f0Gamms

```r
require(tidyverse)
require(mgcv)
require(mgcViz)
require(itsadug)
```

Load prepared f0 data

```r
setwd("C:/Users/Helen/Desktop/Stats/Pruned3_big")

f0Data = read.csv("f0_ready_for_gamms.csv")

f0Data$speaker = as.factor(f0Data$speaker)
f0Data$label = as.factor(f0Data$label)
```

GAM for f0

```r
m1F0=bam(f0 ~ label + te(time, by=label) + s(speaker, bs="re")
         + s(time, speaker, bs="fs", m=1), data=f0Data)
```

```r
summary(m1F0)
```

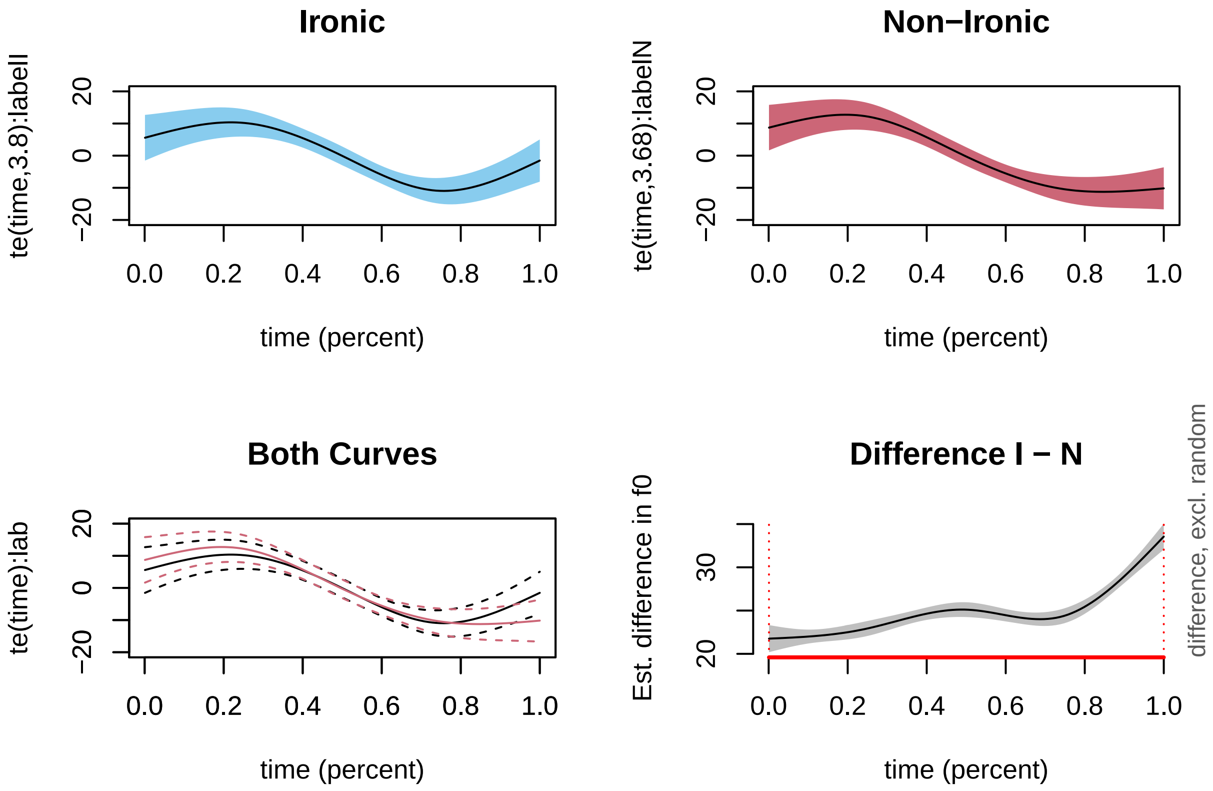
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## f0 ~ label + te(time, by = label) + s(speaker, bs = "re") + s(time,
##     speaker, bs = "fs", m = 1)
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 204.4273    12.0759   16.93   <2e-16 ***
## labelN      -24.9076     0.2002 -124.39   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Approximate significance of smooth terms:
##                     edf  Ref.df         F p-value
## te(time):labelI  3.799   3.843 1.169e+01  <2e-16 ***
## te(time):labelN  3.683   3.745 1.010e+01  <2e-16 ***
## s(speaker)       5.489  11.000 9.980e-01  <2e-16 ***
## s(time,speaker) 80.723 107.000 6.011e+05   0.225
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
##
## R-sq.(adj) =  0.185   Deviance explained = 18.5%
## fREML = 2.4338e+06  Scale est. = 4138.4    n = 435900
```

plots

```
cbPalette = c("#88CCEE", "#CC6677","#DDCC77","#117733","#332288","#AA4499",
              "#44AA99","#999933","#882255","#661100","#6699CC","#888888")

par(mfrow=c(2,2))
plot(m1F0, select=1, main = "Ironic", shade=TRUE,  xlab="time (percent)",
     shade.col = cbPalette[1], ylim=c(-20, 20))
plot(m1F0, select=2, main = "Non-Ironic", shade=TRUE,  xlab="time (percent)",
     shade.col = cbPalette[2], ylim=c(-20, 20))
plot(m1F0, select=1, ylab="te(time):lab", xlab="time (percent)",
     main = "Both Curves", ylim=c(-20, 20));
par(new=TRUE);
plot(m1F0, select=2, ylab="", xlab="", main="", col=cbPalette[2], ylim=c(-20, 20))
plot_diff(m1F0, view="time", shade=TRUE,
          xlab="time (percent)", comp=list(label=c("I", "N")))
```



```
f0Data$pred = predict(m1F0)
```

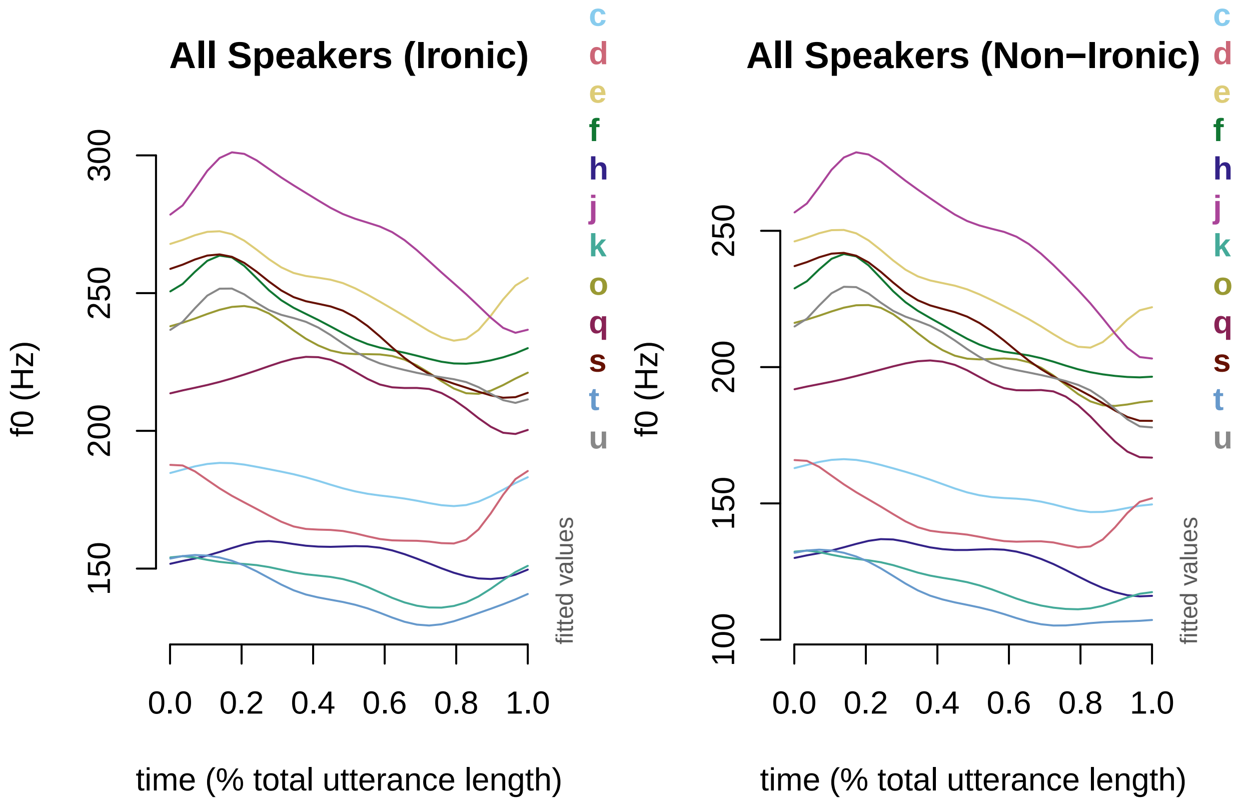
```
par(mfrow=c(1, 2))

plot_smooth(m1F0, view="time", cond=list("label"="I"), plot_all=c("speaker"),
            main = "All Speakers (Ironic)", col = cbPalette[1:12],
            rm.ranef=FALSE, se=0,
            ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

```
## Summary:
##  * label : factor; set to the value(s): I.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor with 12 values; set to the value(s): c, d, e, f, h, j, k, o, q, s, ...
```

```
plot_smooth(m1F0, view="time", cond=list("label"="N"), plot_all=c("speaker"),
            main = "All Speakers (Non-Ironic)", col = cbPalette[1:12],
            rm.ranef=FALSE, se=0,
            ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

```
## Summary:
##  * label : factor; set to the value(s): N.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor with 12 values; set to the value(s): c, d, e, f, h, j, k, o, q, s, ...
```



```r
par(mfrow=c(2, 2))

plot_smooth(m1F0, view="time", cond=list("speaker"="c"), plot_all=c("label"),
            main = "Speaker C", col = cbPalette[1:2], rm.ranef=FALSE, se=0,
             ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

```
## Summary:
##  * label : factor; set to the value(s): I, N.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor; set to the value(s): c.
```

```r
plot_smooth(m1F0, view="time", cond=list("speaker"="d"), plot_all=c("label"),
            main = "Speaker D", col = cbPalette[1:2], rm.ranef=FALSE, se=0,
             ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

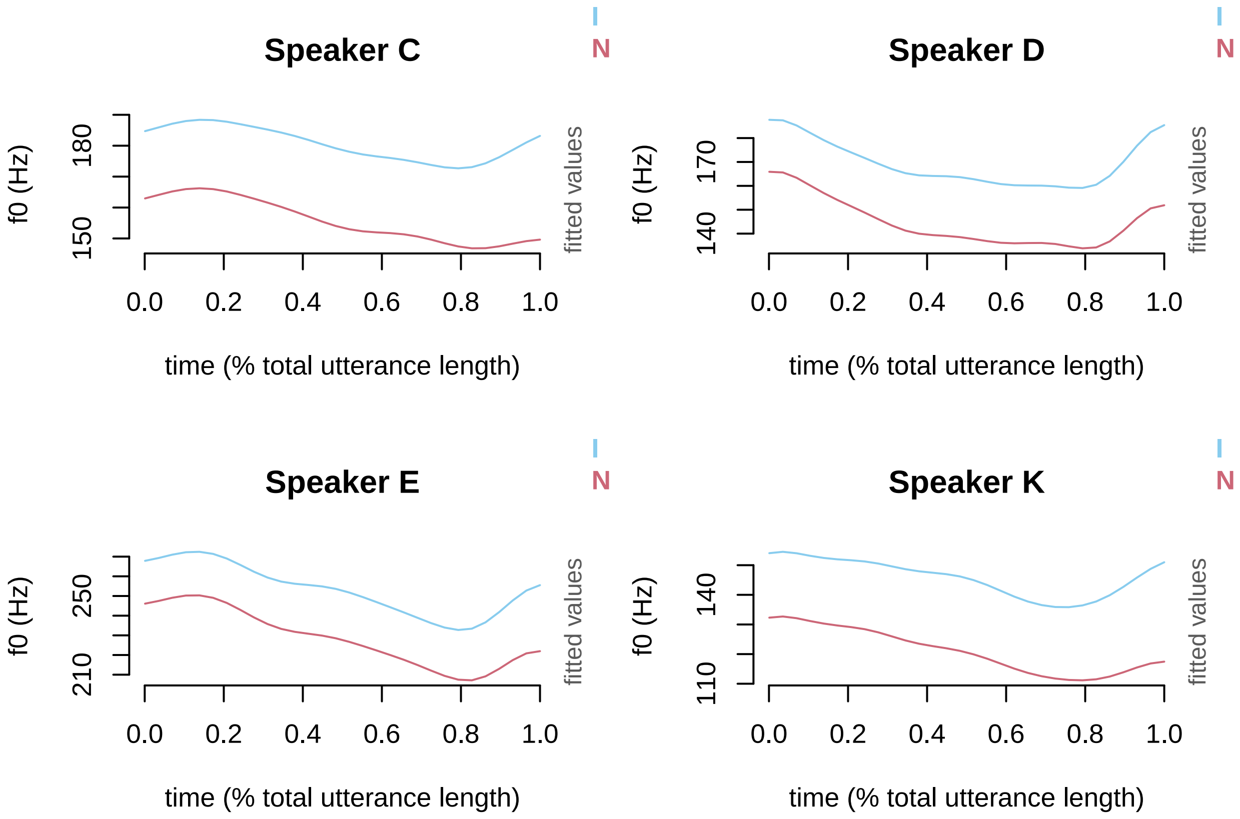
```
## Summary:
##  * label : factor; set to the value(s): I, N.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor; set to the value(s): d.
```

```r
plot_smooth(m1F0, view="time", cond=list("speaker"="e"), plot_all=c("label"),
            main = "Speaker E", col = cbPalette[1:2], rm.ranef=FALSE, se=0,
             ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

```
## Summary:
##  * label : factor; set to the value(s): I, N.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor; set to the value(s): e.
```

```r
plot_smooth(m1F0, view="time", cond=list("speaker"="k"), plot_all=c("label"),
            main = "Speaker K", col = cbPalette[1:2], rm.ranef=FALSE, se=0,
             ylab="f0 (Hz)", xlab="time (% total utterance length)")
```

```
## Summary:
##  * label : factor; set to the value(s): I, N.
##  * time : numeric predictor; with 30 values ranging from 0.000754 to 1.000000.
##  * speaker : factor; set to the value(s): k.
```


Speaker C
I
N
f0 (Hz)
150
180
fitted values
0.0
0.2
0.4
0.6
0.8
1.0
time (% total utterance length)
Speaker D
I
N
f0 (Hz)
140
170
fitted values
0.0
0.2
0.4
0.6
0.8
1.0
time (% total utterance length)
Speaker E
I
N
f0 (Hz)
210
250
fitted values
0.0
0.2
0.4
0.6
0.8
1.0
time (% total utterance length)
Speaker K
I
N
f0 (Hz)
110
140
fitted values
0.0
0.2
0.4
0.6
0.8
1.0
time (% total utterance length)