

# Tracegram pilot

*Stefano Coretta*

*May 17, 2017*

## 1 Data import

```
degg.tracing <- read_csv("./tracegram-pilot/results/results.csv") %>%
  separate(file, c("speaker", "phonation")) %>%
  mutate_if(is.character, as.factor) %>%
  mutate(phonation = factor(phonation, levels = c("modal", "breathy"))) %>%
  gather(peak, value, degg.maximum:degg.minimum)
```

```
## Parsed with column specification:
## cols(
##   file = col_character(),
##   token = col_integer(),
##   time = col_double(),
##   egg.minimum = col_double(),
##   degg.maximum = col_double(),
##   degg.minimum = col_double()
## )
```

```
degg.tracing.norm <- read_csv("./tracegram-pilot/results/results.csv") %>%
  separate(file, c("speaker", "phonation")) %>%
  mutate_if(is.character, as.factor) %>%
  mutate(phonation = factor(phonation, levels = c("modal", "breathy"))) %>%
  group_by(speaker) %>%
  mutate(maximum.norm = scale(degg.maximum),
         minimum.norm = scale(degg.minimum)) %>%
  gather(peak, value.norm, maximum.norm:minimum.norm)
```

```
## Parsed with column specification:
## cols(
##   file = col_character(),
##   token = col_integer(),
##   time = col_double(),
##   egg.minimum = col_double(),
##   degg.maximum = col_double(),
##   degg.minimum = col_double()
## )
```

```
## Warning: attributes are not identical across measure variables; they will
## be dropped
```

```
value.norm <- degg.tracing.norm$value.norm
degg.tracing <- cbind(degg.tracing, value.norm)
```

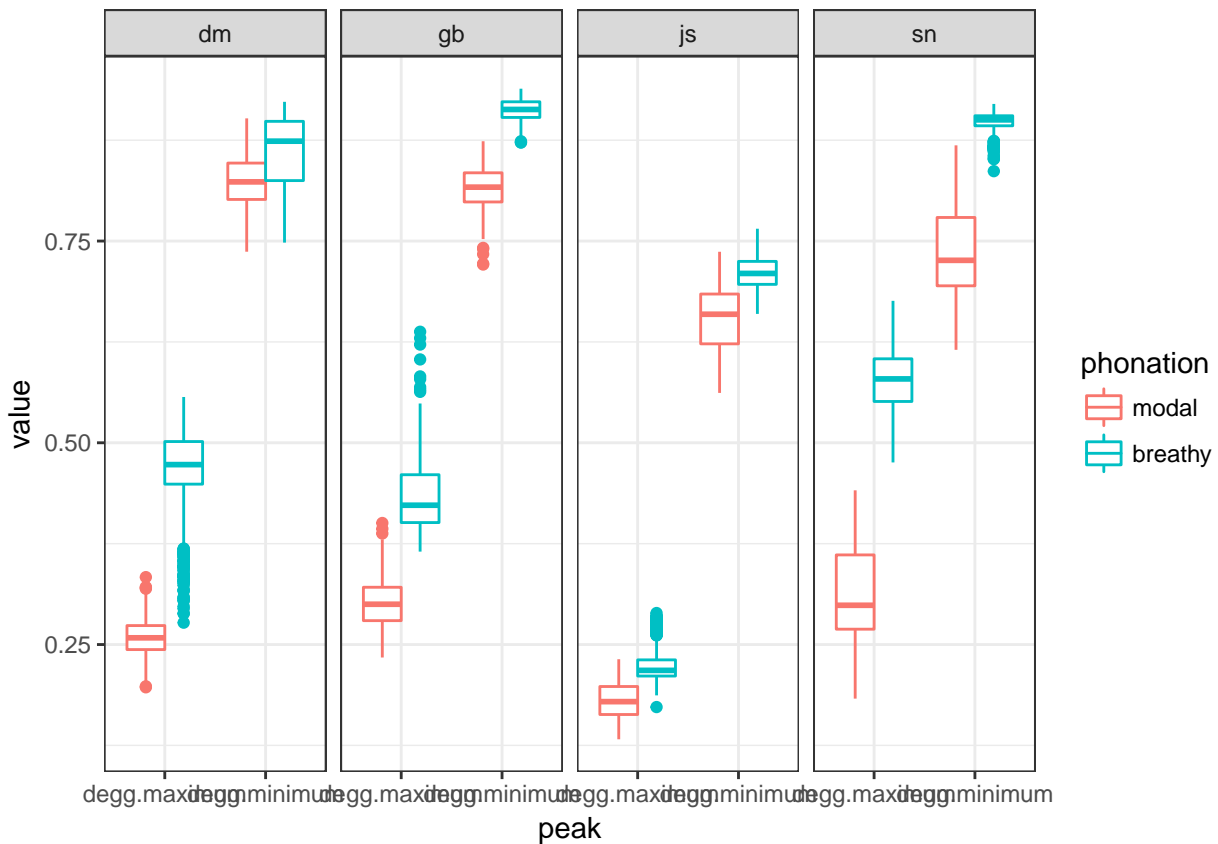
```
quotient <- read_csv("./tracegram-pilot/results/results.csv") %>%
  separate(file, c("speaker", "phonation")) %>%
  mutate_if(is.character, as.factor) %>%
  mutate(phonation = factor(phonation, levels = c("modal", "breathy")),
```

```
closed.quotient = degg.minimum - degg.maximum
)
```

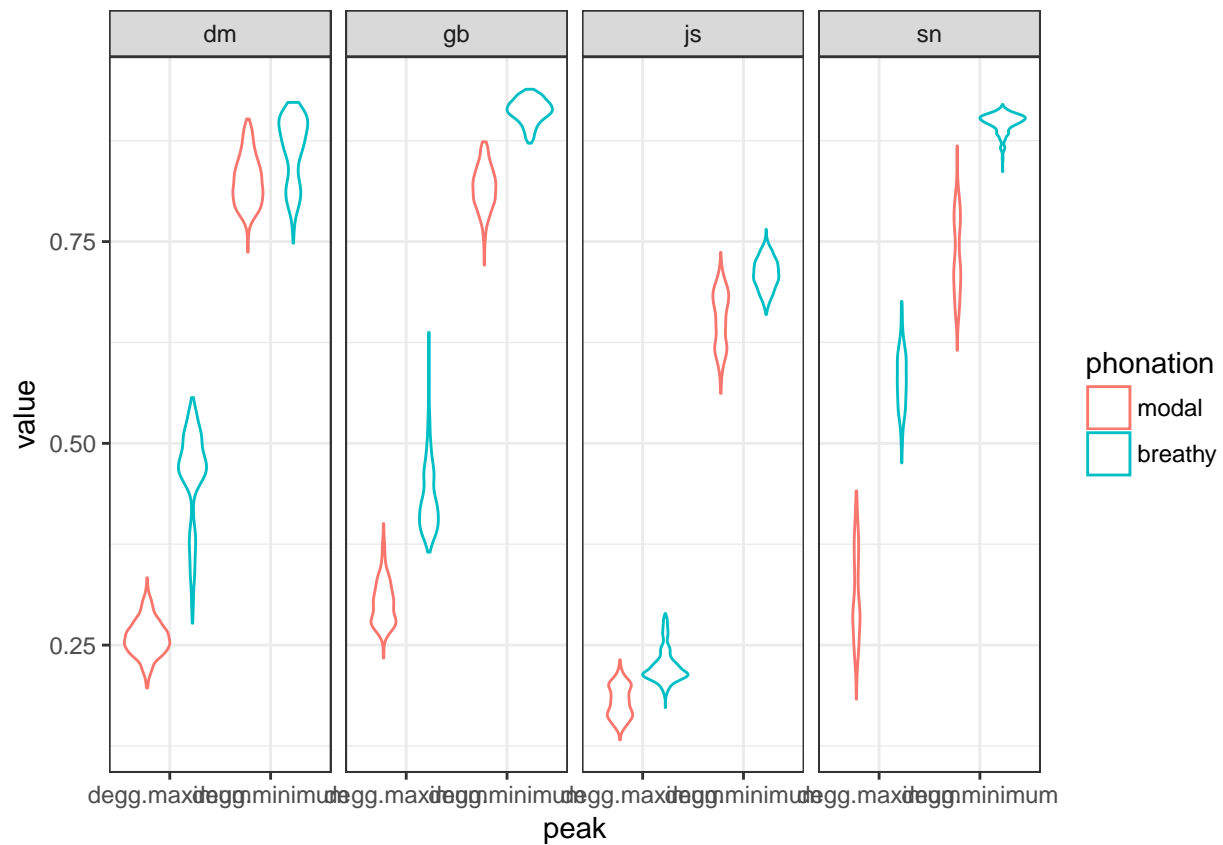
```
## Parsed with column specification:
## cols(
##   file = col_character(),
##   token = col_integer(),
##   time = col_double(),
##   egg.minimum = col_double(),
##   degg.maximum = col_double(),
##   degg.minimum = col_double()
## )
```

## 2 dEGG maxima and minima

```
ggplot(degg.tracing, aes(peak, value, colour = phonation)) +
  geom_boxplot() +
  facet_grid(. ~ speaker)
```

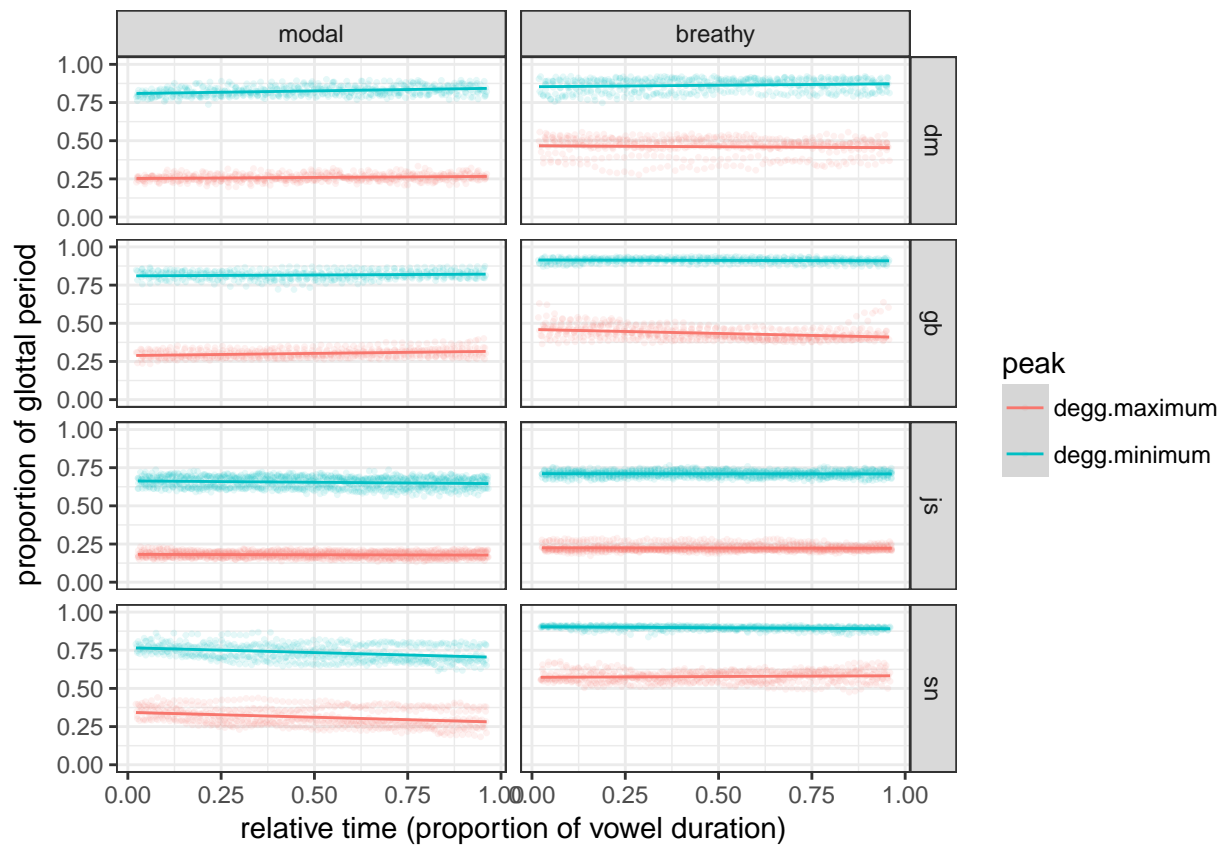


```
ggplot(degg.tracing, aes(peak, value, colour = phonation)) +
  geom_violin() +
  facet_grid(. ~ speaker)
```

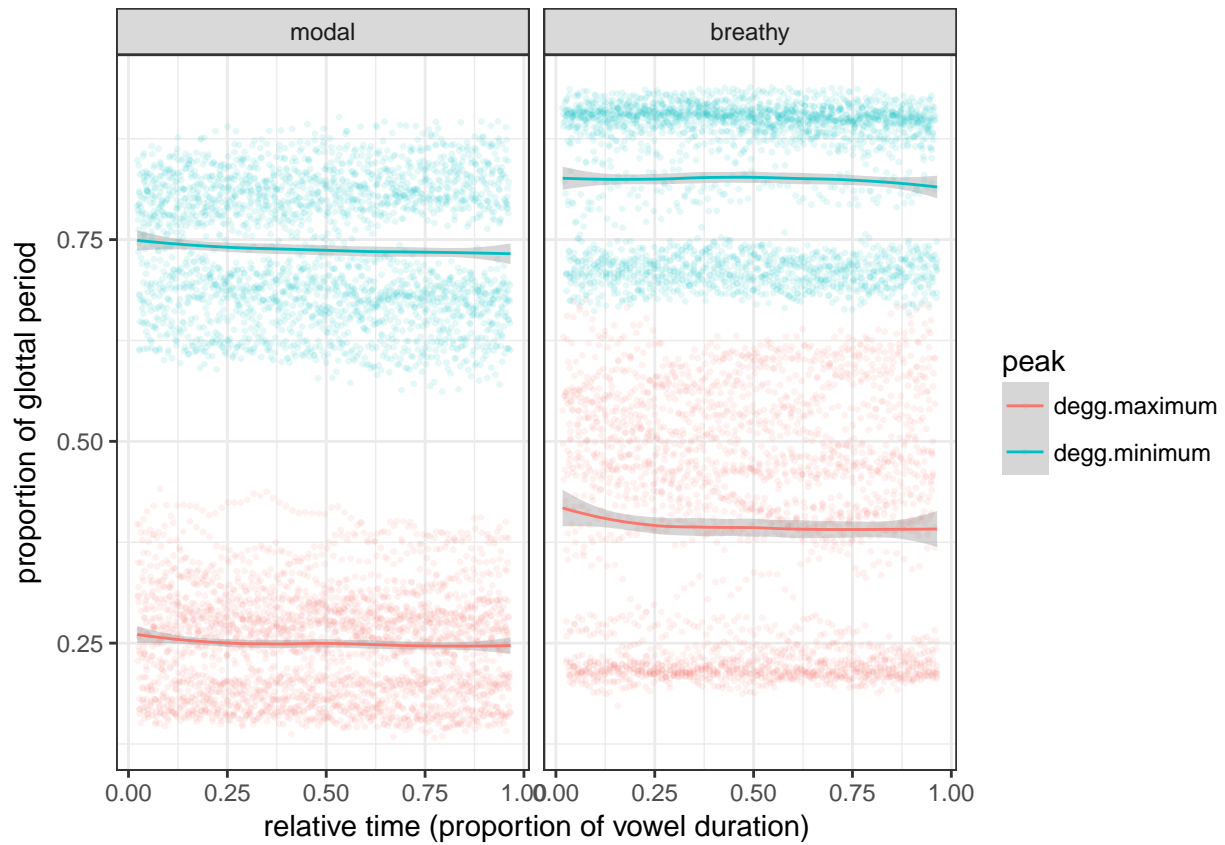


### 3 Tracegram plot

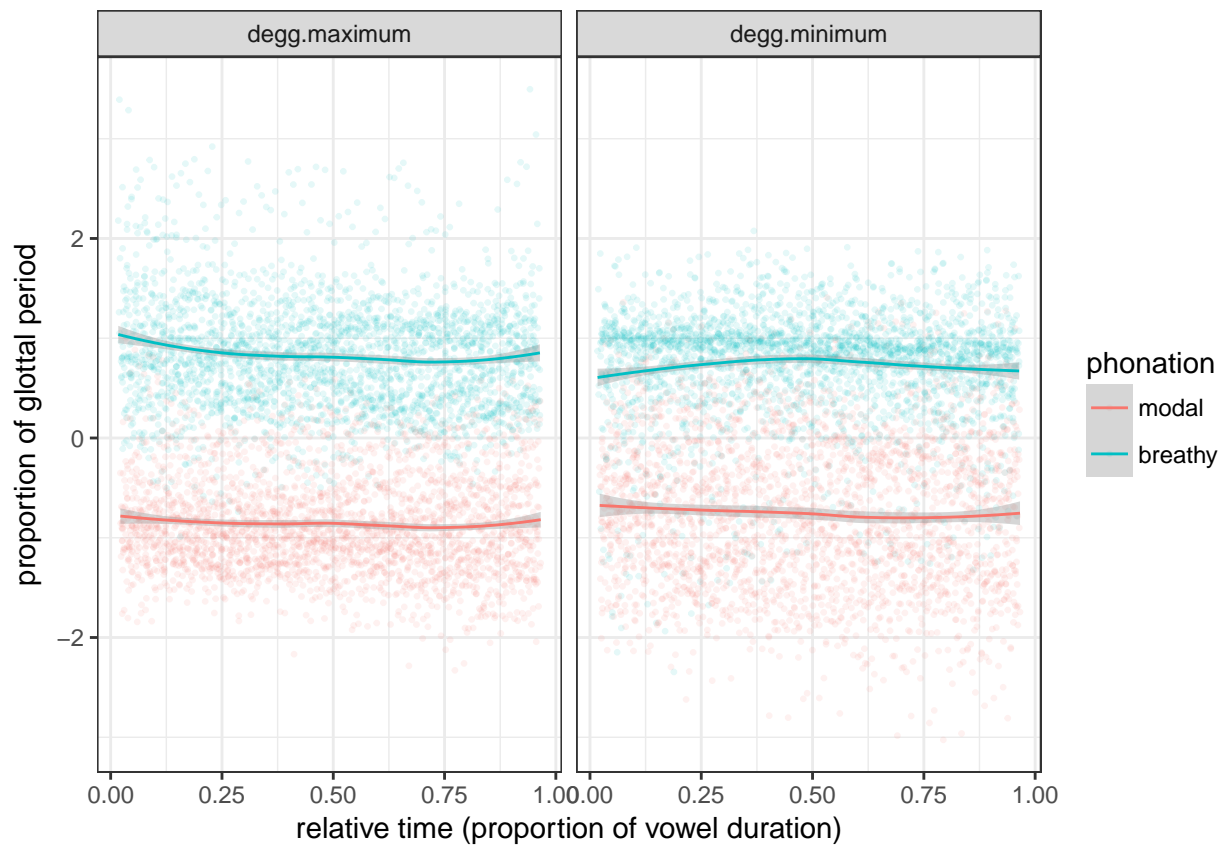
```
ggplot(degg.tracing, aes(time, value, colour = peak)) +
  geom_point(size = 0.5, alpha = 0.1) +
  geom_smooth(size = 0.5, method = "lm") +
  xlab("relative time (proportion of vowel duration)") +
  ylab("proportion of glottal period") +
  facet_grid(speaker ~ phonation) +
  ylim(0, 1)
```



```
ggplot(degg.tracing, aes(time, value, colour = peak)) +
  geom_point(size = 0.5, alpha = 0.1) +
  geom_smooth(size = 0.5, method = "loess") +
  xlab("relative time (proportion of vowel duration)") +
  ylab("proportion of glottal period") +
  facet_grid(. ~ phonation)
```

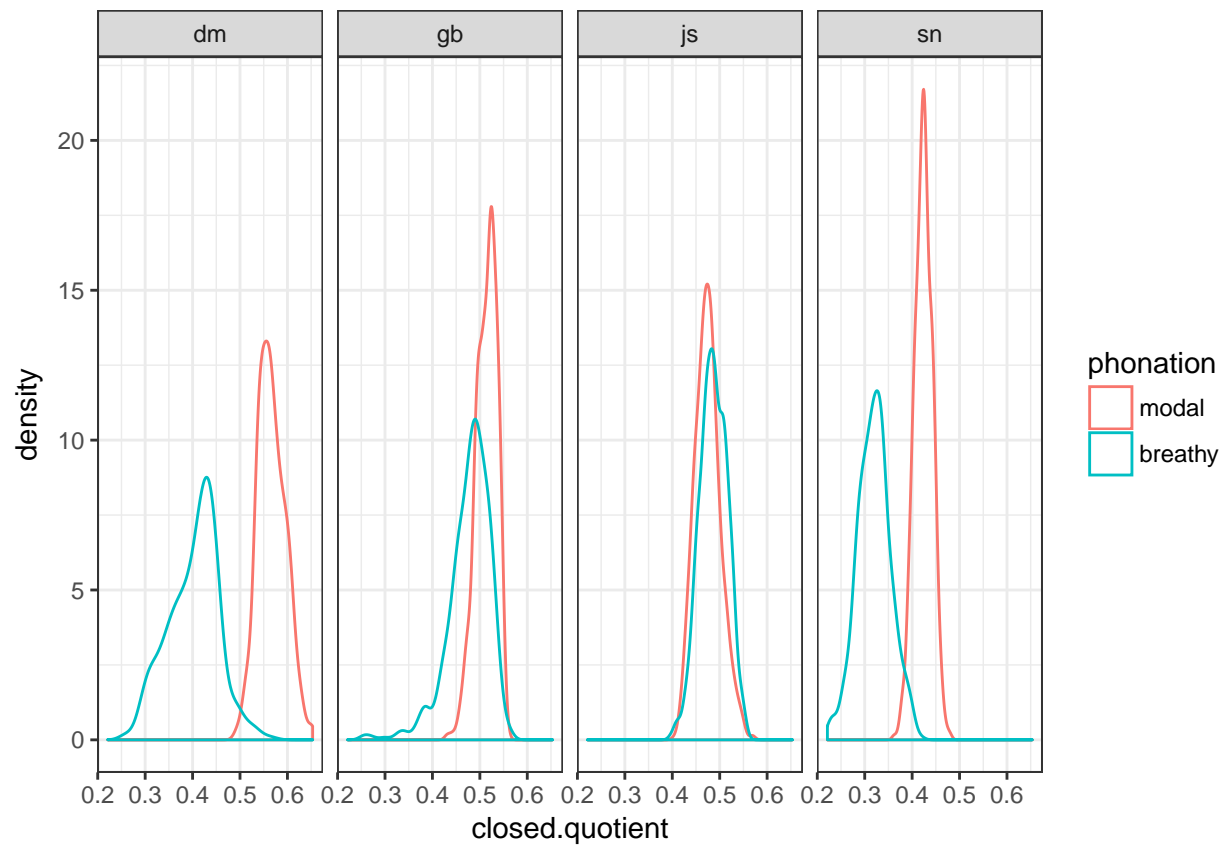


```
ggplot(degg.tracing, aes(time, value.norm, colour = phonation)) +
  geom_point(size = 0.5, alpha = 0.1) +
  geom_smooth(size = 0.5, method = "loess") +
  xlab("relative time (proportion of vowel duration)") +
  ylab("proportion of glottal period") +
  facet_grid(. ~ peak)
```

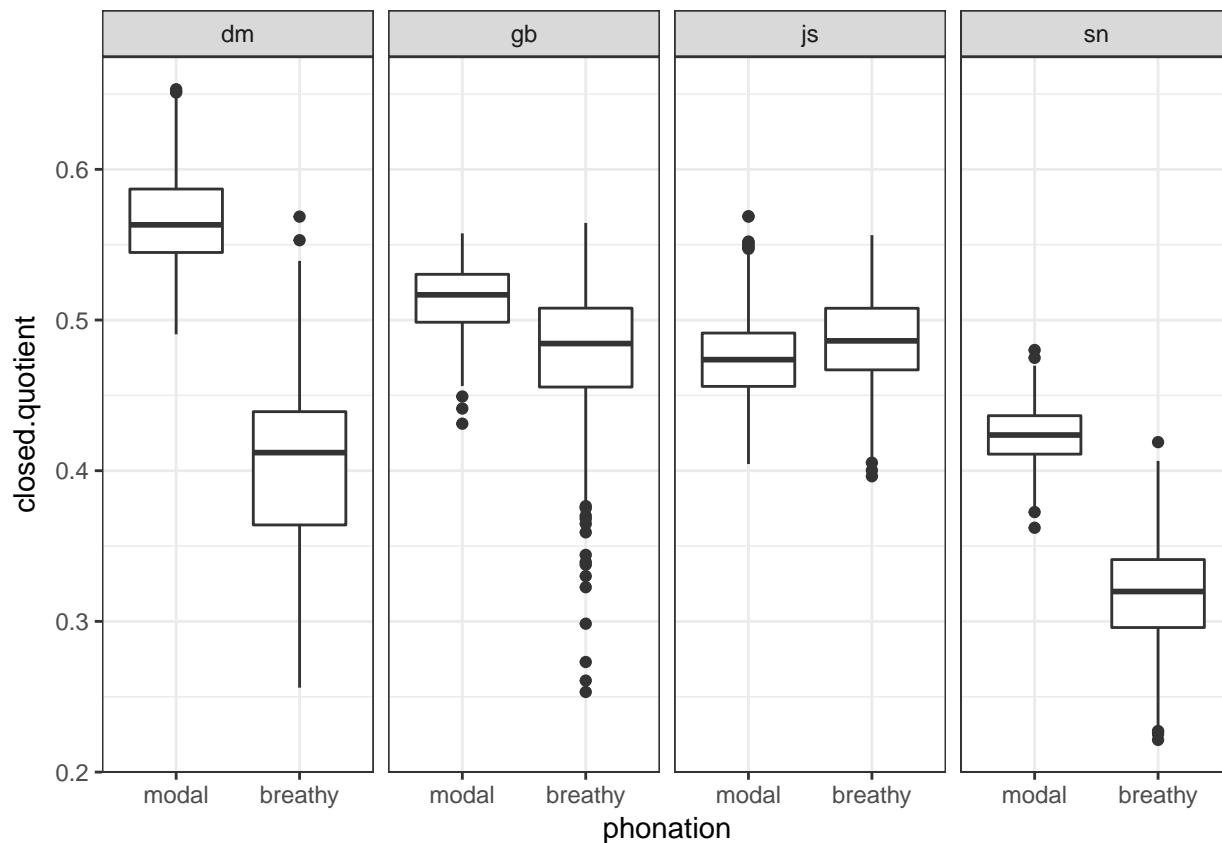


## 4 Closed quotient

```
ggplot(quotient, aes(closed.quotient, colour = phonation)) +
  geom_density() +
  facet_grid(. ~ speaker)
```



```
ggplot(quotient, aes(phonation, closed.quotient)) +
  geom_boxplot() +
  facet_grid(. ~ speaker)
```



```
quotient.lmer <- lmer(
  closed.quotient ~
    phonation +
    (1 + phonation|speaker),
  data = quotient
)
```

```
summary(quotient.lmer)
```

```
## Linear mixed model fit by REML t-tests use Satterthwaite approximations
## to degrees of freedom [lmerMod]
## Formula: closed.quotient ~ phonation + (1 + phonation | speaker)
## Data: quotient
##
## REML criterion at convergence: -19596.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -6.8237 -0.5875  0.0188  0.6320  5.0468
##
## Random effects:
##   Groups   Name                Variance Std.Dev. Corr
##   speaker  (Intercept)          0.003668 0.06057
##            phonationbreathy    0.005922 0.07696  -0.38
##   Residual                        0.001081 0.03289
## Number of obs: 4927, groups: speaker, 4
##
```



```

## Fixed effects:
##           Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    0.49460    0.03029  2.99860   16.33 0.000501 ***
## phonationbreathy -0.07312    0.03849  3.01330    -1.90 0.153295
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##           (Intr)
## phontnbnrthy -0.381

quotient.lmer.null <- lmer(
  closed.quotient ~
    (1 + phonation|speaker),
  data = quotient
)

anova(quotient.lmer.null, quotient.lmer)

## refitting model(s) with ML (instead of REML)

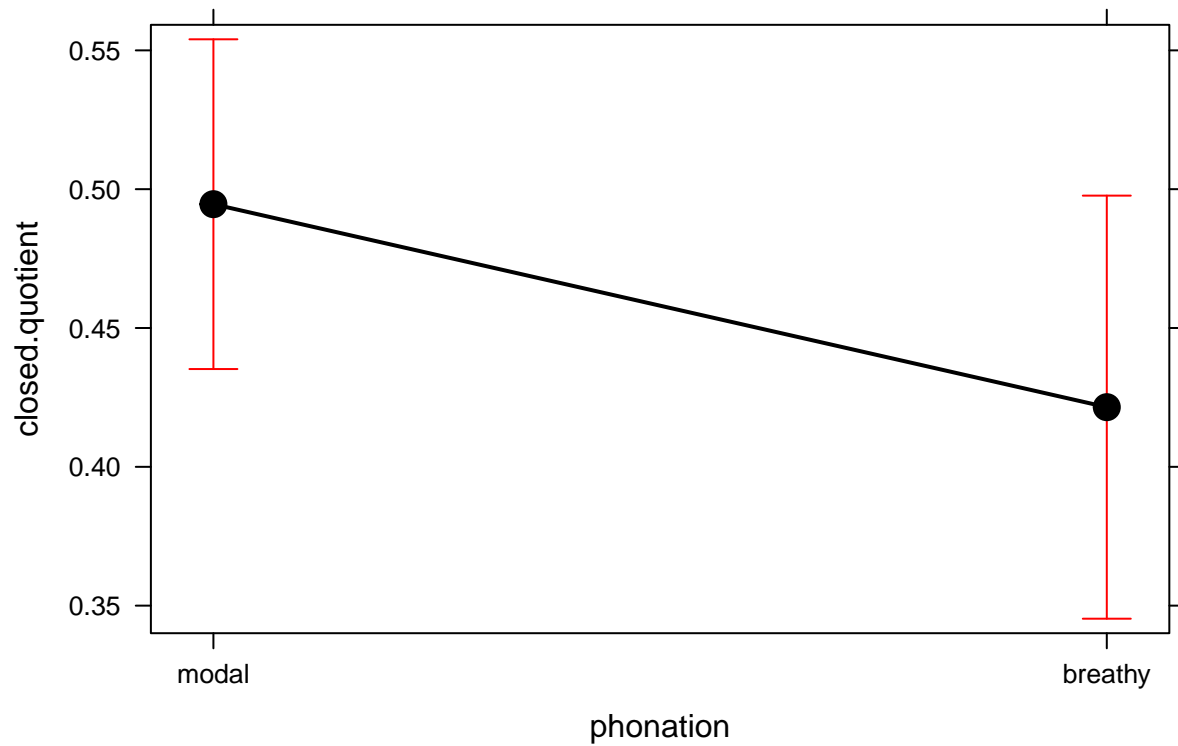
## Data: quotient
## Models:
## quotient.lmer.null: closed.quotient ~ (1 + phonation | speaker)
## quotient.lmer: closed.quotient ~ phonation + (1 + phonation | speaker)
##           Df      AIC      BIC logLik deviance Chisq Chi Df
## quotient.lmer.null  5 -19593 -19561 9801.6   -19603
## quotient.lmer       6 -19594 -19555 9803.2   -19606 3.159    1
##           Pr(>Chisq)
## quotient.lmer.null
## quotient.lmer      0.07551 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

mixed(
  closed.quotient ~
    phonation +
    (1 + phonation|speaker),
  data = quotient
)

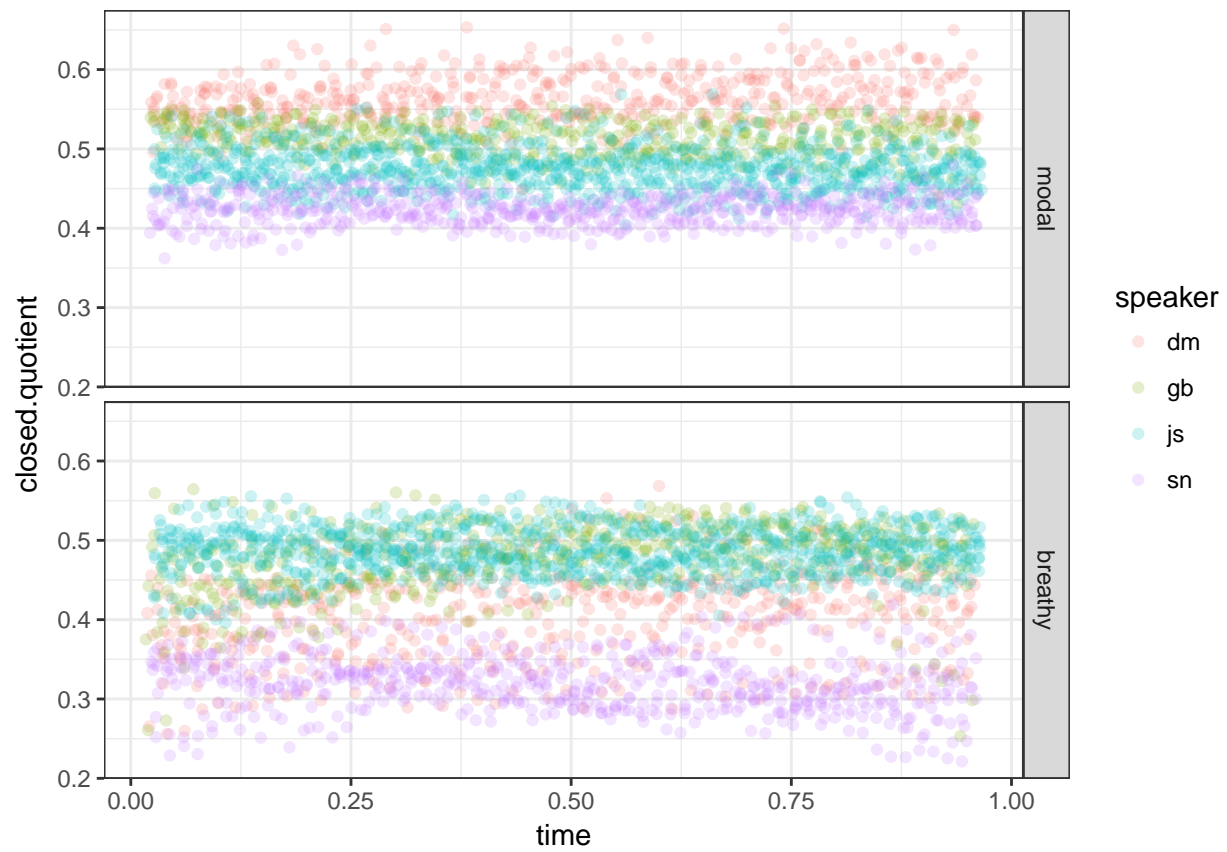
plot(allEffects(quotient.lmer))

```

## phonation effect plot



```
ggplot(quotient, aes(time, closed.quotient, colour = speaker)) +  
  geom_point(alpha = 0.2) +  
  facet_grid(phonation ~ .)
```



```
filter(quotient, token == 1) %>%
ggplot(aes(time, closed.quotient)) +
  geom_point(alpha = 0.2) +
  geom_smooth() +
  facet_grid(phonation ~ speaker)
```

```
## `geom_smooth()` using method = 'loess'
```

