BIOS 7720: Applied Functional Data Analysis

Lecture 7: Scalar on Function Regression (SoFR)

Andrew Leroux

April 1, 2021

Roadmap

1. Final Project Proposal
2. Introduction to SoFR
3. Methods for Estimation
4. In Class Exercises

Project Proposal

1. Due next Thursday (4/8), **ungraded**
2. Very short (1-2 paragraphs) text document:
    1. Description of the dataset you're using
        - Source (e.g. web scraping, data repository, etc.)
        - Data generating mechanism (e.g. clinical trial, observational, etc.)
        - Size of the data (number of observational units, covariates, etc.)
    2. Explicit description of the functional data in your dataset
        - What
    3. Scientific question and the role of your functional data in answering that question
        - What is the relevant scientific question?
        - Functional data as outcome or predictor?

Set Up

- Notation
  - Sampling unit (e.g. participant) by $i = 1, \ldots, N$
  - Scalar outcome y_i
  - Scalar predictor x_i
  - Functional predictor $z_i(s)$ observed on regular grid $s_1, \ldots, s_J$
- Observed data are then of the form

$$[\{y_i, x_i, Z_i(s_j)\}, 1 \leq j \leq J, 1 \leq i \leq N]$$

Motivating Data Set

- NHANES physical activity data
- Outcome (y_i) is 5-year all cause mortality
- Functional predictor is participants log transformed activity profile:

$$z_i(s) = M_i^{-1} \sum_{m=1}^{M_i} \log(1 + \mathrm{AC}_{im}(s))$$

where $s = 1, \ldots, 1440$, $m = 1, \ldots, M_i$ denotes "good" days of data, and $\mathrm{AC}_{im}(s)$ denotes the activity count for subject i on day m at minute s

Motivating Data Set

SoFR
Estimation
In-Class Exercises

- These data are noisy, may want to smooth the data using fPCA via *refund::fpca.face()*

$$\tilde{z}_i(s) = \sum_{k=1}^{K_z} \xi_{ik}^z \phi_k^z(s)$$

- Where ξ_{ik}^z and ϕ_k^z are the scores and PCs estimated from fPCA
- Functional predictor then $\tilde{z}_i(s)$

Motivating Data Set

```
library("here"); library("readr"); library("dplyr")
data <- read_rds(here("data","data_processed","NHANES_AC_processed.rds"))
## create the functional predictor
data <-
  data %>%
  ## only consider good days of data and indiviudals age 50 or over
  filter(good_day %in% 1, Age > 50)
## get mortality data from the rnhanesdata package
library("rnhanesdata")
data_mort <- bind_rows(Mortality_2015_C, Mortality_2015_D)
str(data_mort)

## 'data.frame': 20470 obs. of  8 variables:
##  $ SEQN         : int  21005 21006 21007 21008 21009 21010 21011 21012 21013 21
##  $ eligstat     : int  1 2 2 2 1 1 2 1 2 2 ...
##  $ mortstat     : int  0 NA NA NA 0 0 NA 1 NA NA ...
##  $ permth_exm   : int  150 NA NA NA 135 149 NA 127 NA NA ...
##  $ permth_int   : int  150 NA NA NA 135 149 NA 128 NA NA ...
##  $ ucod_leading : chr  NA NA NA NA ...
##  $ diabetes_mcod: int  NA NA NA NA NA NA NA 0 NA NA ...
##  $ hyperten_mcod: int  NA NA NA NA NA NA NA 0 NA NA ...
```

Motivating Data Set

SoFR

Estimation

In-Class Exercises

```r
## merge with our data and derive 5-year mortality indicator
data      <-
  left_join(data, data_mort, by="SEQN") %>%
  mutate(mort_5yr = as.numeric(permth_exm/12 <= 5 & mortstat %in% 1),
         ## replace accidental deaths within 5 years as NA
         mort_5yr = ifelse(mort_5yr == 1 & ucod_leading %in% "004", NA, mort_5yr))
  ## drop anyone missing mortality data or who had accidental deaths within 5 year
  filter(!is.na(mort_5yr))
```

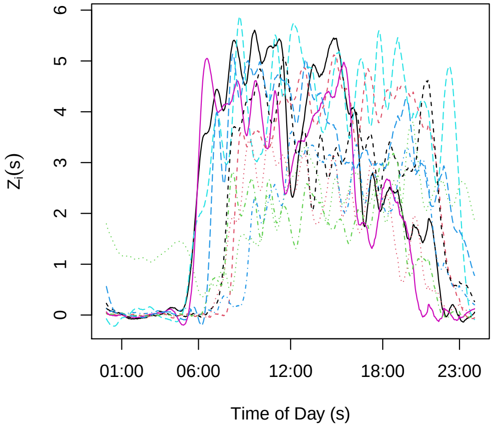
Motivating Data Set


```r
library("refund")
## extract just the activity count data
Z <- log(as.matrix(data[,paste0("MIN",1:1440)])+1)
Z[is.na(Z)] <- 0
## average across days within participants (SEQN)
uid <- unique(data$SEQN)       # unique subject identifiers
nid <- length(uid)             # number of participants
Zmat <- matrix(NA, nid, 1440) # empty container to store average profiles
inx_ls <- lapply(uid, function(x) which(data$SEQN %in% x)) # list of indices
for(i in seq_along(uid)){
  Zmat[i,] <- colMeans(Z[inx_ls[[i]],,drop=FALSE])
}
## do fpca on the log(1+AC)
fpca_Z <- fpca.face(Y=Zmat, knots=50)
Zsm    <- fpca_Z$Yhat
```

Motivating Data Set


```r
## Get a data frame for analysis which contains one row per participant
df <- data[!duplicated(data$SEQN), ]
## drop the activity count columns
df <-
  df %>%
  dplyr::select(-one_of(paste0("MIN",1:1440)))
## add in the activity count matrix using the AsIs class via I()
## note!! be careful when working with dataframes which contain matrixes
df$Zsm  <- I(Zsm)
df$Zraw <- I(Zmat)
## clean up the workspace a bit
rm(Zsm);rm(Zmat);rm(Z)
```

Motivating Data Set

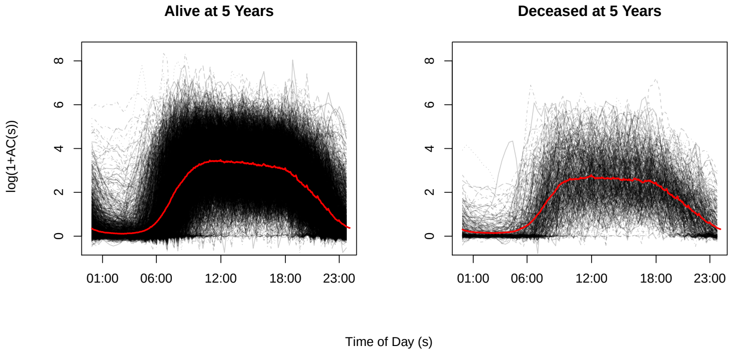


Motivating Data Set

SoFR

Estimation

In-Class Exercises



SoFR: NHANES

- We want to model the association between y and $x, z(s)$
- Naive approach:

$$g(E[y_i | x_i, \boldsymbol{z}_i]) = \alpha_0 + x_u\beta + \sum_{j=1}^{J} \gamma_j z_i(s_j)$$

or

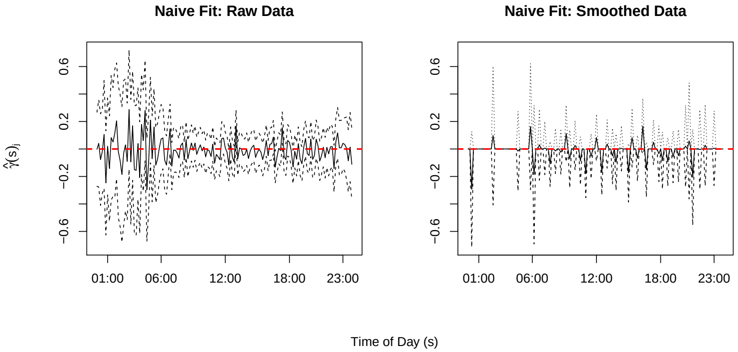
$$g(E[y_i | x_i, \boldsymbol{z}_i]) = \alpha_0 + x_u\beta + \sum_{j=1}^{J} \gamma_j \tilde{z}_i(s_j)$$

- Potential problems?

SoFR: NHANES


```
library("mgcv")
## fit on a subset of minutes (could do all 1440, just long computation time)
cols_regress <- seq(1,1440,by=10)
fit_naive_raw <- gam(mort_5yr ~ df$Zraw[,cols_regress], family=binomial, data=df)
fit_naive_sm  <- gam(mort_5yr ~ df$Zsm[,cols_regress], family=binomial, data=df)
```

SoFR: NHANES


Naive Fit: Raw Data
Naive Fit: Smoothed Data
$\hat{\gamma}(s)_j$
0.6
0.2
−0.2
−0.6
01:00
06:00
12:00
18:00
23:00
Time of Day (s)

Generalized Functional Linear Model (GFLM)

$$g(E[y_i|x_i, \boldsymbol{Z}_i]) = \alpha_0 + x_i\beta + \int_{\mathcal{S}} z_i(s)\gamma(s)ds$$

- $g(\cdot)$ is a link function
- α_0 is the intercept
- β is the linear association between x_i and y_i
- $\gamma(s)$ is the functional coefficient
    - "Linear" effect over the functional domain $\mathcal{S}$
    - Can be thought of as a weight function

GFLM

- Approximation of the integral term

$$
\begin{aligned}
g(E[y_i|x_i, \boldsymbol{Z}_i]) &= \alpha_0 + x_i\beta + \int_{\mathcal{S}} z_i(s)\gamma(s)ds \\
&= \alpha_0 + x_i\beta + \int_{\mathcal{S}} \left[\sum_{k=1}^{K_z} \xi_k^z \phi_k^z(s)\right] \left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s)\right] ds \\
&\approx \alpha_0 + x_i\beta + \sum_{j=1}^{J} l(s_j) \left[\sum_{k=1}^{K_z} \xi_k^z \phi_k^z(s_j)\right] \left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s_j)\right]
\end{aligned}
$$

- Where $l(s_j)$ is the quadrature weight associated with the numeric approximation method

GFLM

- Basis Expansion(s)

$$
\begin{aligned}
g(E[y_i|x_i, \boldsymbol{Z}_i]) &= \alpha_0 + x_i\beta + \int_{\mathcal{S}} z_i(s)\gamma(s)ds \\
&= \alpha_0 + x_i\beta + \int_{\mathcal{S}} \left[\sum_{k=1}^{K_z} \xi_k^z \phi_k^z(s)\right] \left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s)\right] ds
\end{aligned}
$$

GFLM: fPCA Basis

- One option: use the same basis for z and γ
- Convenient: fPC basis

$$g(E[y_i|x_i, \boldsymbol{Z}_i]) = \alpha_0 + x_i\beta + \int_{\mathcal{S}} \left[\sum_{k=1}^{K_z} \xi_k^z \phi_k^z(s) \right] \left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s) \right] ds$$

$$= \alpha_0 + x_i\beta + \sum_{k=1}^{K_z} \xi_{ik}^z \xi_k^\gamma$$

- (generalized) linear regression on the PC scores!
- Because $\int \phi_k^z(s)\phi_l^z(s) = 0$ if $k \neq l$ and 1 if $k = l$
- Choice of K_z becomes a tuning parameter

GFLM: fPCA Basis

SoFR

Estimation

In-Class Exercises

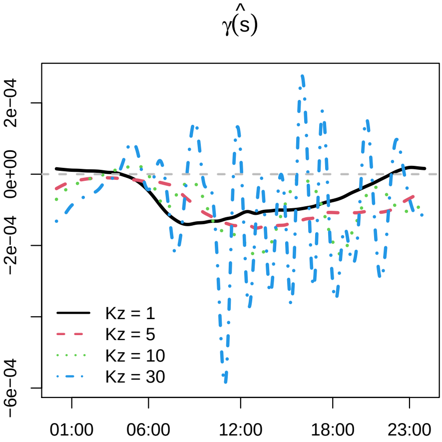
```
df$xi_z <- I(fpca_Z$scores)
Kz <- c(1,5,10,30)
coef_mat <- matrix(NA, length(Kz), 1440)
for(k in seq_along(Kz)){
  K_k    <- Kz[k]
  efuncs_k <- fpca_Z$efunctions[,1:K_k,drop=F]
  fit_k    <- gam(mort_5yr ~ df$xi_z[,1:K_k,drop=F], data=df)
  coef_mat[k,] <- efuncs_k %*% coef(fit_k)[-1]
}
```

GFLM: fPCA Basis

SoFR

Estimation

In-Class Exercises



GFLM: Penalized Splines

- "Fix" $z_i(s)$

$$
\begin{aligned}
g(E[y_i|x_i, \boldsymbol{Z}_i]) &= \alpha_0 + x_i\beta + \int_{\mathcal{S}} z_i(s)\gamma(s)ds \\
&= \alpha_0 + x_i\beta + \int_{\mathcal{S}} z_i(s)\left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s)\right] ds \\
&\approx \alpha_0 + x_i\beta + \sum_{j=1}^{J} l(s_j) z_i(s_j)\left[\sum_{k=1}^{K_\gamma} \xi_k^\gamma \phi_k^\gamma(s_j)\right] \\
&= \alpha_0 + x_i\beta + \sum_{k=1}^{K_\gamma} \xi_k^\gamma \left[\sum_{j=1}^{J} l(s_j) z_i(s_j)\phi_k^\gamma(s_j)\right]
\end{aligned}
$$

- Where $l(s_j)$ is the quadrature weight associated with the numeric approximation method

GFLM: Penalized Splines


```
## set up the functional domain matrix
## mgcv will use this to construct the basis \phi_k^\gamma(s)
sind  <- seq(0,1,len=1440)
smat  <- matrix(sind, nrow(df), 1440, byrow=TRUE)
df$smat <- I(smat)
## set up the matrix of integration weights
df$lmat <- I(matrix(1/1440, nrow(df), 1440))
## multiply integration weights by the functional predictor
df$zlmat <- I(df$lmat*df$Zsm)
fit_fglm_ps  <- gam(mort_5yr ~ s(smat, by=zlmat, bs="cc",k=30), data=df,
                    method="REML", family=binomial)
```

GFLM: Penalized Splines

SoFR

Estimation

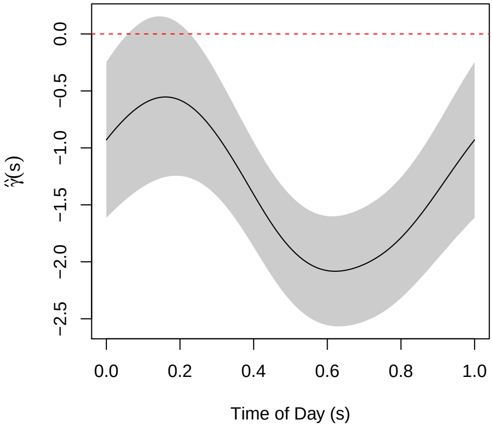
In-Class Exercises

```
summary(fit_fglm_ps)

##
## Family: binomial
## Link function: logit
##
## Formula:
## mort_5yr ~ s(smat, by = zlmat, bs = "cc", k = 30)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.8465     0.1971   4.295 1.75e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf Ref.df Chi.sq p-value
## s(smat):zlmat 3.022  3.462  204.5  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0804   Deviance explained = 9.64%
## -REML =   1089  Scale est. = 1         n = 3425
```

GFLM: Penalized Splines

SoFR
Estimation
In-Class Exercises



GFLM: Penalized Splines

SoFR

Estimation

In-Class Exercises

```
library("pROC")
(roc_in_sample <- roc(df$mort_5yr, fit_fglm_ps$fitted.values))

##
## Call:
## roc.default(response = df$mort_5yr, predictor = fit_fglm_ps$fitted.values)
##
## Data: fit_fglm_ps$fitted.values in 3042 controls (df$mort_5yr 0) < 383 cases (d
## Area under the curve: 0.7252

auc(roc_in_sample)

## Area under the curve: 0.7252
```

In-Class Exercises

1. Fit the unadjusted model using non-cyclic splines.
    - Do you see any differences?
    - Which model predicts better in terms of AUC?
2. Compare the fPC approach for $K_z = 1, 5, 10$ to the penalized regression approach using
    - In-sample AUC
    - 5-fold cross validation

References I