Report about our daily problems using the refund package

José Carlos Soares Junior

03/17/2021

What are we working on?

We are mainly studying regression models for the health area that use functional predictors and has a scalar dependent variable, in other words, we are studying scalar-on-function regression. We are using a database called *Base_TocoColoPerinatal* with data from 336 pregnant women that contains records of measurements of contractions and the cervix measurements obtained during the prenatal period, which are our two functional predictors where the measurements of both was collected to have 5 points evaluated over time. In addition, the database has information on variables that do not change over the course of prenatal care, such as education and age for example. Initially, the data was collected for a study from the Hospital das Clínicas da Faculdade de Medicina da Universidade de Sao Paulo (FMUSP), now we are using it in my undergraduate research.

What is the problem that motivated this report?

To fit the models we were first using the *fda* package, however, it does not allow the fit of two functional predictors at the same time. On the other hand, the *refund* package has options to fit more than one functional predictor at the same time, the option implemented in the package that we are trying to use is the *pfr_old()* function with the argument *funcs*, which performs a Penalized Functional Regression.

What's the problem with the *pfr_old()* function ?

When we tried to use the *pfr_old()* function we were unable to fit de models even with a single functional predictor due to an error that i still don't quite understand. Using our data with 5 evaluations for the functional predictors and the MICE imputation method applied to the missing data, we get the following error:

```
load("objects_mice.RData")
Y <- colo_longf$igp_parto
Y <- as.matrix(Y)
X1 <- colo_longf$medida_colo
X1 <- as.matrix(X1)
X1 <- I(X1)
X2 <- contra_longf$num_contra
X2 <- as.matrix(X2)
X2 <- I(X2)
id <- colo_longf$id
id <- as.matrix(id)

data_asis2 <- data.frame(igp_parto=Y,
                         id=id,
                              colo_perfis=X1,
                              contra_perfis=X2)

rm("Y")
rm("X1")
rm("X2")
rm("id")
Y <- data_asis2$igp_parto
X1 <- data_asis2$colo_perfis
X2 <- data_asis2$contra_perfis
id <- data_asis2$id

# 1 functional predictor:
fitJJ = pfr_old(Y = Y,funcs = X1,subj = id,
            kz = 4,kb = 9,nbasis = 5)
## Error in smooth.construct.tp.smooth.spec(object, dk$data, dk$knots): A
   term has fewer unique covariate combinations than specified maximum
   degrees of freedom

# 2 functional predictors:
fitJJ = pfr_old(Y = Y,funcs = list(X1,X2),subj = id,
            kz = 4,kb = 9,nbasis = 5,smooth.option="fpca.sc")
## Error in smooth.construct.tp.smooth.spec(object, dk$data, dk$knots): A
   term has fewer unique covariate combinations than specified maximum
   degrees of freedom
```

At first we thought that it could be a problem due to the data imputation method used, first we observe this error after an imputation by the mean, then we used the MICE imputation method which the code above was made and the error persists.

Researching about the problem I started to think in some hypothesis. In the documentation of the function and the package it is not very clear whether the data should come in long or wide format, and at first i was trying to use the data in long format (the code above). The point is, it seems that this error occurs when I try to use more knots than evaluated points, and if I enter the data in a long format and the function interprets the number of points contained in a row, the maximum number of knots i should be able to use would be 1 using a functional predictor (for now this is making sense in my head at least), which is not feasible. When trying to use the data in wide format i can get rid of this error, but another one that i haven't been able to see why it happens shows up, let's see it below:

```
Y <- as.numeric(dados_colo_completo[,"igp_parto"])
Y <- as.matrix(Y)
X1 <- dados_colo_completo[,2:6]
X1 <- as.matrix(X1)
X1 <- I(X1)
X2 <- dados_contra_completo[,2:6]
X2 <- as.matrix(X2)
X2 <- I(X2)
id <- dados_contra_completo$id
id <- as.matrix(id)

data_asis1 <- data.frame(igp_parto=Y,
                id=id,
                colo_perfis=X1,
                contra_perfis=X2)
```

```
rm("Y")
rm("X1")
rm("X2")
rm("id")
Y <- data_asis1$igp_parto
X1 <- data_asis1$colo_perfis
X2 <- data_asis1$contra_perfis
id <- data_asis1$id

# 1 functional predictor:
fitJJ = pfr_old(Y = Y,funcs = X1,subj = id,
            kz = 4,kb = 9,nbasis = 5)
## Error in gam(as.vector(G.0) ~ te(row.vec, col.vec, k = nbasis), weights
    = as.vector(cov.count)): Model has more coefficients than data

# 2 functional predictors:
fitJJ = pfr_old(Y = Y,funcs = list(X1,X2),subj = id,
            kz = 4,kb = 9,nbasis = 5,smooth.option="fpca.sc")
## Error in gam(as.vector(G.0) ~ te(row.vec, col.vec, k = nbasis), weights
    = as.vector(cov.count)): Model has more coefficients than data
```

So, i am not sure which format is actually the right one. We started to think that the possible cause of all these problems is the number of points evaluated over time of each pregnant woman, so we changed the approach of the problem to be able to work with 11 points instead of 5 and check if the errors persisted. Now using the data *dados_igs_completas* which is the *Base_TocoColoPerinatal* with MICE imputation, and modified from 5 to 11 evaluations over time of the functional predictors, we obtained the following:

```
load("objects_11aval.RData")
Y <- medida_colo_wide$igp_parto
Y <- as.matrix(Y)
X1 <- medida_colo_wide[,3:13]
X1 <- as.matrix(X1)
X1 <- I(X1)
X2 <- num_contra_wide[,3:13]
X2 <- as.matrix(X2)
X2 <- I(X2)
id <- medida_colo_wide$id
id <- as.matrix(id)

data_asis1 <- data.frame(igp_parto=Y,
                         id=id,
                         colo_perfis=X1,
                         contra_perfis=X2)

rm("Y")
rm("X1")
rm("X2")
rm("id")
Y <- data_asis1$igp_parto
X1 <- data_asis1$colo_perfis
X2 <- data_asis1$contra_perfis
id <- data_asis1$id

#1 functional predictor:
fitJJ = pfr_old(Y = Y,funcs = X1,subj = id,kz = 9,
            kb = 9,nbasis = 10)
## Error in gam(Y ~ X - 1, paraPen = list(X = D), method = method, family
   = family, : Model has more coefficients than data

#2 functional predictors:
fitJJ = pfr_old(Y = Y,funcs = list(X1,X2),subj = id,kz = 9,
            kb = 9,nbasis = 10)
## Error in gam(Y ~ X - 1, paraPen = list(X = D), method = method, family
   = family, : Model has more coefficients than data
```

As you can see, using a wide format data we have the same problem from before. The same problem that the long format data had with 5 points also happens when it is 11 points of evaluation, so or our data has some characteristics that doesn't work well here, or we are just not understanding how it should work with this function.

What's the problem with the *pfr()* function ?

The new version of the *pfr_old()* function is now called *pfr()*, the difference from this function to the previous one is that it uses a formula argument to which a function such as *lf()* or *fpc()* must be added. The fit with one functional predictor occurs well in this new version, the problem is that nowhere in the documentation of the *refund-version:0.1-23* package is it possible to find any guidance regarding the fit of more than one functional predictor at the same time. We tried to explore the functions but we did not find how this type of fit can be made in the new version, the argument 'funcs' from the *pfr_old()* was the one that had this option for the Penalized Functional Regression approach, but it doesn't seem to work on the new one.

Obs: All the scripts can be found at https://github.com/Soju-JC/Undergraduate-research