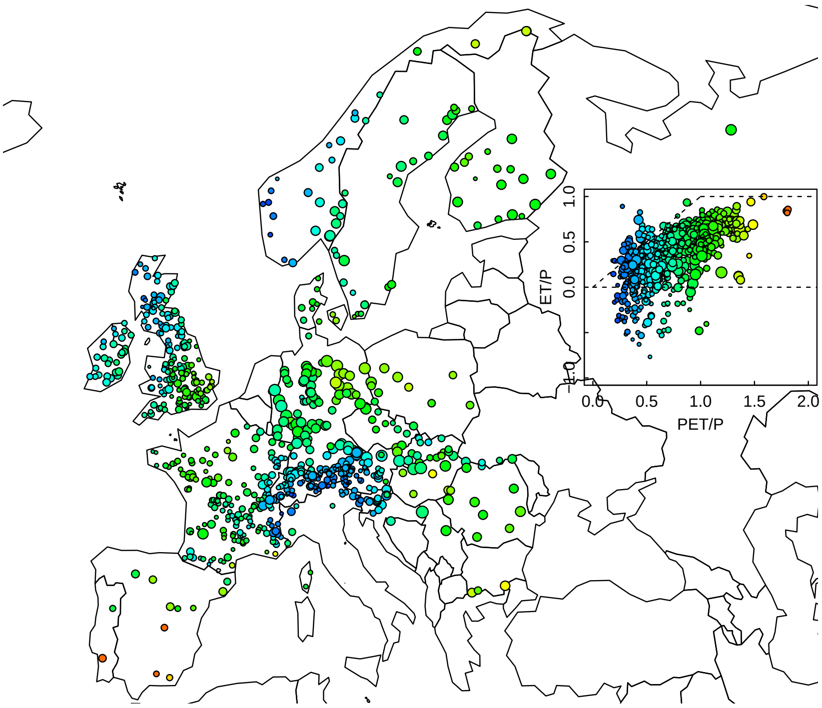
```
[1] 0.4823067
```

Almost 50% of the times the parametric Budyko outperforms the parametric (fitted) regression with MAP and MAT. Still the regression is better. Notice however that the regression has 3 free parameters while the Budyko has one. In your opinion, what should we do next?

6 Compare to the PUB book assessment

To reason about results, I identify catchment in space througn their aridity:

```
library(Hmisc)
# colors related to aridity index
#colori <- rev(rainbow(20, start=0, end=.65, alpha=1))
plot(newMap, xlim=range(CatchmentsEU$lon), ylim=range(CatchmentsEU$lat))
 points(CatchmentsEU$lon, CatchmentsEU$lat, pch=21,
        cex=0.3*log10(CatchmentsEU$area),
        bg=colori[round(10*PETovP)])
subplot(   # needs package 'Hmisc'
 {plot(PETovP, ETovP,
      xlim=c(0,2), ylim=c(-1,1), xlab="PET/P", ylab="ET/P",
      pch=21, bg=colori[round(10*PETovP)], cex=0.3*log10(A))
 segments(x0=c(0,1), x1=c(1,4), y0=c(0,1), y1=c(1,1), lty=2)
 segments(x0=0, x1=4, y0=0, lty=2)},
 c(33,53), c(52,62)
)
```



In the Level 2 Assessment of the PUB book (Blöschl et al., 2013) in Chapter 5 the normalised error and the absolute normalised error in the estimation of annual runoff is calculated.

```
NE07 <- (regMAR07 - MAR)/MAR
ANE07 <- abs(NE07)
NEbdk01 <- (bdkMAR01 - MAR)/MAR
ANEbdk01 <- abs(NEbdk01)
tabella <- data.frame(CatchmentsEU[,c("code", "area", "elev")], temp=MAT, aridity=PETovP,
                      NEregr=round(NE07, 3), ANEregr=round(ANE07, 3),
                      NEbudyko=round(NEbdk01, 3), ANEbudyko=round(ANEbdk01, 3))
 head(tabella, 15)
```

```
      code   area   elev     temp  aridity NEregr ANEregr NEbudyko ANEbudyko
1  6114500  606.0 330.00 13.16917 0.9323429 -0.108   0.108   -0.331     0.331
2  6115500 2721.0  28.00 16.82083 1.8007552 -1.920   1.920   -0.197     0.197
3  6118010  137.0  99.98 11.72917 0.9634795 -0.149   0.149   -0.336     0.336
4  6118015  315.0  88.26 11.55083 1.0160202  0.300   0.300    0.045     0.045
5  6118020  118.0  68.00 11.70667 1.1278134  0.124   0.124   -0.002     0.002
6  6118030   30.2 119.01 11.50167 1.0447785  0.044   0.044   -0.138     0.138
7  6118060  316.0  95.16 11.69167 0.9209745  0.112   0.112   -0.148     0.148
8  6118070   69.7  85.08 11.38083 0.7516084 -0.179   0.179   -0.391     0.391
9  6118150   43.0  91.10 11.39750 0.7619614  0.096   0.096   -0.187     0.187
10 6118165  149.0 255.45 10.29917 1.0137340 -0.084   0.084   -0.261     0.261
11 6118175   44.0  73.86 11.34333 0.8283659 -0.101   0.101   -0.327     0.327
12 6118205   81.5 103.34 11.31500 1.1380359 -0.360   0.360   -0.421     0.421
13 6118210  468.0  83.50 11.53167 1.1017651  0.069   0.069   -0.072     0.072
14 6119010 2575.0 100.44  9.38500 0.5952235 -0.233   0.233   -0.425     0.425
15 6119020  488.0 277.79  6.78500 0.4776441 -0.281   0.281   -0.441     0.441
```

```
aridity_class <- cut(tabella$aridity, breaks=c(-Inf,0.4,0.6,0.8,1,2,Inf))
temp_class <- cut(tabella$temp,  breaks=c(-Inf,3,6,8,10,12,Inf))
elev_class <- cut(tabella$elev, breaks=c(0,300,600,900,1200,1500,Inf))
area_class <- cut(tabella$area, breaks=c(0,50,100,500,1000,5000,Inf))
```

```
add_points <- function(performance="ANE", variable="area", classes, table) {
 # to add points in a nice way
 for (j in 1:length(levels(classes))) {
  dummy <- table[as.numeric(classes) == j,]
  perf <- dummy[, performance]
  stratif <- dummy[, variable]
  if (variable == "area") stratif <- log(stratif)
  if (length(stratif) > 0) {
   if (length(stratif) == 1) {
    points(j, perf, pch=21,
           bg=colori[round(10*dummy$aridity)],
           cex=.5*log10(dummy$area))
   } else {
      points(j + 0.15*(stratif - mean(stratif))/sd(stratif),
             perf, pch=21,
             bg=colori[round(10*dummy$aridity)],
             cex=.5*log10(dummy$area))
   }
  }
 }
}
```

Fig 5.27 at page 98 of the book:

```
layout(matrix(1:9, nrow=3, byrow=TRUE))
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="ANE",
                     characteristic="Aridity", ylim=c(3,0),
                     main="Global_regr")
 add_points(performance="ANEregr", variable="aridity", classes=aridity_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="ANE",
                     characteristic="Aridity", ylim=c(3,0),
                     main="Regional_regr")
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="ANE",
                     characteristic="Aridity", ylim=c(3,0),
                     main="Budyko")
 add_points(performance="ANEbudyko", variable="aridity", classes=aridity_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="ANE",
                     characteristic="MAT", ylim=c(3,0))
 add_points(performance="ANEregr", variable="temp", classes=temp_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="ANE",
                     characteristic="MAT", ylim=c(3,0))
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="ANE",
                     characteristic="MAT", ylim=c(3,0))
 add_points(performance="ANEbudyko", variable="temp", classes=temp_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="ANE",
                     characteristic="Area", ylim=c(3,0))
 add_points(performance="ANEregr", variable="area", classes=area_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="ANE",
                     characteristic="Area", ylim=c(3,0))
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="ANE",
                     characteristic="Area", ylim=c(3,0))
 add_points(performance="ANEbudyko", variable="area", classes=area_class, table=tabella)
```

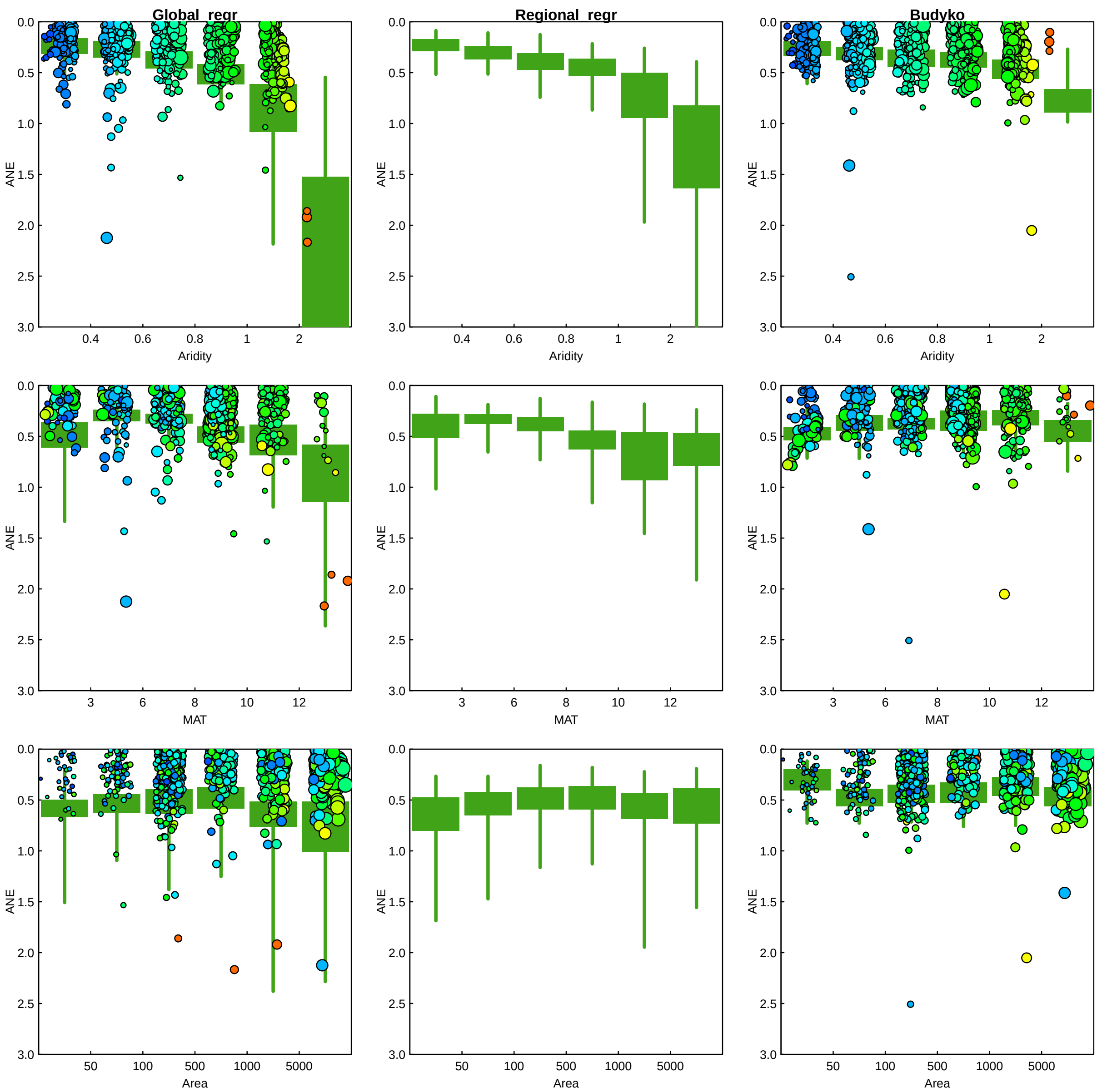
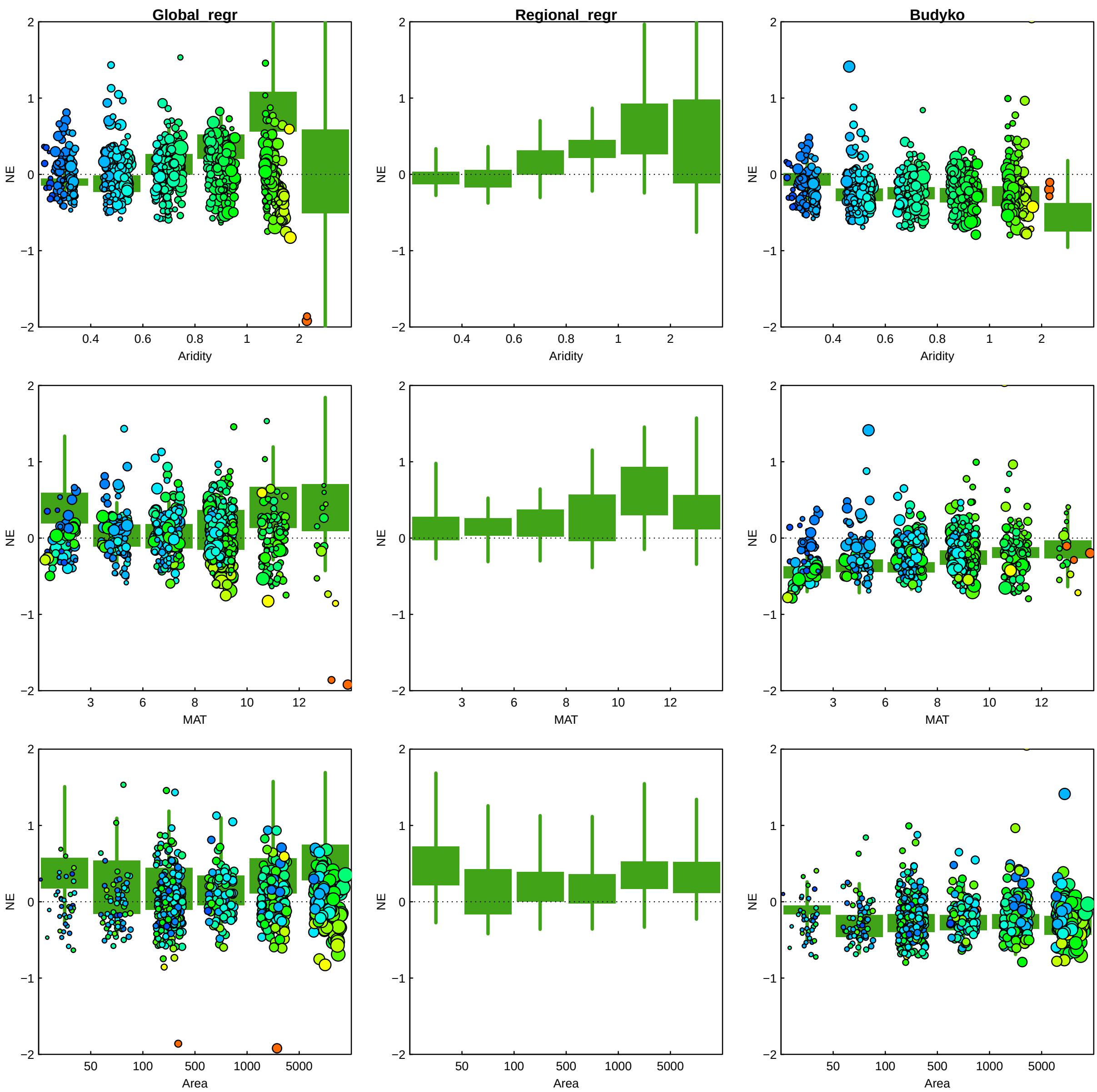


Fig 5.28 at page 99 of the book:

```
layout(matrix(1:9, nrow=3, byrow=TRUE))
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="NE",
                     characteristic="Aridity", ylim=c(-2,2),
                     main="Global_regr"); abline(h=0, lty=3)
 add_points(performance="NEregr", variable="aridity", classes=aridity_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="NE",
                     characteristic="Aridity", ylim=c(-2,2),
                     main="Regional_regr"); abline(h=0, lty=3)
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="NE",
                     characteristic="Aridity", ylim=c(-2,2),
                     main="Budyko"); abline(h=0, lty=3)
 add_points(performance="NEbudyko", variable="aridity", classes=aridity_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="NE",
                     characteristic="MAT", ylim=c(-2,2)); abline(h=0, lty=3)
 add_points(performance="NEregr", variable="temp", classes=temp_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="NE",
                     characteristic="MAT", ylim=c(-2,2)); abline(h=0, lty=3)
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="NE",
                     characteristic="MAT", ylim=c(-2,2)); abline(h=0, lty=3)
 add_points(performance="NEbudyko", variable="temp", classes=temp_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Global_regr", performance="NE",
                     characteristic="Area", ylim=c(-2,2)); abline(h=0, lty=3)
 add_points(performance="NEregr", variable="area", classes=area_class, table=tabella)
plotPUBfiguresLevel2(chapter=5, method="Regional_regr", performance="NE",
                     characteristic="Area", ylim=c(-2,2)); abline(h=0, lty=3)
plotPUBfiguresLevel2(chapter=5, method="Budyko", performance="NE",
                     characteristic="Area", ylim=c(-2,2)); abline(h=0, lty=3)
 add_points(performance="NEbudyko", variable="area", classes=area_class, table=tabella)
```



Notice that, contrary to our phylosophy, no cross-validation has been performed. In your opinion, why not?

References

Blöschl, G., Sivapalan, M., Wagener, T., Viglione, A. and Savenije, H. (2013) *Runoff Prediction in Ungauged Basins: Synthesis Across Processes, Places and Scales*, University Press, Cambridge, 484 pages, ISBN:9781107028180.

McMahon, T.A., G. Laaha, J. Parajka, M.C. Peel, H.H.G. Savenije, M. Sivapalan, J. Szolgay, S.E. Thompson, A. Viglione, R.A. Woods and D. Yang (2013) Prediction of annual runoff in ungauged basins. In *Runoff Prediction in Ungauged Basins: Synthesis Across Processes, Places and Scales*, University Press, Cambridge, 70-101, ISBN:9781107028180.

Parajka, J., Merz, R., and Blöschl, G. (2003) Estimation of daily potential evapotranspiration for regional water balance modeling in Austria. In *Transport of Water, Chemicals and Energy in the Soil - Crop Canopy - Atmosphere System*, 11th International Poster Day and Institute of Hydrology Open Day, pages 299-306.

Schroedter, H. (1985) *Verdunstung - Anwendungsorientierte Messverfahren und Bestimmungsmethoden*, Springer, 186 pages, ISBN:3540153551, 9783540153559.