GAM modelling workshop: computer lab exercises

Matteo Fasiolo and Simon N. Wood

The main packages we will need are `mgcv`, `mgcViz`, `qgam` and `mgcFam` which should have been installed by default by doing

```
install.packages("devtools")
library(devtools)
install_github("mfasiolo/mgcFam")
install_github("mfasiolo/testGam")
```

All the data sets we will use in the workshop should now be available in your `R` system. The rest of the material for the workshop can be downloaded from https://github.com/mfasiolo/workshop_RSS19. If you download it as a zip file and extract it, the solutions can be found in the `exercises/solutions` folder.

First session

In the first session you could try one or more of the following exercises (suggested track is ex 2, 1 and potentially 3, and the number of * indicates the difficulty level):

1. Retinopathy among diabetics (sol: "Retinopathy_mgcv.html"). Simple exercise on basic GAM modelling in `mgcv`. *

2. Modelling the simulated motorcycle accident data set (solution in: "motorcycle_mgcv.html"). Pedagogical exercise, using a 1D example to illustrate adaptive smooths, heteroscedastic data and location-shape GAM models. **

3. $C0_2$ modelling (sol: "CO2_mgcv.html"). Featuring cyclic seasonal smooths and the dangers of extrapolation. **

4. Ozone modelling (sol: "Ozone_mgcv.html"). Exercise focusing on manual variable selection via p-values and residual checking, and adjusting the mean-variance relationship. *

Second session

In this session you could try one or more of the following exercises (suggested track is ex 5 and 6):

5. Forecasting electricity demand on GEFCom2014 data (solution in: "gefcom_small_mgcv.html"). Simple exercise, focused on models building using residual checks and using only 1D effects. *

6. Larynx cancer in Germany (sol: "Larynx_mgcv.html"). Focused on spatial modelling using Markov Random Field, isotropic and tensor-product effects. *

7. Retinopathy among diabetics part 2 (sol: "Retinopath_mgcv_2.html"). Features 2D smooth interactions, automatic variable selection and the use of GCV vs REML for smoothing parameter selection. **

Third session

In this session you could try one or more of the following exercises (suggested track either ex 8 or 9):

8. Big GAM modelling of GEFCom14 electricity demand data (sol: "gefcom_big.html"). Featuring Big Data GAM methods and 2D tensor interactions. **

9. Individual electricity demand modelling (solution in: "Ind_elect.html"). Featuring Big Data GAM methods and by-costumer smooth effects. **

10. Mackerel egg data (sol: "Mackerel.html"). Featuring 2D spatial interactions. *

11. Bone mineral density modelling (sol: "bone_density.html"). Featuring simple random effects. *

Fourth session

In this session you could try one or more of the following exercises (suggested track either ex 12 or 13, and then 14):

12. GAMLSS modelling Body Mass Index (BMI) of Dutch boys (sol: "bmi_GAMLSS.html"). Basic exercise featuring adaptive smoothers and visual interactive model building. *

13. GAMLSS modelling of rent prices in Munich (sol: "Rent_munich_GAMLSS.html"). Featuring linear interactions. *

14. QGAM modelling of UK electricity demand (sol: "UKload_QGAM.html"). QGAM model-building on UK aggregate electricity demand. *

15. QGAM modelling of rainfall in Switzerland (sol: "Swiss_rainfall_QGAM.html"). Featuring spatio-temporal effects constructed using tensor product bases. **

1 Retinopathy among diabetics: part 1

Data frame `wesdr` (taken from Chong Gu's `gss` package) contains a subset of data from a Wisconsin study on development of retinopathy among diabetics. The following variables are provided:

- `ret`, a binary indicator of development of retinopathy by first follow up of study.
- `bmi`, the body mass index at entry to the study (between 18 and 25 is considered healthy).
- `dur`, the duration of diabetes, in years, at entry.
- `gly`, the percentage of glycosylated haemoglobin (HbA1C) in the blood (haemoglobin to which glucose has bound). 2.5-3.5% is normal for non-diabetics. 6.5% is generally considered good control for diabetics.

The aim is to model the probability of retinopathy as a function of the other variables.
Questions:

1. Load `testGam`, `mgcv` and the data (`data(wesrd)`). Have a look at the relation between the different variables using `pairs(wesdr)` or other exploratory visualisations.

2. Start by fitting a logistic regression model (`family = binomial`), with smooth effects for duration and body mass index. Use `summary` and `plot` to verify whether the effects are strong.

3. Use `qq.gam` to get a QQ-plot of the residuals (you can set the `rep` argument to, say, 100 to get simulation-based reference intervals). Is the plot interpretable? Now plot the residuals against the values of `wesdr$gly`, do you see any pattern?

4. Include a smooth effects for `gly` in the model formula and refit. Looks at the fitted effects and verify their significance using `plot` and `summary`. Compare the new model to the first model you fitted using `AIC`. Use `gam.check` to produce further diagnostics. Does the text suggest that the you should modify the number of basis functions used?

2 Simulated motorcycle accident

Here we consider the classic simulated motorcycle accident data set from the `MASS` package. The data frame gives a series of measurements of head acceleration in a simulated motorcycle accident, used to test crash helmets. See `?mcycle` for details.
Questions:

1. Load `mgcv` and the data (`library(MASS); data("mcycle")`). Have a look at the data using `head` and a scatterplot.

2. Fit a Gaussian GAM, with acceleration as response and a smooth effect for `times`. Obtain the fitted values and corresponding standard errors using `predict`, and plot the fitted values with confidence intervals on top of a scatterplot of acceleration vs time. Do you see any issues with this fit?

3. Use `gam.check` to get some diagnostics. Does the text output suggest that you should modify the number of basis function used? Try to increase the number of basis functions to 20, re-fit and re-check.

4. Now re-fit the model, using an adaptive basis for the effect of time `s(accel, k = 20, bs = 'ad')`. Use the `gam.check` or `summary` to check the number of EDF used. Has it changed relative to the non-adaptive fit? If so, think about why this has happened. Compare the fitted effect of time under the adaptive and non-adaptive basis, do you see any differences?

5. Use `qq.gam` to get a QQ-plot of the residuals (you can set the `rep` argument to, say, 100 to get simulation-based reference intervals). Does this plot reveal any problem? Now produce a scatterplot of absolute residuals vs time, does the size of the residuals depend on time? The answer is 'yes', and this is not taken into account by our model. We can address this 'manually' be estimating the variance of the residuals as a function of time, and use it to re-weight the observations when fitting the model (see next point).

6. Regress the log squared residuals on `times` using a Gaussian GAM. If we call this fit (say) `resFit`, the resulting fitted values in `resFit$fitted.values` are estimates of the expected value of the log squared residuals, as a function of time. Hence `exp(resFit$fitted.values)` is an approximation to the residuals variance. Re-fit the acceleration vs time Gaussian GAM with weights inversely proportional to the variance (you have to use the `weights` argument in `gam`). Compare the resulted fitted effect and intervals with those you obtained under the un-weighted fit.

7. **Extra:** Use the `gaulss` family in `mgcv` to fit a Gaussian GAM where the both the mean and the variance of the acceleration depend on time. See `?gaulss` for details.

3 CO_2 modelling

This question is about modelling data with seasonality, and the need to be very careful if trying to extrapolate with GAMs (or any statistical model). The data frame `co2s` contains monthly measurements of CO_2 at the south pole from January 1957 onwards. The columns are `co2`, the month of the year, `month`, and the cumulative number of months since January 1957, `c.month`. There are missing `co2` observations in some months.
Questions:

1. Load `mgcv` and the data with `library(gamair); data(co2s)`
2. Plot the CO_2 observations against cumulative months.
3. Fit a Gaussian additive model with a smooth effect for `c.month`, using the `gam` function. Use the `cr` basis, and a basis dimension of 100.
4. Obtain the predicted CO_2 for each month of the data, plus 36 months after the end of the data, as well as associated standard errors. Produce a plot of the predictions with twice standard error bands. Are the predictions in the last 36 months credible? NB: to produce the plot you have to write your own code, `mgcv` does not produce such plots.
5. Fit the model $\mathbb{E}(CO_2) = f_1(\texttt{c.month}_i) + f_2(\texttt{month}_i)$ where f_1 and f_2 are smooth functions. Use a basis of dimension 50 for f_1 and a cyclic basis for f_2. In the `gam` call, you will need to set argument `knots` to `list(month=c(1,13))` to make so that that the effect of January is the same as January, not that December and January are the same!
6. Repeat the prediction and plotting in question 4 for the new model. Are the predictions more credible now? Explain the differences between the new results and those from question 4.

4 Ozone modelling

Data frame `ozone` contains daily(ish) ozone measurements over Los Angeles (`O3`, ppm), along with:

`vh` the height at which the atmospheric pressure is 500mb, in metres.

`wind` the wind speed (reported as miles per hour, but this seems improbable).

`humidity` (usual % scale).

`temp` air temperature (Fahrenheit).

`ibh` the inversion layer base height in feet.

`ibt` the inversion base temperature (Fahrenheit).

`dpg` 'Dagget air pressure gradient' (mmhg).

`vis` visibility in miles.

`doy` Julian day, where 1 is Jan 1 1976.

The aim is to build a GAM model to explore the relationship between ozone and the other variables.
Questions:

1. Load `testGam` and the data (`data(ozone)`), and use something like `pairs(ozone)` to have a look at it.

2. Load mgcv and use gam to fit a Gaussian GAM with O3 as response, where $\log(\mathbb{E}(\texttt{O3}))$ is given by a sum of smooth functions (e.g. s(wind)) of each of the predictors. You will need to use the log-link, which requires using the argument family=gaussian(link=log) in the call to gam. Plot the fitted effects using plot.

3. Check the model residuals using the gam.check functions. Do you see any residual pattern when you plot the residuals against the fitted values or linear predictor?

4. Refit the model using a Gamma as response distribution (Gamma(link=log)), and re-check the residuals. Does the residual distribution look better?

5. Fit an alternative model where you are using the identity-link (Gamma(link=identity)). Does a model with an additive (i.e. identity-link) structure do better than that with a multiplicative (log-link) structure in terms of AIC?

6. Plot the smoothed effects again and use the summary function to see which effects are significant. Try simplifying the model.

7. Once you have converged on a model, plot it and interpret the fitted smooth effects: do they make sense?

5 Forecasting electricity demand on GEFCom2014 data

Here we consider the electricity demand dataset taken from the GEFCom2014 challenge. The dataset covers the period January 2005 to December 2011 and it contains the following variables:

- NetDemand net electricity demand between 11am and 12am.
- wM instantaneous temperature.
- wM_s95 exponential smooth of wM, that is wM_s95[i] = a*wM[i] + (1-a)*wM_s95[i] with a=0.95.
- Posan periodic index in [0, 1] indicating the position along the year.
- Dow factor variable indicating the day of the week (I think that 0=Sunday and 6=Saturday, but I am not sure).
- Trend progressive counter, useful for defining the long term trend.
- NetDemand.24 lagged version of NetDemand, that is demand at the same time of the previous day.
- Year should be obvious.

Questions:

1. Load testGam and the data (data(gefcom_small)). Have a look at it by, for instance, using pairs(gefcom_small).

2. Fit a Gaussian GAM where the model formula contains: smooth effects for wM, wM_s95, Posan (optionally use a cyclic basis for the latter by doing s(Posan, bs="cc")); parametric effects for Trend, Dow and NetDemand.24. Plot the fitted effects using plot, and look at the relative importance of the effects.

3. Plot the residuals against the `Trend` variable, do you see any non-linear dependence (you might need to zoom in using `ylim` because of an outlier)? Use `gam.check` to check whether you should increase `k` for any of the smooth effects.

4. Increase `k` for all effects, introduce a smooth effect for `Trend`, the re-fit and repeat the checks in the previous point. Does everything look good? `gam.check` shows that the effect of `Trend` is using all the basis functions available, is this a problem? Once you have converged on a model, compare your new model to the old one in terms of `AIC`.

5. Use `qq.gam` to produce a QQ-plot of the residuals, do you see any problem? Refit the same model, but now use a scaled Student-t distributions by setting `family = scat`. Any improvement in AIC? How do the residuals look now?

6. Check whether a scaled Student-t with log-link function `scat(link=log)` achieves lower AIC. Then plot all the fitted effects of final model using `plot` (you can set `all.terms=TRUE` to plot also the parametric effects). Do the effects make sense?

6 Larynx cancer in Germany

First load some data on cancer of the larynx by health reporting districts in Germany.

```
library(testGam)
library(mgcv)
data("Larynx")       # load Larynx cancer death data 'Larynx'
data("german.polys") # load polygons defining German regions 'german.polys'
# Get regions "midpoints"
X <- t(sapply(german.polys,colMeans,na.rm=TRUE))
```

The variables in the `Larynx` dataframe are:

region code identifying region;

E expected number of deaths (according to population and pan German total);

Y number of deaths from Larynx cancer 1986-1990;

x measure of smoking rate in region.

Questions:

1. Run the code above and then use `gam` to fit a Poisson GAM with a smooth effects for `x` and the following Markov Random Field (MRF) effect for region:

   ```
   s(region, k = 200, bs = "mrf", xt = list(polys=german.polys))
   ```

   and the offset term `offset(log(E))`, meant to take into account the fact that the number of death is proportional to the population of each region. Plot the fitted effects.

2. Now substitute the MRF smooth either with the isotropic smooth `s(X[,1],X[,2],k=200)`. Plot the 2D fitted effect in different ways using the `scheme` argument (see `plot.gam`) Which model does better in terms of AIC?

3. Now use a tensor product smooth `te(X[,1],X[,2],k=c(15, 15))` for the spatial effect. Plot it as before and compare the three spatial effects fitted so far. Which of the models does better in terms of AIC?

4. Check the last model we fitted using `check.gam`. Do you get an error? This is because so far we adopted the bad practice of using global variables (`X`) in our model formulas! Add each column of `X` as a proper variable in the `larynx` data set, modify the model formula accordingly and re-fit. Is the error gone? Now use the `vis.gam` function to visualise the spatial effect in 3D. You can use the `theta` and `phi` arguments to modify the viewpoint (see `?vis.gam`).

7 Retinopathy among diabetics (continued)

Data frame `wesdr` (taken from Chong Gu's `gss` package) contains a subset of data from a Wisconsin study on development of retinopathy among diabetics. The following variables are provided:

- `ret`, a binary indicator of development of retinopathy by first follow up of study.
- `bmi`, the body mass index at entry to the study (between 18 and 25 is considered healthy).
- `dur`, the duration of diabetes, in years, at entry.
- `gly`, the percentage of glycosylated haemoglobin (HbA1C) in the blood (haemoglobin to which glucose has bound). 2.5-3.5% is normal for non-diabetics. 6.5% is generally considered good control for diabetics.

The aim is to model the probability of retinopathy as a function of the other variables.
Questions:

1. Load `testGam`, `mgcv` and the data (`data(wesrd)`). Have a look at the relation between the different variables using `pairs(wesdr)` or other exploratory visualisations.

2. In a previous exercise we found out that `dur`, `gly` and `bmi` are all important predictors of retinopathy. Now we are looking for interaction of these variables. It is not immediately clear what interactions should appear in the linear predictor, hence in the first instance use all smooth main effects plus all two-way interactions using `ti` terms with `k=10`. Use a logistic regression model (`family = binomial`). Use `summary` verify which effects seems important and visualise them using `plot`. Do you see any problem? (**NB**: here we are using a large number of basis functions, $(k-1) \times (k-1)$, for each smooth interaction to make a point.)

3. Refit the same model, but now use `method = "REML"` to select the smoothing parameters by REstricted Marginal Likelihood, rather than via the default Generalized Cross Validation (GCV) method. Use `summary` and `plot` to check whether the model is still over-fitting.

4. Refit the same model, but now use `select = TRUE` to do automatic variable selection. Use `summary` to check whether the EDF used by the interactions have changed, relative to the first fit. Has the shape of the interaction terms changed as well?

5. Simplify the model by removing non-significant effects, re-fit and visualise the effects. Is the model with smooth interaction(s) better than a model with linear interactions (e.g. in terms of AIC)?

8 Big GAM modelling of GEFCom14 electricity demand data

Here we use again data from the GEFCom14 challenge, but this data set is 24 times larger than the one used in the previous exercise. This is because it contains data corresponding to all the 24 hourly slots. The variable `Instant` indicates the hourly window corresponding to each row of the data set. All remaining covariates have the same interpretation as before.
Questions:

1. Load `testGam`, `mgcViz` and the data (`data("gefcom_big")`). Create a model formula with smooth effects `wM`, `wM_s95`, `Instant`, `Trend` and `Posan`. Use regression splines bases (`bs='cr'`) for all smooths apart from `Posan`, for which you should use a cyclic basis (`bs='cc'`). Use `k = 6` for `Trend` and `k = 20` for `Posan`. Leave `k` to its default for the other smooths. Use parametric fixed effects for `Dow` and `NetDemand.24`. Use this formula within a `bamV` call to fit a Gaussian GAM. When calling `bamV` set `aGam=list(discrete=TRUE)` to speed up computations (do this in all subsequent `bamV` calls) and `aViz = list(nsim = 50)` to perform the response simulations needed for residuals checking. Having fitted the model, look at the effects using `plot` (recall that you can use argument `allTerms=TRUE` to plot also the parametric effects).

2. Use `check` to verify whether the number of basis functions used for the smooth effects is sufficiently large. Also, use the `check1D` function with the `l_gridCheck1D` layer to look for residual patterns across the variables.

3. Double `k` for any of the effects where the number of basis functions seems to small, and re-fit. After re-fitting, check whether AIC has improved and repeat the residual checks.

4. We expect that several of the effects might depend on the time of day. Use the `check2D` function with the `l_gridCheck2D` layer to look for interactions between `Instant` and `NetDemand.24`, `wM`, `wM_s95` and `Posan`. Notice that the binned mean residuals should ideally fall in the range (-2, 2) if the model was correct. Do you see any residual pattern? If so, fit a model which includes the necessary tensor product interactions (e.g. `ti(wM, Instant, k = c(10, 10))`) and repeat the checks. Are the patterns still there?

5. Assuming that we are now satisfied with our model, we'll now have a detailed look at the fitted smooth effects. First, look at the marginal effects using the `plot` function. Use the expression `print(plot(fit2, select = ???), pages = 1)` to plot all the marginal effects on one page (substitute `???` with the indexes of the univariate effects in your model). Do the same to plot the 2D interactions. Think about whether each effect makes physical sense to you. As an alternative to `plot`, recall that you can extract any effect using the `sm` function and produce a plot with customized layers. You can use the `listLayers` function to get a list of the available layers. Then, use the `plotRGL` function to manipulate each bivariate effect interactively.

6. **Extra question**: the model could be improved further. For instance, use the `check2D` function with the `l_gridCheck2D` layer to look at how the standard deviation and skewness of the residuals vary across pairs of covariates (the `e1071` package provides a `skewness` function, then you simply need to set `gridFun = skewness` in the call to `l_gridCheck2D`). Do you see any pattern? At this point we could consider a GAMLSS model with linear predictors for location, variance and skewness (e.g. the `gaulss` or `shash` family). However, `bam` methods does not yet support such models, so you'll need to use `gam` which can be much slower for large models.

9 Individual electricity demand modelling

Here we consider electriciy demand from 28 commercial costumers. The dataset covers roughly three months and it contains the following variables:

- `load` power usage from an individual costumer (in KW, I guess);
- `DateTime` the date and the time of day;
- `instant` the time of day, where 1 corresponds to 00:00-00:30, 2 to 00:30-01:00 and so on;
- `dow` factor variable indicating the day of the week;
- `temp` instantaneous temperature;
- `tempL` exponential smooth of `temp`, that is `tempL[i] = a*temp[i] + (1-a)*tempL[i-1]` with `a=0.95`;
- `ID` the unique ID of each individual costumer;
- `load48SM` lagged version of smoothed `load`, where the smoothing was performed as for `tempL`.
- `day` a counter depending on the day.

Questions:

1. Load `testGam`, `mgcViz` and the data (`load("Ind_elect")`). Then use `bamV` to fit a Gaussian GAM model with smooth effects for `instant`, `temp` and `day`, and parametric effects for `dow` and `ID`. In the call to `bamV` set `aViz = list(nsim = 50)` to perform the response simulations needed for residuals checking. Look at the model output using `plot` and `summary`.

2. Now we start looking for interactions. Use the `check2D` function with the `l_gridCheck2D` layer to look for interactions between `ID` and `instant`, `temp` and `day`. Notice that the binned mean residuals should ideally fall in the range (-2, 2), if the model is correct. Do you see large deviation? If so for which costumer(s) in particular?

3. Modify the model formula to include by-factor smooths, that is `s(instant, by = ID, id = 1)` `s(temp, by = ID, id = 2)` and `s(day, by = ID, id = 3)`. The `id` argument make so that each of the 3 by-factor smooths has its own smoothing parameter, but the same smoothing parameter is used across all costumers. Refit the model using `bamV`, and set the argument `aGam=list(discrete=TRUE)` to speed up computation by discretisation. Compare this models to the previous one using AIC, and repeat the residuals checks. Any improvement?

4. Use `check` to verify whether the number of basis functions used for the smooth effects is sufficiently large. Double `k` for any of the effects where the number of basis functions seems to small, and re-fit. After re-fitting, check whether AIC has improved.

5. Use the `check2D` function with the `l_gridCheck2D` layer to look for interactions between `ID` and `load48SM`. If the effect of `load48SM` seems important, include the corresponding by-factor smooth by adding `s(load48SM, by = ID, id = 4)` to the model and re-fit.

6. Look at the model output using `plot`, using the `select` argument to plot any specific effect (you can't plot them all together, because the model includes tens of them). Compare the consumption of some of the individual costumers with the model predictions (which you can find in `fittedModel$fitted.values`). Do some costumers look much harder to predict than others?

10 Mackerel egg data

The following code loads and plots some data from a fish egg survey, for the purposes of spatial modelling.

```
library(testGam); library(mgcViz); data("mack"); data("coast")
## plot data....
with(mack,plot(lon,lat,cex=0.2+egg.dens/150,col="red"))
lines(coast)
ind <- c(1,3,4,5,10,11,16)
pairs(mack[,ind])
```

The main variables of interest in the `mack` data set are:

- `egg.count` number of eggs found in the net;
- `c.dist` distance from 200m seabed contour;
- `b.depth` depth of the ocean;
- `temp.surf` surface temperature of the ocean;
- `temp.20m` water temperature at a depth of 20 meters;
- `lat` latitude;
- `lon` longitude;
- `salinity`;
- `net.area` the area of the net used in m^2.

Questions:

1. Use the code above to load and plot the data;
2. Create a new variable `mack$log.net.area <- log(mack$net.area)`, and use `gamV` to fit a Poisson GAM with `egg.count` as response variable and 1D smooth effects for all the other variables, with the exceptions of `net.area` and `log.net.area`. Instead, include in the model formula the term `offset(log.net.area)`, meant to take into account the fact that the number of eggs captured is proportional to the net area.
3. Look at the model residuals using `qq`. What kind of problem do you see? Re-fit the models using a negative binomial (`family=nb`) or Tweedie (`family=tw`) response distribution, and check which model is better in terms of residuals QQ-plots and AIC.
4. Let `fit` be the best of the three GAM models you just fitted. Use `fit<-getViz(fit,nsim=50)` to get some simulated residuals, and then use the `check2D` function with the `l_gridCheck2D` layer to look for residual patters across `lon` and `lat`. Then refit the model using a bivariate isotropic effect `s(lon, lat, k=100)`, re-check the residuals and see whether AIC has improved.
5. Use `check` to verify whether the number of basis functions used for the smooth effects is sufficiently large. Then use the `check1D` function with the `l_gridCheck1D` layer look for residual patterns across some of the variables. If necessary, modify the model.
6. Plot the fitted effects using `plot`. Which effects look more important (look at the scales)? Use the `plotRGL` function to manipulate spatial effect interactively.

11 Bone mineral density modelling

This dataset is taken from the package `lava`. It consists of 112 girls randomized to receive calcium or placebo. The response variable of interests consists of longitudinal measurements of bone mineral density (g/cm^2) measured approximately every 6th month for 3 years. All girls are approximately 11yo at the start of the trial. The main variables are:

- `bmd` bone mass density;
- `group` placebo or supplement;
- `person` factor indicating the id of each girl;
- `age` the age of each girl at the time of each measurement;

Questions:

1. Load `testGam`, `mgcViz` and the data `data("calcium")`. Then use `gamV` to fit a Gaussian GAM model with `bmd` as response and linear effects for `age` and `group`. In the call to `gamV` set the argument `aViz=list(nsim = 50)` to have some simulated responses for residuals checks. Use `summary` to print the model output. Is the placebo effect significant? (which is the same as asking whether the treatment effect is significant)

2. Use `check1D` with the `l_gridCheck1D` layer to check that the mean of the negative residuals does not depart too much from 0, for any of the subjects. If you see significant departures add a random effect for `person` to the models formula (`s(person, bs="re")`), then re-fit and re-check the residuals. Print the model output again using `summary`.

3. Now modify the model formula to use a smooth effect for `age`, and plot the fitted effects using `plot`. Use the function `AIC` to compare the model with a smooth effects for `age` with the model which uses a linear `age` effect.

4. Verify whether the smooth age effect is different between the placebo and the treatment group, by using a by-factor smooth. To do this substitute `s(age)` with `s(age, by=group)` in the model formula, refit and then plot the fitted effects. To see the difference between the two smooths more clearly, use the `plotDiff` function with the `l_fitLine` and `l_ciLine` layers.

12 Body Mass Index (BMI) of Dutch boys

This simple data set comes from the Fourth Dutch Growth Study, which is a cross-sectional study that measures growth and development of the Dutch population between the ages 0 and 21 years. Here we have only two variables: `bmi` and `age`. The data is taken from the `gamlss.data` package.
Questions:

1. Load `testGam`, `mgcViz` and the data (`data("dbbmi")`). The use `gamV` to fit a Gaussian GAM with simply a single smooth effect for `age`. Set argument `aViz=list(nsim = 50)` to have some simulated responses for residuals checks. Then plot the data (a scatterplot `bmi` vs `age`) and add a line representing the fitted mean BMI (you can use the `predict` function).

2. Check the residual distribution using `qq`: do you see any problem? Then use the `check1D` function together with the `l_gridCheck1D(gridFun=sd)` layer to check whether the conditional standard deviation of the residuals varies with age. If so address this by fitting a Gaussian GAMLSS model

(`family = gaulss`), with model formula `list(bmi ~ s(age), ~ s(age))`. Then repeat the residuals checks. Any improvement?

3. Use `check` to verify whether the number of basis functions used for the smooth effects is sufficiently large. Then increase the number of basis functions used for each effect to 20 (`k=20`), and use an adaptive basis fom the effect of age on mean BMI (`bs = "ad"`). Is this model better in terms of AIC? Does the output of `check` look ok now? Plot the smooth effects, and decide whether they make sense. Do you see why we used an adaptive smooth for the effect of age on mean BMI?

4. Now we look at residual skewness. Load the `e1071` package, and use the `check1D` function together with the `l_gridCheck1D(gridFun=skewness)` layer to check whether the conditional skewness of the residuals varies with age. To take skewness into account, load the `mgcFam` package, and fit a shash GAM model (`family=shash`) with model formula:

```
list(bmi ~ s(age, k = 20, bs = "ad"), ~ s(age, k = 20), ~ s(age), ~ 1)
```

   Do we get lower AIC, and how does a residuals QQ-plot look? Plot all the smooth effect and use `check` to verify that everything is ok.

5. Now we plot the fitted conditional distribution. Let `fit4` be the shash model you just fitted, then you can plot several estimated conditional quantiles by doing:

```
plot(bmi~age, data=dbbmi, col = "grey")
pr <- predict(fit4)
for(.q in c(0.01, 0.25, 0.5, 0.75, 0.9)){
  q_hat <- fit4$family$qf(.q, pr, wt = fit4$prior.weights, scale = 1)
  lines(dbbmi$age, q_hat, col = 2)
}
```

13 Rent modelling in Munich

This data set comes from `gamlss.data` package. The main variables are:

- `R` rent response variable, the monthly net rent in DM;
- `Fl` floor space in square meters;
- `A` year of construction;
- `B` a binary indicating whether there is a bathroom, 1, (1925 obs.) or not, 0, (44 obs.);
- `H` a binary indicating whether there is central heating, 1, (1580 obs.) or not, 0, (389 obs.);
- `L` a binary indicating whether the kitchen equipment is above average, 1, (161 obs.) or not, 0, (1808 obs.);
- `loc` a factor indicating whether the location is below, 1, average, 2, or above average 3.

Questions:

1. Load `testGam`, `mgcViz` and the data (`data("munich_rent")`) and have a look at it by doing `pairs(munich_rent)`. Then use `gamV` to fit a Gaussian GAM with `rent` as response, smooth effects for `Fl` and `A` and fixed effects for the remaining covariates. Set argument `aViz=list(nsim = 50)` to have some simulated responses for residuals checks. Use `summary` and `plot` to see which are the most important effects.

2. The effect of `Fl` looks fairly linear, but it should depend on the location's desirability (`loc`). Substitute the smooth effect for `Fl` with a linear effect for `Fl` and the interaction `Fl:loc`. Is there any improvement in AIC? Do the fitted coefficient reported by `summary` make sense?

3. Use the `check1D` function together with the `l_gridCheck1D(gridFun=sd)` layer to check whether the conditional standard deviation of the residuals varies with any of the covariates. If so address this by fitting a Gaussian GAMLSS model (`family = gaulss`), with the same formula for mean and for scale. Then repeat the residuals checks. Any improvement? Do you get lower AIC?

4. Now look at the residuals distribution using `qq`. Do you see any departure from normality? Check whether the conditional skewness of the residuals varies with any of the covariates by loading the `e1071` package, and using the `check1D` function together with the `l_gridCheck1D(gridFun=skewness)` layer. To take skewness into account, load the `mgcFam` package, and fit shash GAM model (`family=shash`) with model formula:

```
list(R ~ Fl + Fl:loc + s(A) + loc + B + H + L,
       ~ Fl + Fl:loc + s(A) + loc + B + H + L,
       ~ Fl + Fl:loc + s(A) + loc + B + H + L,
       ~ 1)
```

   Do we get lower AIC, and how does a residuals QQ-plot look? Do the skewness checks obtained with `check1D` look better now? Finally plot the smooth effects.

14 Quantile modelling of UK electricity demand

Here we consider a UK electricity demand dataset, taken from the national grid. The dataset covers the period January 2011 to June 2016 and it contains the following variables:

- `NetDemand` net electricity demand between 11:30am and 12am.
- `wM` instantaneous temperature, averaged over several English cities.
- `wM_s95` exponential smooth of `wM`, that is `wM_s95[i] = a*wM[i] + (1-a)*wM_s95[i-1]` with `a=0.95`.
- `Posan` periodic index in `[0, 1]` indicating the position along the year.
- `Dow` factor variable indicating the day of the week.
- `Trend` progressive counter, useful for defining the long term trend.
- `NetDemand.48` lagged version of `NetDemand`, that is `NetDemand.48[i] = NetDemand[i-1]`.
- `Holy` binary variable indicating holidays.
- `Year` and `Date` should obvious, and partially redundant.

Questions:

1. Load `mgcViz` and the data (`data("UKload")`). Then create a model formula (e.g. `y~s(x)`) containing: smooth effects for `wM`, `wM_s95`, `Posan` and `Trend` with 20, 20, 50 and 4 knots and cubic regression splines bases (`bs='cr'`); parametric effects for `Dow`, `NetDemand.48` and `Holy`.

2. Use the `qgamV` function to fit this model for the median. Call (say) `fit` the fitted model and use `plot(fit)` and `summary(fit)` to visualise the fitted effects and to see which effects are significant. Do you notice anything problematic about the effect of `Posan`? How many degrees of freedom are we using for this smooth effect (you can read it from the output of `summary`)?

3. Modify the effect of `Posan` to use an adaptive (`bs='ad'`) spline basis. Then refit the model and plot the smooth effects. Has the effect of `Posan` changed? How many degrees of freedom are we using now for `Posan`? Explain what happened.

4. Use `mqgamV` to fit this model to the five quantiles `qu=seq(0.1,0.9,length.out=5)`. Use `plot` to visualize the smooth effects corresponding to each quantile. You can set `allTerms=TRUE` to plot also the parametric effects. How do the smooth and parametrics effects differ between quantiles? NB: here we are plotting the smooth effects, not the predicted quantiles, hence the effects corresponding to, say, quantile 0.9 can fall below that of quantile 0.1.

5. Now we check the median fit. If the output of `mqgamV` is called `fitM` then the median fit is `fitM[[3]]`. Use `check1D` with the `l_gridQCheck1D` layer to check that the fraction of negative residuals does not depart too much from 0.5 along any of the covariates.

15 Rainfall modelling in Switzerland

This question is about modelling extreme rainfall in Switzerland, mainly using spatio-temporal effects. The main variables are:

- `exra`: the highest rainfall observed in any 12 hour period in that year, in mm;
- `N`: degrees North;
- `E`: degrees East;
- `elevation`: metres above sea level;
- `climate.region`: factor variable indicating one of 12 climate regions;
- `nao`: annual North Atlantic Oscillation index, based on the difference of normalized sea level pressure (SLP) between Lisbon, Portugal and Stykkisholmur/Reykjavik, Iceland. Positive values are generally associated with wetter and milder weather over Western Europe;
- `year`: year of the observation;

Questions:

1. Load `mgcViz`, `gamair` and the data with `data(swer)`. Use `qgamV` to fit an additive quantile regression model for the median of `exra`, with smooth effects for `nao`, `elevation` and `year` (use `k=5` for the latter), an isotropic smooth for `E` and `N` (i.e. `s(E,N)`), and a fixed effect for `climate.region`. Look at the significance of the fitted effects using `summary` and plot them using `plot`.

2. We might be interested in verifing whether the rainfall trend is different depending on the climate region. To assess this, modify the model formula to include a by-factor smooth as follows `s(year, climate.region, bs = "fs", k = 5)` (you will have to remove the fixed `climate.region` effect). Refit and use `summary` to verify whether the by-region trend term is significant, and plot the by-region trends by extracting it using `sm` and the `l_fitLine(alpha = 1)` layer.

3. We can also verify whether the bivariate spatial effect changes with time, by creating a tensor product between the 2D effect of `E` and `N`, and the effect of `year`. Such an effect can be set up using `te(E, N, year, d = c(2, 1), k = c(20, 5))`. Fit the corresponding median QGAM model, and plot several slices of the 3D tensor product across `year`, using the `plotSlice` function with the `l_fitRaster` and `l_fitContour` layers.

4. Visualize individual 2D slices (across `year`) of the 3D spatio-temporal smooth using the `plotRGL` function (see `?plotRGL.mgcv.smooth.MD` for examples).

5. Go back to the simpler model formula used in the first question and fit the corresponding model to the quantiles `qu = seq(0.1, 0.9, length.out = 9)`, using `mqgamV`. Plot only the univariate effects using `plot` and its `select` argument, and see how they differ between quantiles. Do the same for the spatial effect and for the effect of the climate region.