Fig_3EFG

```
source("source.R")

path <- "data/dat_all.csv"
path_id <- "data/id.csv"

i <- 4
.pc <- 5
```

```
dat_raw <-
  path %>%
  read_csv()

dat_ngram <-
  dat_raw %>%
  arrange_ngram()

dat_id <-
  path_id %>%
  read_csv()
```

```
dat_ngram_i <-
  dat_ngram %>%
  select(type, Name, Stage, PM, str_c("gram", i)) %>%
  unnest(gram4) %>%
  pivot_wider(
    names_from = key,
    values_from = n,
    values_fill = 0
  )

fit_pca <-
  dat_ngram_i %>%
  select(!c(type, Name, Stage, PM)) %>%
  prcomp(scale = FALSE)

fit_pca %>% summary()
```

```
## Importance of components:
##                             PC1     PC2      PC3      PC4      PC5     PC6
## Standard deviation     124.3961 89.2483 13.05027 12.68617 11.78217 8.55388
## Proportion of Variance   0.6372  0.3280  0.00701  0.00663  0.00572 0.00301
## Cumulative Proportion    0.6372  0.9652  0.97219  0.97882  0.98454 0.98755
##                             PC7     PC8     PC9    PC10    PC11    PC12    PC13
## Standard deviation     7.37200 6.39692 5.08907 4.52516 4.24954 4.04519 3.54365
## Proportion of Variance 0.00224 0.00168 0.00107 0.00084 0.00074 0.00067 0.00052
```

```
## Cumulative Proportion  0.98979 0.99147 0.99254 0.99338 0.99413 0.99480 0.99532
##                           PC14    PC15    PC16    PC17   PC18    PC19    PC20
## Standard deviation     3.44295 3.14067 2.41617 2.33009 2.2193 2.06598 1.97258
## Proportion of Variance 0.00049 0.00041 0.00024 0.00022 0.0002 0.00018 0.00016
## Cumulative Proportion  0.99581 0.99621 0.99645 0.99668 0.9969 0.99705 0.99721
##                           PC21    PC22    PC23    PC24    PC25    PC26   PC27
## Standard deviation     1.87631 1.74663 1.73697 1.66567 1.63052 1.60897 1.5867
## Proportion of Variance 0.00014 0.00013 0.00012 0.00011 0.00011 0.00011 0.0001
## Cumulative Proportion  0.99736 0.99748 0.99761 0.99772 0.99783 0.99794 0.9980
##                           PC28    PC29    PC30    PC31    PC32    PC33    PC34
## Standard deviation     1.48409 1.42569 1.41357 1.35268 1.32835 1.31360 1.28069
## Proportion of Variance 0.00009 0.00008 0.00008 0.00008 0.00007 0.00007 0.00007
## Cumulative Proportion  0.99813 0.99822 0.99830 0.99838 0.99845 0.99852 0.99859
##                           PC35    PC36    PC37    PC38    PC39    PC40    PC41
## Standard deviation     1.25102 1.23107 1.19463 1.17564 1.12386 1.10212 1.08504
## Proportion of Variance 0.00006 0.00006 0.00006 0.00006 0.00005 0.00005 0.00005
## Cumulative Proportion  0.99865 0.99871 0.99877 0.99883 0.99888 0.99893 0.99898
##                           PC42    PC43    PC44    PC45    PC46    PC47    PC48
## Standard deviation     1.04802 1.03747 1.01628 0.97736 0.96910 0.93785 0.93484
## Proportion of Variance 0.00005 0.00004 0.00004 0.00004 0.00004 0.00004 0.00004
## Cumulative Proportion  0.99902 0.99907 0.99911 0.99915 0.99919 0.99923 0.99926
##                           PC49    PC50    PC51    PC52    PC53    PC54    PC55
## Standard deviation     0.92506 0.90036 0.89693 0.87044 0.85746 0.83369 0.80480
## Proportion of Variance 0.00004 0.00003 0.00003 0.00003 0.00003 0.00003 0.00003
## Cumulative Proportion  0.99930 0.99933 0.99936 0.99939 0.99943 0.99945 0.99948
##                           PC56    PC57    PC58    PC59    PC60    PC61    PC62
## Standard deviation     0.79940 0.78355 0.76476 0.75328 0.73644 0.71816 0.70821
## Proportion of Variance 0.00003 0.00003 0.00002 0.00002 0.00002 0.00002 0.00002
## Cumulative Proportion  0.99951 0.99953 0.99956 0.99958 0.99960 0.99962 0.99964
##                           PC63    PC64    PC65    PC66    PC67    PC68    PC69
## Standard deviation     0.69601 0.67942 0.67135 0.65836 0.65074 0.63033 0.61980
## Proportion of Variance 0.00002 0.00002 0.00002 0.00002 0.00002 0.00002 0.00002
## Cumulative Proportion  0.99966 0.99968 0.99970 0.99972 0.99974 0.99975 0.99977
##                           PC70    PC71    PC72    PC73    PC74    PC75    PC76
## Standard deviation     0.61590 0.59762 0.58719 0.57085 0.55421 0.54061 0.53497
## Proportion of Variance 0.00002 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99978 0.99980 0.99981 0.99983 0.99984 0.99985 0.99986
##                           PC77    PC78    PC79    PC80    PC81    PC82    PC83
## Standard deviation     0.51562 0.49865 0.48505 0.46845 0.45437 0.44241 0.41840
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99987 0.99988 0.99989 0.99990 0.99991 0.99992 0.99993
##                           PC84    PC85    PC86    PC87    PC88    PC89   PC90
## Standard deviation     0.41216 0.40604 0.39515 0.37473 0.37263 0.35634 0.3403
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001 0.0000
## Cumulative Proportion  0.99993 0.99994 0.99995 0.99995 0.99996 0.99996 1.0000
##                           PC91   PC92   PC93   PC94   PC95   PC96   PC97   PC98
## Standard deviation     0.3251 0.3127 0.2987 0.2898 0.2853 0.2686 0.2657 0.2177
## Proportion of Variance 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
## Cumulative Proportion  1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
##                           PC99  PC100  PC101  PC102   PC103   PC104     PC105
## Standard deviation     0.2013 0.1861 0.1441 0.1154 0.08279 0.06699 5.406e-13
## Proportion of Variance 0.0000 0.0000 0.0000 0.0000 0.00000 0.00000 0.000e+00
## Cumulative Proportion  1.0000 1.0000 1.0000 1.0000 1.00000 1.00000 1.000e+00
```

```r
dat_ngram_pcs <-
  dat_ngram_i %>%
  select(c(type, Name, Stage, PM)) %>%
  bind_cols(fit_pca$x %>% data.frame() %>% select(str_c("PC", 1:.pc)))
```

```r
.cols <- str_c("PC", 1:.pc)


dat_ngram_MahaD <-
  dat_ngram_pcs %>%
  group_nest(Stage) %>%
  mutate(base = map(data, \(x){
    x %>%
      filter(type == "UE") %>%
      select(starts_with("PC"))
  })) %>%
  mutate(base = map(data, \(x){
    x %>% filter(type == "UE")
  })) %>%
  mutate(
    x = map(data, \(x){select(x, .cols)}),
    center = map(base, \(x){select(x, .cols) %>% summarise_all(mean) %>% unlist()}),
    cov = map(base, \(x){select(x, .cols) %>% var()})
  )%>%
  mutate(maha = pmap(list(x, center, cov), mahalanobis))
```

```r
.threshold <- qchisq(0.95, length(.cols))

dat_mahaD <-
  dat_ngram_MahaD %>%
  select(Stage, data, maha) %>%
  unnest(everything())

write_csv(dat_mahaD, "data/data_MahaD.csv")
```

```r
g_maha <-
  dat_mahaD %>%
  ggplot() +
  aes(PM, log10(sqrt(maha)), color = type) +
  geom_rect(xmin = 0, xmax = 6, ymin = log10(0), ymax = log10(sqrt(.threshold)),
            color = NA, fill = "lightgrey") +
  geom_vline(xintercept = c(2.375, 3.875), color = "white") +
  geom_path(aes(group = Name), alpha = 0.25) +
  geom_point() +
  scale_color_manual(values = c(UE = "black", VPA = "red")) +
  theme(legend.title = element_blank()) +
  labs(x = "PM", y = "log10(Mahalanobis D)")
```

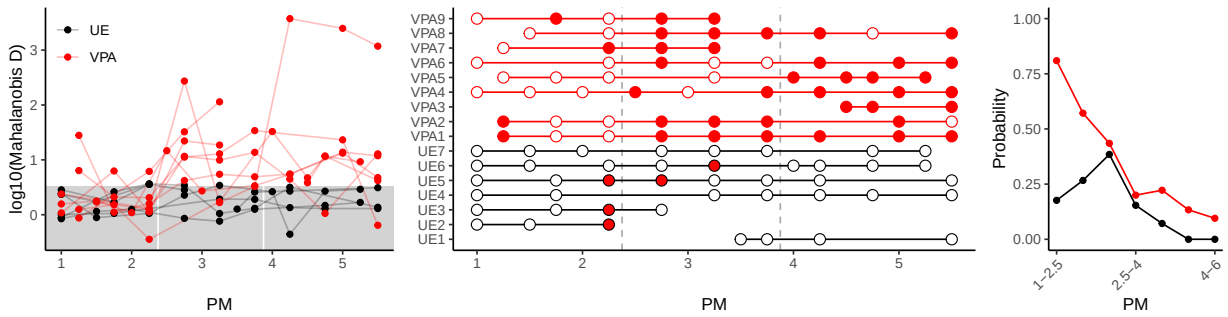
```r
dat_judge <-
  dat_mahaD %>%
  mutate(judge = if_else(maha <= .threshold, 1, 0))  %>%
  left_join(dat_id %>% select(Name, Pup), by = "Name") %>%
```

```
  mutate(y = factor(Pup) %>% as.numeric()) %>%
#  mutate(y = str_c(type, "_", Name),
#         y = factor(y) %>% as.numeric()) %>%
  group_by(type) %>%
#  mutate(ylab = str_c(type, "_", y - min(y) + 1)) %>%
  ungroup()


.labs <-
  dat_judge %>%
  select(y, Pup) %>%
  distinct() %>%
  arrange(y) %>%
  .$Pup

g_judge <-
  dat_judge %>%
  mutate(judge = if_else(judge == 0, "out", "in")) %>%
  ggplot() +
  aes(PM, y, color = type) +
  geom_vline(xintercept = c(2.375, 3.875), linetype = "dashed", color = "darkgrey") +
  geom_line(aes(group = Name))+
  geom_point(size = 3,aes(fill = judge), shape = 21) +
  scale_fill_manual(values = c("white", "red")) +
  scale_color_manual(values = c(UE = "black", VPA = "red")) +
  scale_y_continuous(breaks = 1:16, labels = .labs) +
  theme(axis.title.y = element_blank(),
        legend.title = element_blank())+
  labs(x = "PM")
```

```
dat_r <-
  dat_judge %>%
  filter(type == "VPA") %>%
  group_nest() %>%
  mutate(section = list(seq(1, 4, by = 0.5))) %>%
  unnest(section) %>%
  mutate(end = section + 1.5) %>%
  mutate(end = if_else(end > 5, 6, end)) %>%
  mutate(data = map2(data, section, ~.x %>% filter(PM >= .y))) %>%
  mutate(data = map2(data, end, ~.x %>% filter(PM < .y))) %>%
  mutate(n = map_dbl(data, nrow),
         FN = map_dbl(data, ~sum(.$judge))) %>%
  mutate(r = FN / n) %>%
  mutate(xlab = str_c(section, "-", end))

dat_r2 <-
  dat_judge %>%
  filter(type == "UE") %>%
  group_nest() %>%
  mutate(section = list(seq(1, 4, by = 0.5))) %>%
  unnest(section) %>%
  mutate(end = section + 1.5) %>%
  mutate(end = if_else(end > 5, 6, end)) %>%
```

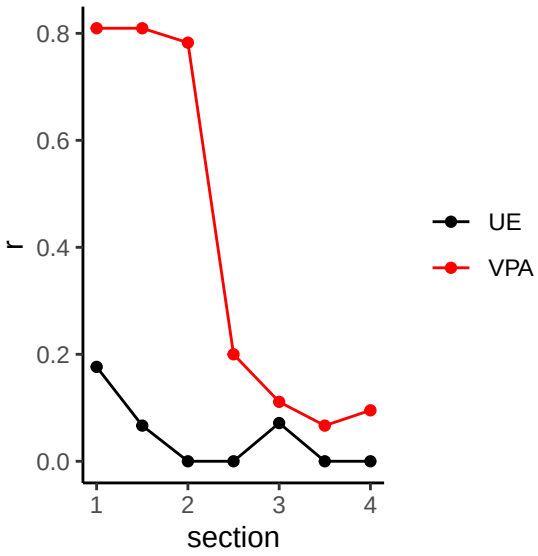
```
  mutate(data = map2(data, section, ~.x %>% filter(PM >= .y))) %>%
  mutate(data = map2(data, end, ~.x %>% filter(PM < .y))) %>%
  mutate(n = map_dbl(data, nrow),
         FN = map_dbl(data, ~sum(.$judge))) %>%
  mutate(r = 1 - FN / n) %>%
  mutate(xlab = str_c(section, "-", end))

g_FN <-
  dat_r %>%
  ggplot() +
  aes(section, r) +
  geom_path(color = "red") +
  geom_point(color = "red") +
  geom_path(data = dat_r2, color = "black") +
  geom_point(data = dat_r2, color = "black") +
  scale_y_continuous(limits = c(0, 1),
                     breaks = seq(0, 1, by = 0.25)) +
  scale_x_continuous(breaks = c(1, 2.5, 4),
                     labels = c(dat_r$xlab %>% .[c(1, 4, 7)])) +
  labs(x = "PM", y = "Probability") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
g <-
  patchwork::wrap_plots(
  g_maha +
    theme(legend.position = c(0, 1),
          legend.justification = c(0, 1),
          legend.background = element_rect(fill = NA)),
  g_judge +
    theme(legend.position = "none"),
  g_FN,
  widths = c(2, 3, 1)
)

g
```




```
ggsave("fig/gram4_pca_mahaD.png", g,
       width = 11, height = 4)

ggsave("fig/gram4_pca_mahaD.svg", g,
       width = 11, height = 3)
```

```r
.cols <- str_c("PC", 1:.pc)


dat_ngram_MahaD2 <-
  dat_ngram_pcs %>%
  group_nest() %>%
  mutate(section = list(seq(1, 4, by = 0.5))) %>%
  unnest(section) %>%
  mutate(end = section + 1.5) %>%
  mutate(end = if_else(end > 5, 6, end)) %>%
  mutate(data = map2(data, section, ~.x %>% filter(PM >= .y))) %>%
  mutate(data = map2(data, end, ~.x %>% filter(PM < .y))) %>%
  mutate(base = map(data, \(x){
    x %>%
      filter(type == "UE") %>%
      select(starts_with("PC"))
  })) %>%
  mutate(base = map(data, \(x){
    x %>% filter(type == "UE")
  })) %>%
  mutate(
    x = map(data, \(x){select(x, .cols)}),
    center = map(base, \(x){select(x, .cols) %>% summarise_all(mean) %>% unlist()}),
    cov = map(base, \(x){select(x, .cols) %>% var()})
  )%>%
  mutate(maha = pmap(list(x, center, cov), mahalanobis))


.threshold <- qchisq(0.95, length(.cols))

dat_judge2 <-
  dat_ngram_MahaD2 %>%
  select(section, end, data, maha) %>%
  unnest(everything()) %>%
  mutate(judge = if_else(maha <= .threshold, 1, 0)) %>%
  group_nest(section, end, type) %>%
  mutate(IN = map_dbl(data, ~sum(.$judge)),
         n = map_dbl(data, nrow),
         r = IN / n,
         r = if_else(type == "UE", 1 - r, r))

dat_judge2 %>%
  ggplot() +
  aes(section, r, color = type) +
  geom_path() +
  geom_point() +
  scale_color_manual(values = c(UE = "black", VPA = "red"))
```


0.8
0.6
0.4
0.2
0.0
r
1
2
3
4
section
UE
VPA