

# French lexicon project: deciles sampling distributions & individual shift functions

*Guillaume A. Rousselet*

*2019-01-16*

## Contents

Lexical decision dataset	2
Compute shift functions for all participants	2
Superimpose all shift functions	3
20% trimmed mean group shift function	4
Median group shift function	4
Illustrate sampling distributions of decile differences	4
Summary figure . . . . .	5
Skewness of sampling distributions . . . . .	5
Kurtosis of sampling distributions . . . . .	6
Stochastic dominance . . . . .	6
Pattern across deciles . . . . .	6
Illustrate sampling distributions of Word deciles	7
Illustrate sampling distributions of Non-Word deciles	8

```
# dependencies
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 3.4.2
library(tibble)
```

```
## Warning: package 'tibble' was built under R version 3.4.3
library(rogme)
library(beepr)
```

```
## Warning: package 'beepr' was built under R version 3.4.4
source("../functions/skew.txt")
sessionInfo()
```

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS 10.14.2
##
## Matrix products: default
```

```
## BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_GB.UTF-8/en_GB.UTF-8/en_GB.UTF-8/C/en_GB.UTF-8/en_GB.UTF-8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] beeper_1.3      rogm_0.2.0     tibble_1.4.2  cowplot_0.9.1 ggplot2_3.0.0
##
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.19    pillar_1.3.0    compiler_3.4.0  plyr_1.8.4
## [5] bindr_0.1.1     tools_3.4.0     digest_0.6.12   evaluate_0.10.1
## [9] gtable_0.2.0    pkgconfig_2.0.2 rlang_0.2.2     rstudioapi_0.8
## [13] yaml_2.1.15     bindrcpp_0.2.2  withr_2.1.0     dplyr_0.7.6
## [17] stringr_1.2.0   knitr_1.17      rprojroot_1.2   grid_3.4.0
## [21] tidyselect_0.2.4 glue_1.3.0      R6_2.3.0        rmarkdown_1.8
## [25] purrr_0.2.5     magrittr_1.5    backports_1.1.1 scales_0.5.0
## [29] htmltools_0.3.6 assertthat_0.2.0 colorspace_1.3-2 stringi_1.1.6
## [33] lazyeval_0.2.1  munsell_0.4.3   crayon_1.3.4    audio_0.1-5.1
```

## Lexical decision dataset

Data from the French Lexicon Project. Click on “French Lexicon Project trial-level results with R scripts.zip”. The `.RData` dataset was created by applying the script `/code/getflprtdata.Rmd`.

```
# get data - tibble = `flp`
load("./data/french_lexicon_project_rt_data.RData")
# columns =
#1 = participant
#2 = rt
#3 = acc = accuracy 0/1
#4 = condition = word/non-word
np <- length(unique(flp$participant))
```

## Compute shift functions for all participants

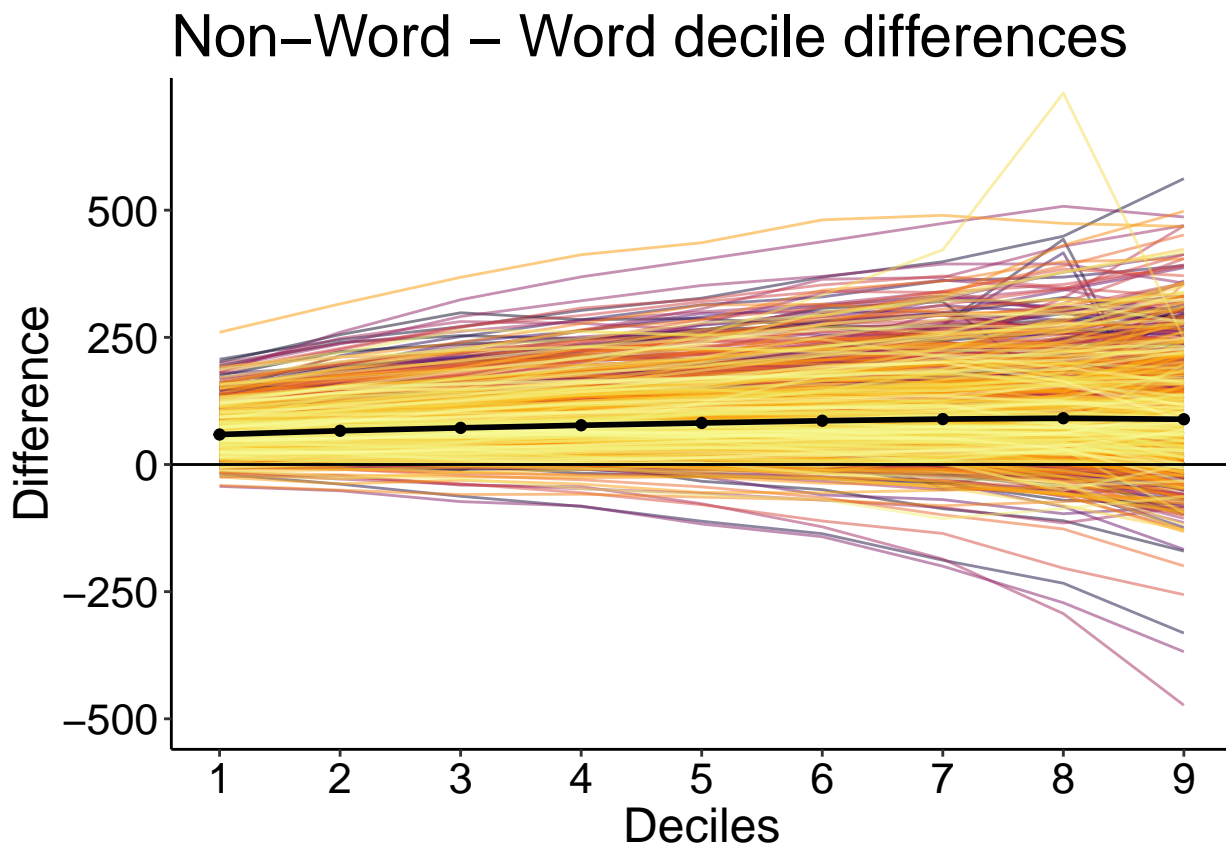
```
qseq <- seq(0.1,0.9,0.1)
# get data:
flp.s <- subset(flp, condition == "word")
q.w <- matrix(unlist(tapply(flp.s$rt, list(flp.s$participant), quantile, probs = qseq, type = 8)), nrow = np, byrow = TRUE)
flp.s <- subset(flp, condition == "non-word")
q.nw <- matrix(unlist(tapply(flp.s$rt, list(flp.s$participant), quantile, probs = qseq, type = 8)), nrow = np, byrow = TRUE)
q.diff <- q.nw - q.w # non-word - word
```

## Superimpose all shift functions

```
df <- tibble(diff = as.vector(q.diff),
             dec = rep(seq(1,9), np),
             participants = factor(rep(seq(1:np), each = 9)))

df.md <- tibble(diff = apply(q.diff, 1, mean, trim = 0.2),
               dec = seq(1,9))

p <- ggplot(df, aes(x = dec, y = diff, colour = participants)) + theme_classic() +
  geom_line(alpha = 0.5) +
  geom_abline(slope = 0, intercept = 0) +
  geom_line(data = df.md, colour = "black", size = 1) +
  geom_point(data = df.md, colour = "black") +
  scale_colour_viridis_d(option = "B") +
  scale_x_continuous(breaks = seq(1,9)) +
  scale_y_continuous(breaks = seq(-500,700,250)) +
  coord_cartesian(ylim = c(-500, 700)) +
  theme(legend.position = "none",
        plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18)) +
  labs(x = "Deciles", y = "Difference") +
  ggtitle("Non-Word - Word decile differences")
p
```



```
p.hsf <- p
```

## 20% trimmed mean group shift function

```
round(apply(q.diff, 1, mean, trim = 0.2))
```

```
## [1] 59 66 72 77 82 86 89 91 89
```

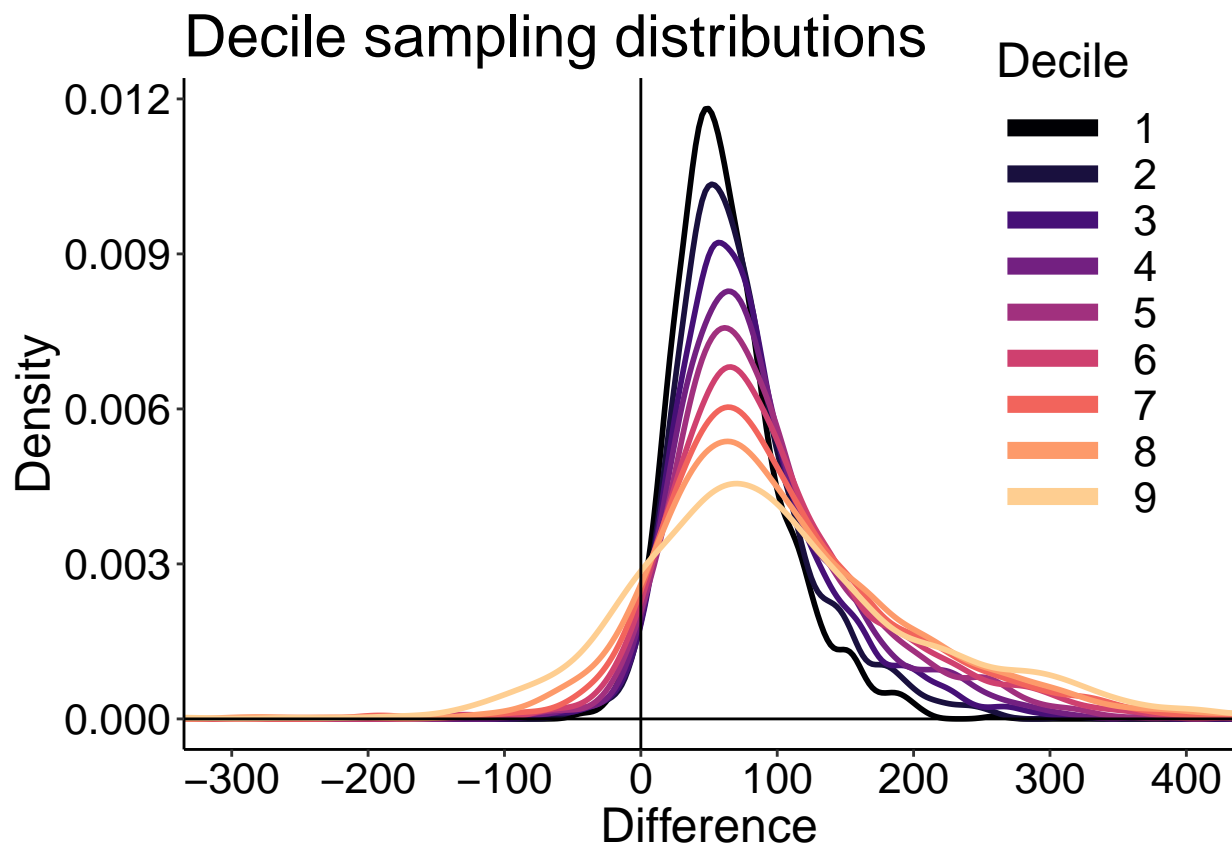
## Median group shift function

```
round(apply(q.diff, 1, median))
```

```
## [1] 58 64 70 74 78 83 84 86 86
```

## Illustrate sampling distributions of decile differences

```
df <- tibble(rt = as.vector(q.diff),
             Decile = factor(rep(seq(1,9), np)))
p <- ggplot(df, aes(x = rt, colour = Decile)) + theme_classic() +
  # geom_density(size = 1) +
  stat_density(geom = "line", position = "identity", size = 1) +
  scale_colour_viridis_d(option = "A", end = 0.9) +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0) +
  coord_cartesian(xlim = c(-300, 400)) +
  scale_x_continuous(breaks = seq(-300, 400, 100)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.7)) +
  guides(colour = guide_legend(override.aes = list(size = 3))) +
  xlab("Difference") +
  ylab("Density") +
  ggtitle("Decile sampling distributions")
p
```



```
p.sampdist <- p
```

### Summary figure

```
cowplot::plot_grid(p.hsf, p.sampdist,
  labels = c("A", "B"),
  ncol = 1,
  nrow = 2,
  label_size = 20,
  hjust = -0.5,
  scale=.95)

# save figure
ggsave(filename='./figures/figure_flp_dec_samp_dist.pdf',width=8,height=10)
```

### Skewness of sampling distributions

```
apply(q.diff, 1, skew)
```

```
## [1] 0.8407714 1.0289034 1.1077492 1.0800119 1.0082261 0.9693580 0.8334035
## [8] 0.8879903 0.4041340
```

## Kurtosis of sampling distributions

```
apply(q.diff, 1, kurt)

## [1] 4.230266 4.771753 5.115873 4.910816 4.724138 4.687401 4.520075 6.048764
## [9] 4.824009
```

## Stochastic dominance

Participants with all decile differences  $> 0$

```
pdmt0 <- apply(q.diff > 0, 2, sum)
sum(pdmt0 == 9)

## [1] 798
round(100 * sum(pdmt0 == 9) / np, digits = 1)

## [1] 83.2
```

Participants with all decile differences  $< 0$

```
pdlt0 <- apply(q.diff < 0, 2, sum)
sum(pdlt0 == 9)

## [1] 13
round(100 * sum(pdlt0 == 9) / np, digits = 1)

## [1] 1.4
```

## Pattern across deciles

Spearman function

```
spear<-function(x,y=NULL){
  # Compute Spearman's rho
  #
  if(!is.null(y[1])){
    m=elimna(cbind(x,y))
    n=nrow(m)
    x=m[,1]
    y=m[,2]
    corv<-cor(rank(x),rank(y))
  }
  if(is.null(y[1])){
    x=elimna(x)
    n=nrow(x)
    m<-apply(x,2,rank)
    corv<-cor(m)
  }
}
```

```

test <- corv * sqrt((n - 2)/(1. - corv^2))
sig <- 2 * (1 - pt(abs(test), length(x) - 2))
if(is.null(y[1])) sig <- matrix(sig, ncol = sqrt(length(sig)))
if(sig < 0.05){
  out <- sign(corr)
} else {
  out <- 0
}
out
}

```

### Participants with monotonic increase across deciles

```

qseq <- seq(0.1, 0.9, 0.1)
pdmi <- apply(q.diff, 2, spear, qseq)

sum(pdmi == 1)

## [1] 507
round(100 * sum(pdmi == 1) / np, digits = 1)

## [1] 52.9

```

### Participants with monotonic decrease across deciles

```

sum(pdmi == -1)

## [1] 143
round(100 * sum(pdmi == -1) / np, digits = 1)

## [1] 14.9

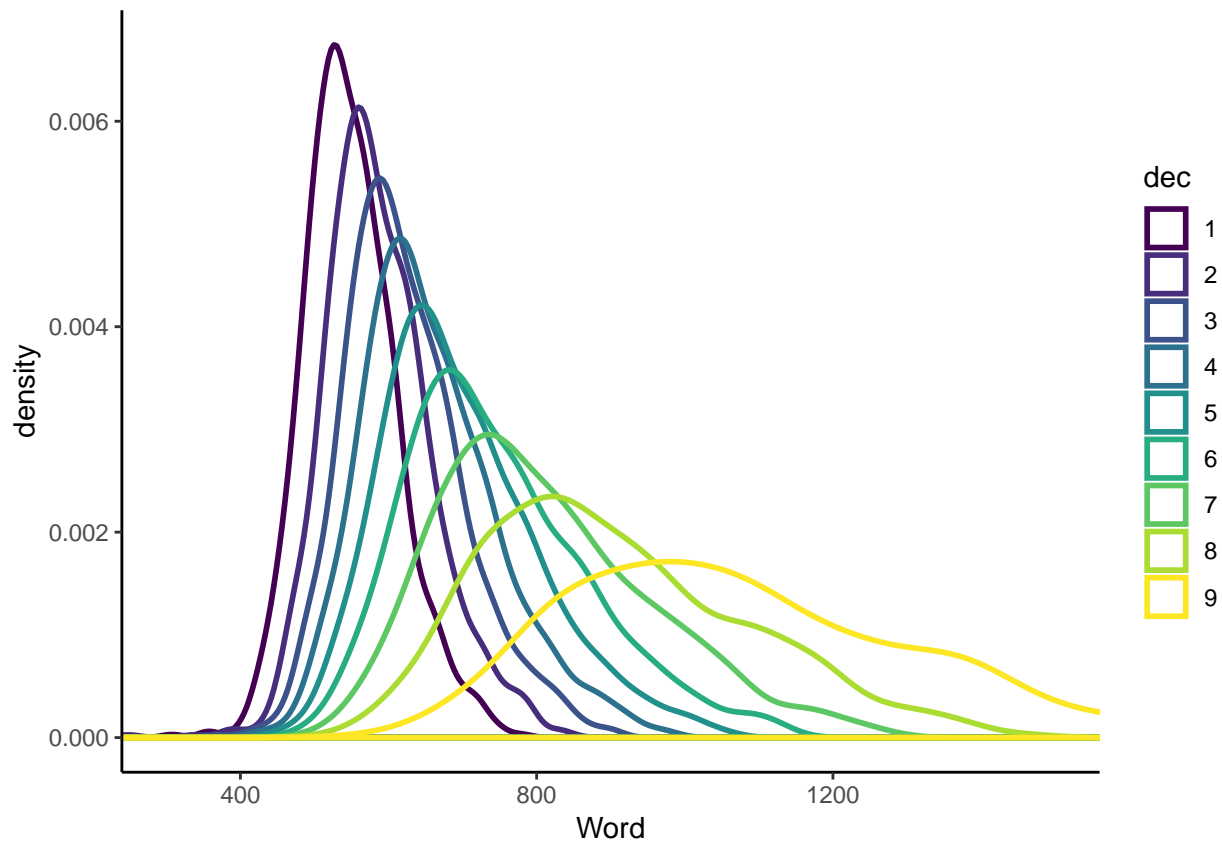
```

## Illustrate sampling distributions of Word deciles

```

df <- tibble(rt = as.vector(q.w),
             dec = factor(rep(seq(1, 9), np)))
ggplot(df, aes(x = rt, colour = dec)) + theme_classic() +
  geom_density(size = 1) +
  scale_colour_viridis_d() +
  coord_cartesian(xlim = c(300, 1500)) +
  xlab("Word")

```



### Skewness of sampling distributions

```
apply(q.w, 1, skew)
```

```
## [1] -0.05522597  0.38608525  0.58142236  0.66220389  0.68908574  0.68118856
## [7]  0.64509494  0.56786921  0.68698836
```

### Kurtosis of sampling distributions

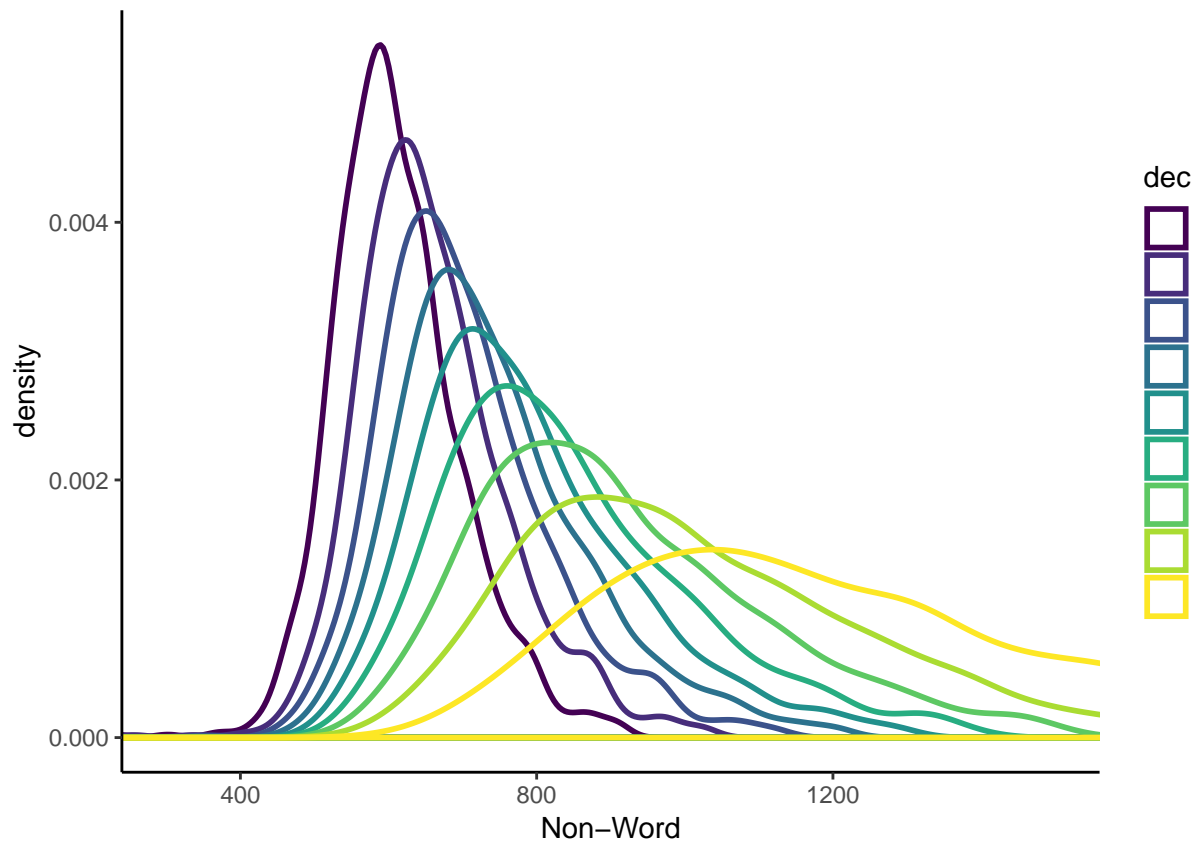
```
apply(q.w, 1, kurt)
```

```
## [1] 5.743937 3.886107 3.463645 3.406306 3.327856 3.197752 3.038115 2.813393
## [9] 3.450524
```

## Illustrate sampling distributions of Non-Word deciles

```
df <- tibble(rt = as.vector(q.nw),
              dec = factor(rep(seq(1,9), np)))
ggplot(df, aes(x = rt, colour = dec)) + theme_classic() +
  geom_density(size = 1) +
  scale_colour_viridis_d() +
  coord_cartesian(xlim = c(300, 1500)) +
  xlab("Non-Word")
```





### Skewness of sampling distributions

```
apply(q.nw, 1, skew)
```

```
## [1] 0.3010317 0.6730341 0.8372175 0.8838682 0.8708904 0.8294558 0.7557497
## [8] 0.7053264 0.5652509
```

### Kurtosis of sampling distributions

```
apply(q.nw, 1, kurt)
```

```
## [1] 4.821474 4.105147 3.987164 3.848922 3.680687 3.458316 3.201474 3.281279
## [9] 2.844496
```