R package SRWHessS2S

Simon Schick, Ole Rössler, Rolf Weingartner

June 28, 2017

About

The R package `SRWHessS2S` implements the model building procedure from the article "Monthly streamflow forecasting at varying spatial scales in the Rhine basin". The package serves as documentation and thus is not very user friendly, e.g. the functions do not test (apart from a very few exceptions) the calling arguments for suitability. In addition, much time could be spent to gain execution speed.

Preparation

The package does not depend on other packages not part of the standard library and has been tested for R version 3.4.0, though older versions very likely also work. You can install and load the package for example via:

```
install.packages('/somewhere/on/the/disk/SRWHessS2S_0.99.tar.gz
   ',repos=NULL)
library(SRWHessS2S)
?SRWHessS2S
```

To avoid copying the following code snippets, visit the help topic of `?series` where the complete use case is contained in the examples section.

Since the package assumes that the data is stored in `csv` or `rds` files, we load some test data and write it to file. The test data contains daily series of streamflow (Q), precipitation (P), and surface air temperature (T) of an anonymous catchment:

```
data('series',package='SRWHessS2S')
summary(series)
saveRDS(series,'/somewhere/on/the/disk/series.rds')
```

Outline

The general workflow consists of applying four functions, which are explained in more detail in the following sections:

1. `setVariables`: Describe all involved variables.
2. `getSeries`: Fetch the time series.
3. `buildBag`: Build the bagged model.
4. `applyBag`: Make predictions for new data.

Set variables

Each variable is described by a list which contains the type (`y` for the predictand, `x` for a predictor), the file name and column, a function for the time aggregation, and a sequence of aggregation intervals (for the predictand only one value is allowed):

```
f <- '/somewhere/on/the/disk/series.rds'
yag <- 30
xagb <- -1*seq(10,720,by=10)
xagf <- seq(10,yag,by=10)
l <- list(list(type='y',file=f,col='Q',fun='mean',agg=yag),
          list(type='x',file=f,col='P',fun='sum',agg=xagb),
          list(type='x',file=f,col='P',fun='sum',agg=xagf),
          list(type='x',file=f,col='T',fun='mean',agg=xagb),
          list(type='x',file=f,col='T',fun='mean',agg=xagf))
v <- setVariables(v=l)
```

The list `v` now contains all necessary information to fetch the data and to build the model. Additional arguments for the aggregation functions can also be specified, e.g. `na.rm=T`. See `?setVariables` for further details concerning additional arguments and supported file formats.

Fetch data

The data is fetched by specifying the period of investigation and passing the list returned by `setVariables` to `getSeries`:

```
p <- c(1981:2010)
d <- getSeries(v=v,p=p)
head(d)
```

Build the model

In order to build the model, we need to select the number of bootstrap replicates and the date of prediction. The latter is specified by the month followed by the day, e.g. '06-01' equals the 1st June. Since the predictands time aggregation is set to 30 days, the model tries to predict mean streamflow during 1st to 30th June:

```
set.seed(1)
B <- 100
t1 <- '06-01'
mo <- buildBag(v=v,p=p[1:20],d=d,t1=t1,na.rm=F,B=B,shuffle=F)
str(mo)
```

Here, we only used the years 1981 to 2000 and reserved the years 2001-2010 for model testing. In addition, the model building procedure can be shuffled by perturbing the predictand. This can be useful to obtain a zero skill model.

A little bit of model diagnostic is provided by inspecting the regression coefficients and aggregation periods:

```
par(mfrow=c(1,2))
boxplot(mo$mo[,grepl('^b\\.x',dimnames(mo$mo)[[2]])],main='
   regression coefficients')
boxplot(mo$mo[,grepl('^ag\\.x',dimnames(mo$mo)[[2]])],main='
   aggregation periods',ylab='days')
```

Based on the out of bag prediction error (an approximation to leave one out validation) we might also estimate the prediction error:

```
sk <- getSkill(va=mo$va)
sk$si
par(mfrow=c(1,2))
plot(x=sk$ts[,'p'],y=sk$ts[,'o'],xlab='prediction',ylab='
   observation')
abline(a=0,b=1,lty=3)
plot(sk$es,xlab='number of replicates',ylab='MSEP')
```

The last plot screens the mean squared error along an increasing number of bootstrap replicates. This series should converge in order to have a big enough bag of models.

Make predictions

Finally, we apply the model for data not used in model training:

```
op <- applyBag(v=v,p=p[21:30],d=d,mo=mo$mo,sc=mo$sc,t1=t1,na.rm
   =F)
par(mfrow=c(1,1))
plot(x=op[,c('p','o')],xlab='prediction',ylab='observation')
abline(a=0,b=1,lty=3)
```

Bagging

The concept of combining models out of bootstrap replicates was introduced by Leo Breiman in "Bagging Predictors" (url) and "Out-Of-Bag Estimation" (url).