Functional Response Models

Response is a set of curves \( y_i(t) \) \( i = 1, \ldots, n \).

Covariates may be
- group labels
- scalar values
- functions

Today we will focus on the first two.

Partitioning Effects

Just as in the standard ANOVA, let

\[ x_{ij}(t) = \text{ith curve in jth group} \]

with \( n_j \) curves in group \( j \).

- An over-all mean
  \[ \bar{x}(t) = \frac{1}{\sum n_j} \sum_{j=1}^{K} \sum_{i=1}^{n_j} x_{ij} \]

- effects for each group
  \[ \alpha_j(t) = \frac{1}{n_j} \sum_{i=1}^{n_j} (x_{ij}(t) - \bar{x}(t)) \]

- an error process
  \[ e_{ij}(t) = x_{ij}(t) - \alpha_j(t) - \bar{x}(t) \]

Penetration Data

Penetration ANOVA
Penetration Residuals

Residuals display considerable heteroskedasticity.

\[ x_{ij}(t) = \mu(t) + \alpha_j(t) + \epsilon_{ij}(t) \]

Suppose we observe scalar covariates \( z_{i1}, \ldots, z_{ip}, \ i = 1, \ldots, n \)

The functional model is

\[ y_i(t) = \beta_0(t) + \sum_{j=1}^{p} \beta_j(t)z_{ij} + \epsilon_i(t) \]

and we assume that the \( \epsilon_i(t) \) are random error processes with at least

\[ E\epsilon_i(t) = 0 \]

That is, we now have coefficient functions.

Fitting a Model by Least Squares

At each time \( t \) we have

\[ y_i(t) = z_i\beta(t) + \epsilon_i(t) \]

so we can solve the least-squares equations to get

\[ \beta(t) = (Z^TZ)^{-1}Z^Ty(t) \]

But we would like to represent \( \beta(t) \) as a functional data object.

General Scalar Covariates

ANOVA Model

\[ x_{ij}(t) = \mu(t) + \alpha_j(t) + \epsilon_{ij}(t) \]

Basis Expansions

More generally, we want to represent

\[ \beta_j(t) = \Phi_j(t)c_j \]

then we need a new least-squares criterion:

\[ SSE(\beta) = \sum_{i=1}^{n} \int (y_i(t) - z_i\beta(t))^2 \ dt \]

if the \( y_i(t) \) and the \( \beta(t) \) share the same basis, this is exactly the same as the point-wise solution.

However, sometimes we may want to make the \( \beta(t) \) smoother.
Some Mechanics

\[ \text{SSE}(\beta) = \sum_{i=1}^{n} \int (y_i(t) - z_i\beta(t))^2 dt \]

write

\[ b = [c_1^T \cdots c_p^T]^T \]

and

\[ \psi_i(t) = [z_i1\Phi_1(t) \cdots z_ipo(t)] \]

then

\[ \hat{b} = \left[ \sum \int \psi_i(t)\psi_i(t)^T dt \right]^{-1} \left[ \sum \int \psi_i(t)^T y_i(t) dt \right] \]

Mechanics for the Functional ANOVA

\[ x_i(t) = \mu(t) + \alpha_j(t) + \epsilon_i(t) \]

Requires a constraint:

\[ \sum_{j=1}^{k} \alpha_j(t) = 0, \forall t \]

Incorporated automatically in most software. 
\[ fda \] library requires manual support

Microwaves: Predicting Penetration from Year Introduced

Model: \( y_i(t) = \beta_0(t) + \beta_1(t)(\text{year} - 1977) + \epsilon_i(t) \)

\[ Z\text{list3} = \text{list()} \]
\[ Z\text{list3}[[1]] = \text{rep(1,N)} \]
\[ Z\text{list3}[[2]] = \text{penetration$yr\_intro[MW.\text{oven}]-1977} \]

\[ \text{betalist3} = \text{list()} \]
\[ \text{for}(i \in 1:2)\{ \text{betalist3}[[i]] = \text{fdPar(bbasis,2,0)} \} \]

\[ \text{MWres} = \text{fRegress(MWfd,Zlist3,betalist3)} \]
Penetration Differences in Whitegoods

\[ z_{mat} = \text{matrix}(0, N, K) \]
\[ \text{for}(i \in 1:K) \{ z_{mat}[\text{product}==i, i] = 1 \} \]
\[ z_{mat} = \text{cbind}(\text{rep}(1, N), z_{mat}) \]
\[ z_{mat} = \text{rbind}(z_{mat}, c(0, \text{rep}(1, K))) \]

\[ Zlist = \text{list()} \]
\[ \text{for}(i \in 1:(K+1)) \{ Zlist[[i]] = z_{mat}[, i] \} \]

\[ \text{coef} = \text{Sfd$coefs} \]
\[ \text{coef} = \text{cbind}(\text{coef}, \text{rep}(0, \text{nrow}(\text{coef}))) \]
\[ \text{Sfd} = \text{fd}(\text{coef}, \text{bbasis}) \]
\[ \text{Sfd$fdnames}[[3]] = \text{"penetration"} \]
\[ \text{Sfd$fdnames}[[1]] = \text{"year"} \]

\[ \text{betalist} = \text{list()} \]
\[ \text{for}(i \in 1:(K+1)) \{ \text{betalist}[[i]] = \text{fdPar}(\text{bbasis}, \text{int2Lfd}(0), 0) \} \]

\[ \text{AOVres} = \text{fRegress}($\text{Sfd}, Zlist, betalist) \]

Error Covariance

From the model
\[ y_i(t) = z_i\beta(t) + \epsilon_i(t) \]
we can fit
\[ \hat{\epsilon}_i(t) = y_i(t) - z_i\hat{\beta}(t) \]
and set
\[ C(s, t) = \frac{1}{n-p} \sum \epsilon_i(t)\epsilon_i(s) \]

Error Covariance for Microwaves

\[ \text{MWsig} = \text{var}($\text{MWfd-$\text{MWres}$$yhatfdobj}$) \]
\[ \text{tfine} = \text{seq}(0, 20, 0.1) \]
\[ \text{MWmat} = \text{eval.bifd}($\text{tfine, tfine, MWbifd}$) \]

\[ \text{library(rgl)} \]
\[ \text{persp3d}($\text{tfine, tfine, MWmat, col='lightblue'}$) \]

Confidence Intervals

We have
\[ \hat{\beta} = \sum_i \int S_i(t)y_i(t)dt \]
with
\[ \text{Cov}[y_i(t), y_i(s)] = C(s, t) \]
then
\[ \text{var}[\hat{\beta}] = \sum_i \int S_i(s)C(s, t)S_i(t)dsdt \]
and
\[ \text{var}[\beta_j(t)] = [0 \Phi_j(t) 0\text{var}[^{\text{b}}] \left[ \begin{array}{c} 0 \\ \Phi_j(t)^T \\ 0 \end{array} \right] \]
Confidence Intervals II

But isn’t \( y_i(t) \) already estimated? And shouldn’t we account for that?

Suppose that \( y_i = y_{i1}, \ldots, y_{iN} \)

\[ y_i(t) = \Phi(t)C\lambda y_i \]

then

\[
\hat{b} = \sum_i \int S_i(t)\Phi(t)dt C\lambda y_i \\
= [S_1(t)\Phi(t)dt \cdots S_n(t)\Phi(t)dt] \begin{bmatrix} C\lambda & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & C\lambda \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \\
= c2bMap \circ y2cMap \ y
\]

Confidence Intervals II

If the \( y_i \) all share the same measurement times \( t \), we can look at

\[ \epsilon_i = y_i - z_i\hat{\beta}(t) \]

and estimate

\[ \hat{\Sigma} = \frac{1}{n - K} \sum \epsilon_i\epsilon_i^T \]

then

\[ \text{var} [\hat{b}] = c2bMap \circ y2cMap \begin{bmatrix} \hat{\Sigma} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \hat{\Sigma} \end{bmatrix} y2cMap^T \circ c2bMap^T \]

Covariance from Year Introduced

\[
M\hat{w}d = \text{smooth.basis}(0:20, MW, MWfdPar) \\
yhatmat3 = \text{eval.fd}(0:20, MWres$yhatfdobj) \\
MWerr = MW-yhatmat3 \\
SigmaE3 = MWerr%*%t(MWerr)/45 \\
MWres.std = fRegress.err(res3, MWfd$y2cMap, SigmaE3)
\]

Confidence Intervals III

The code is set up assuming we have \( y2cMap \).

What if we don’t? Or what if the \( y_i \) are not measured at the same times?

A first option is to ignore the estimation of \( y_i(t) \).

How?

\[
c = \left[ \Phi(t)^T\Phi(t) \right]^{-1} \Phi(t)^T\Phi(t)\Sigma
\]

so evaluate \( y_i(t) \) and then use an unpenalized smooth to get \( y2cMap \).

NOTE: \( t \) must have as many points as there are basis functions.
Product Effect Confidence Intervals

```r
> yhatmat = eval.fd(seq(0,20,0.5),AOVres$yhatfdobj)
> ymat = eval.fd(seq(0,20,0.5),Sfd$fd)
> AOVerr = ymat-yhatmat[,1:99]

> uPar = fdPar(bbasis,2,0)
> y2cmap = smooth.basis(seq(0,20,0.5),ymat,uPar)$y2cMap

> SigmaE = err%*%t(err)/96
> AOVres.std = fRegress.stderr(res,y2cmap,SigmaE)
```

Functional $R^2$

Usual statistic is

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

here we have a functional equivalent

```r
> Rfd = 1-mean( (MWfd-MWres$yhatfdobj)^2)*
     mean((center.fd(MWfd))^2)^(-1)
```

Residual Diagnostics

Plots of functional residuals against scalar covariates are possible, re-scaling residuals may be helpful.

```r
cerr = solve(diag(sqrt(apply(MWerr^2,1,mean))),MWerr)
plot3d(0:20,Zlist3[[2]][1],cerr[,1],type='l')
for(i in 2:20){
  lines3d(0:20,Zlist3[[2]][i],cerr[,i],add=TRUE,lwd=2,col=4)
}
Significance

To test significance, we can define a pointwise $F$-statistic

$$F(t) = \frac{\text{Var}(\hat{y}(t))}{\sum (y_i - \hat{y}_i)^2}$$

indicates where there is a large amount of signal relative to variance.

Test over-all regression significance based on

$$F^* = \max F(t)$$

Permutation Test

Do $B$ times

1. Permute the indexes $1, \ldots, n$ to get $i_1, \ldots, i_n$
2. Define $y^b_j(t) = y_{i_j}(t)$
3. Estimate the model using $y^b(t)$ as the response.
4. Measure $F^*_b$.

If $\frac{1}{B} \sum_b I(F^* - F^*_b) > \alpha$ reject $H_0: \forall t: Ey(t) = 0$

Penetration Data

First we remove the main effect and the zero response

```r
> tSfd = Sfd[-100]
> tZlist = Zlist
> for(i in 1:length(Zlist)){tZlist[[i]] = tZlist[[i]][-100] }
```

```r
> fres = Fperm.fd(tSfd,tZlist[-1],betalist[-1])
```

ANOVA Regression on Year

A Two-Way ANOVA

We have measurements for each of the goods for each of 14 countries

```r
> countries
[1] Algeria Argentina Bolivia Chile China
[6] Colombia Egypt India Indonesia Jordan
```

```r
> Zlist2 = list()
> Zlist2[[1]] = c(rep(1,length(C2)),0,0)
> for(i in 1:K1){ Zlist2[[i+1]] = c(1*(product==i),1,0) }
> for(i in 1:K2){ Zlist2[[i+K+1]] = c(1*(country==i),0,1) }
```

```r
> betalist2=list()
> for(i in 1:length(Zlist2)){ betalist2[[i]] = fdPar(bbasis,3,0)
```

```r
> Sfd2 = fd(cbind(Sfd2$coef,rep(0,nrow(Sfd2$coef))),bbasis)
```

```r
> AOVres2 = fRegress(Sfd2,Zlist2,betalist2)
```
Some Diagnostics

Summary

- Functional responses regressed on scalar covariates \(\Rightarrow\) just linear regression at each time \(t\).
- Basis expansions make things more interesting
- Covariance can be calculated taking into account smoothing to produce \(y(t)\).
- Permutation \(F\) tests for over-all significance.
- Still to come – smoothing, random-effects view of functional ANOVA, functional covariates.