LDP_LD

Yawen Yu and Dan Yurovsky

12/5/2017

- Library

```
library(koRpus)
library(tidyverse)
library(tm)
library(lme4)
library(dplyr)
library(feather)
```

- Load and Clean Data

```
# Read in LDP data
ldp <- src_sqlite("/Users/Yawen/Desktop/application/lexical diversity/trial5_ldp/ldp.db")

# Get all utterances
utter <- tbl(ldp, "utterances") %>%
  collect()

# Get all participants
subjs <- tbl(ldp, "subjects") %>%
  collect() %>%
  rename(subject = id)

# Get visit data
visits <- tbl(ldp, "visits") %>%
  collect() %>%
  select(subject, session, date, child_age, child_age_years, child_age_months,
         income)

# Get measures data
measures <- tbl(ldp, "measures") %>%
  collect() %>%
  select(-last_update) %>%
  left_join(y=visits, by = c("subject", "session")) %>%
  mutate(ttr = word_types/word_tokens)

# remove unintelligible utterances
murmur = c("xxx", "yyy", "yyy_yyy","---")
utter_clean <- utter %>%
  filter(!c_utts %in% murmur) %>%
  mutate(c_utts = removeWords(c_utts, murmur),
         c_utts = gsub("[^[:alnum:]]", " ", c_utts)) %>%
  filter(!grepl("^\\s*$", c_utts))


#### Notice: Age column is NOT completely coded, but Session column is complete
# complete missing Age data according to Session data
session_80 <- visits %>%
  filter(session > 12) %>%
```

```
  mutate(age = round(mean(child_age_months))) %>%
  select("session", "age") %>%
  unique(.)

session_age <- measures %>%
  select("session", "child_age_months") %>%
  mutate(child_age_months = round(child_age_months)) %>%
  rename(age = child_age_months) %>%
  unique(.) %>%
  head(n=12) %>%
  rbind(session_80)

# collapse utterances at one month to one row
utter_collapsed <- utter_clean %>%
  filter(subject == subject) %>%
  filter(session == session) %>%
  group_by(subject, session) %>%
  summarise(utts = paste0(c_utts, collapse = " "))

# keep only complete rows
utter_tokenize <- utter_collapsed[complete.cases(utter_collapsed),] %>%
                        split(paste0(.$subject, "-", .$session, "-"))
```

- Measure Lexical Diversity

```
##### TTR is not very informative
measures %>%
  filter(speaker == "C") %>%
  ggplot(aes(x=session, y=ttr)) +
  geom_smooth(se = F) +
  theme_classic()

# measure LD with MTLD
tokenize_mtld <- function(df) {
  tokenized_df <- koRpus::tokenize(df$utts, format = "obj", lang = "en", tag = TRUE) #tokenize texts
  MTLD <- MTLD(tokenized_df) # measure LD
  mtld <- data_frame(mtld = MTLD@MTLD$MTLD) # get LD measurement
  length <- data_frame(length = MTLD@tt$num.tokens)  # get length meaasurement
  merge(x = length,y = mtld, by = NULL) # combine LD and lenghth values
}

# measure and filter accurate data
utter_ld <- map(utter_tokenize, tokenize_mtld)%>%
  bind_rows(.id = "id") %>%
  separate(col = id, into = c("subject", "session"), sep = "-") %>%
  mutate(subject = as.integer(subject),
         session = as.integer(session)) %>%
  filter(!mtld == as.numeric("inf")) %>%
  filter(!length < 100) # MTLD yield inaccurate results when utterance length <100 words

# filter kids' data with more than 2 observeations
mtld_filter <- function(df) {
  nmtld <- aggregate(df$mtld,
  by = list(subject = df$subject),length)
```

```
   nchild <- nmtld$subject[(nmtld$x > 2)]

  df <- df %>%
  filter(subject %in% nchild) %>%
  group_by(subject)

  return(df)
}

filter_data <- mtld_filter(utter_ld)


# Measure Mother's Lexical Diversity
# remove unintelligible utterances
utter_mom <- utter %>%
  filter(!p_utts %in% murmur) %>%
  mutate(p_utts = removeWords(p_utts, murmur),
         p_utts = gsub("[^[:alnum:]]", " ", p_utts)) %>%
  filter(!grepl("^\\s*$", p_utts)) %>%
  select(subject, session, p_utts)%>%
  left_join(x=session_age, by = c("session"))

# collapse each month's utterances to one row
mom_col <- utter_mom %>%
  group_by(subject, session, age) %>%
  summarise(utts = paste0(p_utts, collapse = " "))

# keep only complete rows
mom_tok <- mom_col[complete.cases(mom_col),] %>%
  split(paste0(.$subject, "-", .$session, "-", .$age))

mom_ld <- map(mom_tok, tokenize_mtld)%>%
  bind_rows(.id = "id") %>%
  separate(col = id, into = c("subject", "session"), sep = "-") %>%
  mutate(subject = as.integer(subject),
         session = as.integer(session)) %>%
  filter(!mtld == as.numeric("inf")) %>%
  filter(!length < 100)

mom_data <- mtld_filter(mom_ld)


# merge data
complete_data <- filter_data %>%
  left_join(y = mom_data,by = c("subject", "session")) %>%
  rename(kid_mtld = mtld.x,
         kid_length = length.x,
         mom_mtld = mtld.y,
         mom_length = length.y) %>%
  left_join(y = session_age, by = c("session")) %>%
  left_join(y = subjs, by = c("subject")) %>%
  filter(lesion == "") %>% ## keep data of typically-developing children
  left_join(visits) %>%
```

```
  group_by(subject) %>%
  mutate(income = replace(income, is.na(income), round(mean(income, na.rm = T))))  %>%
  ungroup(.) %>%
  filter(complete.cases(.)) %>%
  select(subject, session, age, kid_length, kid_mtld, mom_length, mom_mtld, sex, race, ethn, income)


# 66 kids in total
id <- unique(complete_data$subject)

# write data to disk
write_feather(filter_data,
              "/Users/Yawen/Desktop/lexical diversity/trial5_ldp/filter_data.feather")
write_feather(mom_data,
              "/Users/Yawen/Desktop/lexical diversity/trial5_ldp/mom_data.feather")
write_feather(complete_data,
              "/Users/Yawen/Desktop/lexical diversity/trial5_ldp/complete_data.feather")
```

Model 1: Growth Trajectory

```
filter_data <-
  read_feather("/Users/Yawen/Desktop/lexical diversity/trial5_ldp/filter_data.feather")
mom_data <-
  read_feather("/Users/Yawen/Desktop/lexical diversity/trial5_ldp/mom_data.feather")
complete_data <-
  read_feather("/Users/Yawen/Desktop/lexical diversity/trial5_ldp/complete_data.feather")


# Model1: child's lexical diversity of each session as response
age_model <- lmer(kid_mtld ~ 1 + log(age) +
                    (1+log(age)|subject), data = complete_data)

Model1 <-age_model
summary(Model1)
```

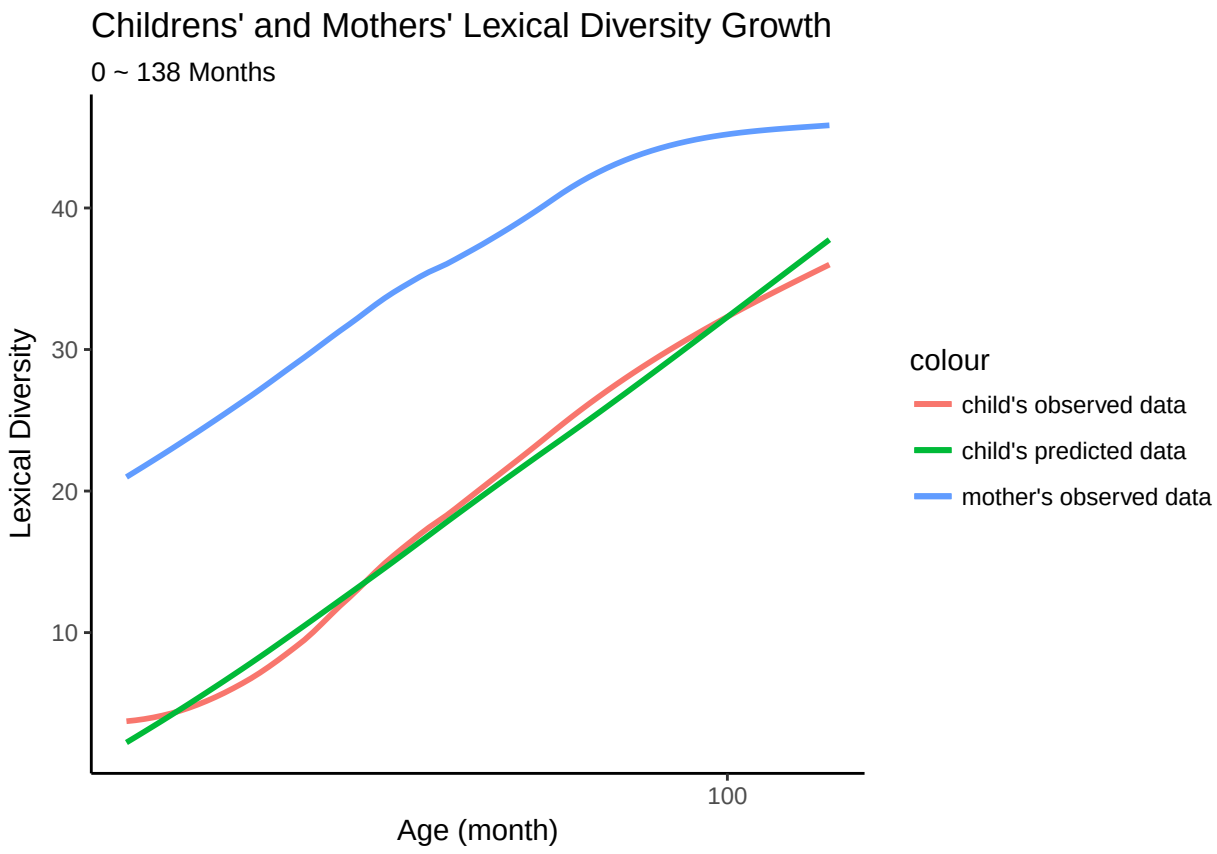
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: kid_mtld ~ 1 + log(age) + (1 + log(age) | subject)
##    Data: complete_data
##
## REML criterion at convergence: 4078.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2306 -0.5882 -0.0656  0.5362  4.3106
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  subject  (Intercept) 52.715   7.260
##           log(age)     4.719   2.172    -0.94
##  Residual             13.770   3.711
## Number of obs: 718, groups:  subject, 66
##
```

```
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) -43.9138     1.4059  -31.24
## log(age)     16.5943     0.3993   41.56
##
## Correlation of Fixed Effects:
##          (Intr)
## log(age) -0.969
```

```
# get kid's slope based on Model1
fixed_effects <- fixef(Model1)
ran_effects <- as_data_frame(ranef(Model1)$subject)
kid_effects <-  ran_effects %>%
  mutate(subject = as.integer(rownames(ran_effects)),
         kid_slope = `log(age)` + fixed_effects[2]) %>%
  select(subject, kid_slope)


# plot predicted lexical diversity growth
predict_data <- complete_data %>%
  mutate(predicted = predict(Model1, newdata =.))

ggplot()+
  geom_smooth(aes(x = age, y = mom_mtld,
                  color = "mother's observed data"),se = FALSE, data = predict_data)+
  geom_smooth(aes(x=age, y=kid_mtld,
                  color = "child's observed data"),se = FALSE, data = predict_data)+
  geom_smooth(aes(x=age, y=predicted,
                  color = "child's predicted data"),se = FALSE, data = predict_data)+
  labs(title = "Childrens' and Mothers' Lexical Diversity Growth",
       subtitle = "0 ~ 138 Months") +
  theme_classic() +
  ylab("Lexical Diversity")+
  xlab("Age (month)") +
  scale_x_log10()
```



Get Estimated Intercept

```
# Child's predicted mtld value at 20 month as intercept
kid_data <- kid_effects %>%
  mutate(age = 20)%>%
  mutate(kid_intercept = predict(Model1, newdata = .)) %>%
  select(-c(age))

# predict mother's intercept at 20 month
mom_slope_model <- lmer(mom_mtld ~  1+ log(age)
                        + (1 + log(age)|subject), data = complete_data)

mom_fix <- fixef(mom_slope_model)
mom_random <- as_data_frame(ranef(mom_slope_model)$subject)
mom_effects <-  mom_random %>%
  mutate(subject = as.integer(rownames(mom_random)),
         mom_slope = `log(age)` + mom_fix[2]) %>%
  select(subject, mom_slope) %>%
  mutate(age = 20) %>%
  mutate(mom_intercept = predict(mom_slope_model, newdata = .))  %>%
  select(-c(age))


# combine data
all_data <- kid_data %>%
```

```
  left_join(mom_effects) %>%
  left_join(complete_data) %>%
  group_by(subject, kid_slope, kid_intercept,
           mom_slope, mom_intercept, race, sex, ethn) %>%
  summarise(income = mean(income)) %>%
  ungroup() %>%
  mutate(race = factor(race, levels = c("WH", "BL", "2+"))) %>%
  mutate(sex = factor(sex, levels = c("M", "F")))

month_data <- complete_data %>%
  left_join(select(all_data, subject, mom_slope, mom_intercept))  %>%
  ungroup() %>%
  mutate(race = factor(race, levels = c("WH", "BL", "2+")))%>%
  mutate(sex = factor(sex, levels = c("M", "F")))


# residual as response
age_model <- lm(kid_mtld ~ scale(log(age)), data = month_data)

resid_resid <- month_data %>%
  mutate(residual = residuals(age_model)) %>%
  lm(residual ~ mom_slope + mom_intercept + income + sex + age +ethn, data = .) %>%
  residuals(.)

# Mixed effect model with kid's monthly mtld measurement as response
full_model <- lmer(scale(kid_mtld) ~ scale(log(age)) +
                     scale(mom_slope) + scale(mom_intercept) +
                     (income > median(income)) +
                     (scale(log(age)) | subject), data = month_data)

month_data %>%
  mutate(residual = residuals(age_model)) %>%
  lm(residual ~ mom_slope + mom_intercept + income, data = .) %>%
  summary()
```

```
##
## Call:
## lm(formula = residual ~ mom_slope + mom_intercept + income, data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.1296  -2.8166  -0.2247   2.4436  24.0649
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -20.00690    1.93495 -10.340  < 2e-16 ***
## mom_slope       0.83359    0.10107   8.248 7.76e-16 ***
## mom_intercept   0.33015    0.03594   9.185  < 2e-16 ***
## income          0.33606    0.09350   3.594 0.000348 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.34 on 714 degrees of freedom
## Multiple R-squared:  0.1389, Adjusted R-squared:  0.1353
```

```
## F-statistic: 38.38 on 3 and 714 DF,  p-value: < 2.2e-16
```

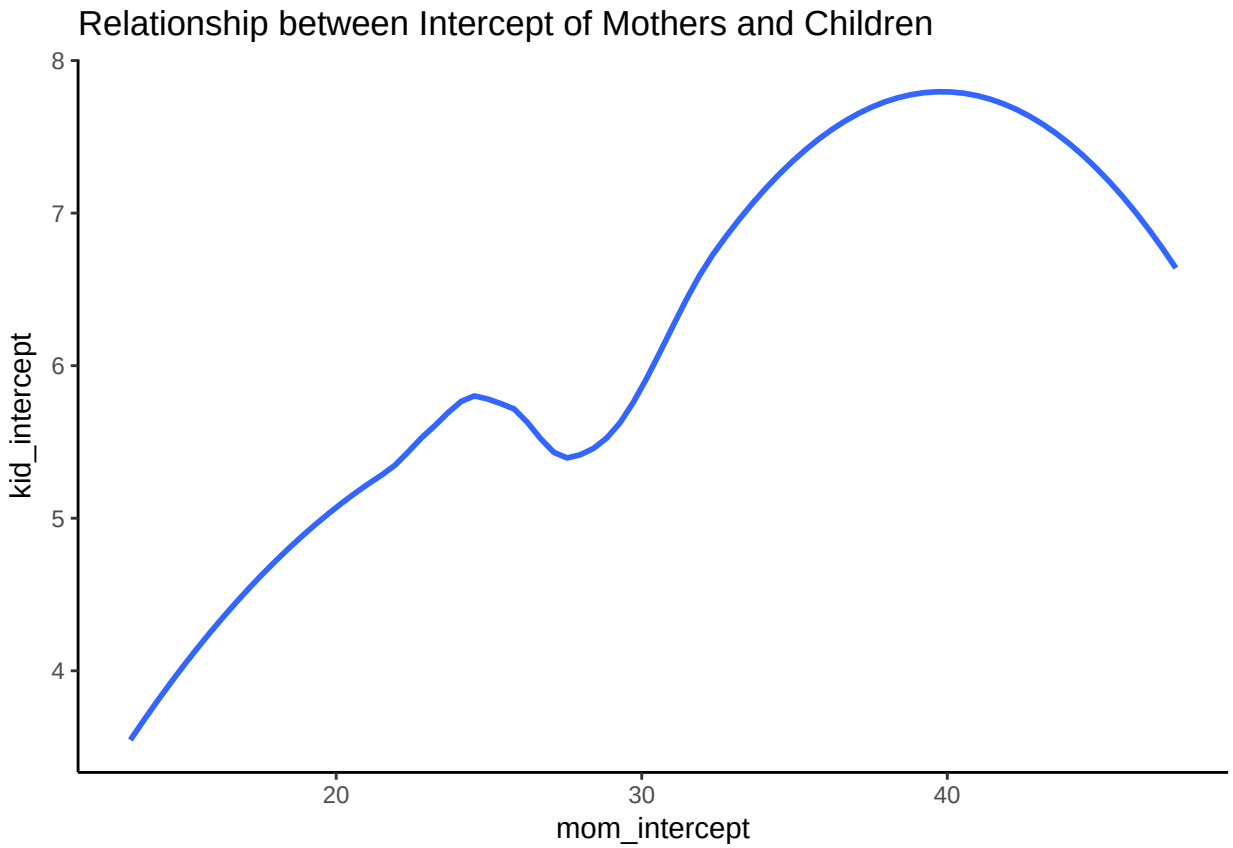
Model2: Child Intercept as Repsonse

```
null <- lm(kid_intercept ~ 1 , data = all_data)
full <- lm(kid_intercept ~ mom_slope + mom_intercept +
             sex + race + income, data = all_data)
stats::step(null, scope=list(lower=null, upper=full), direction="both", na.rm = TRUE)
```

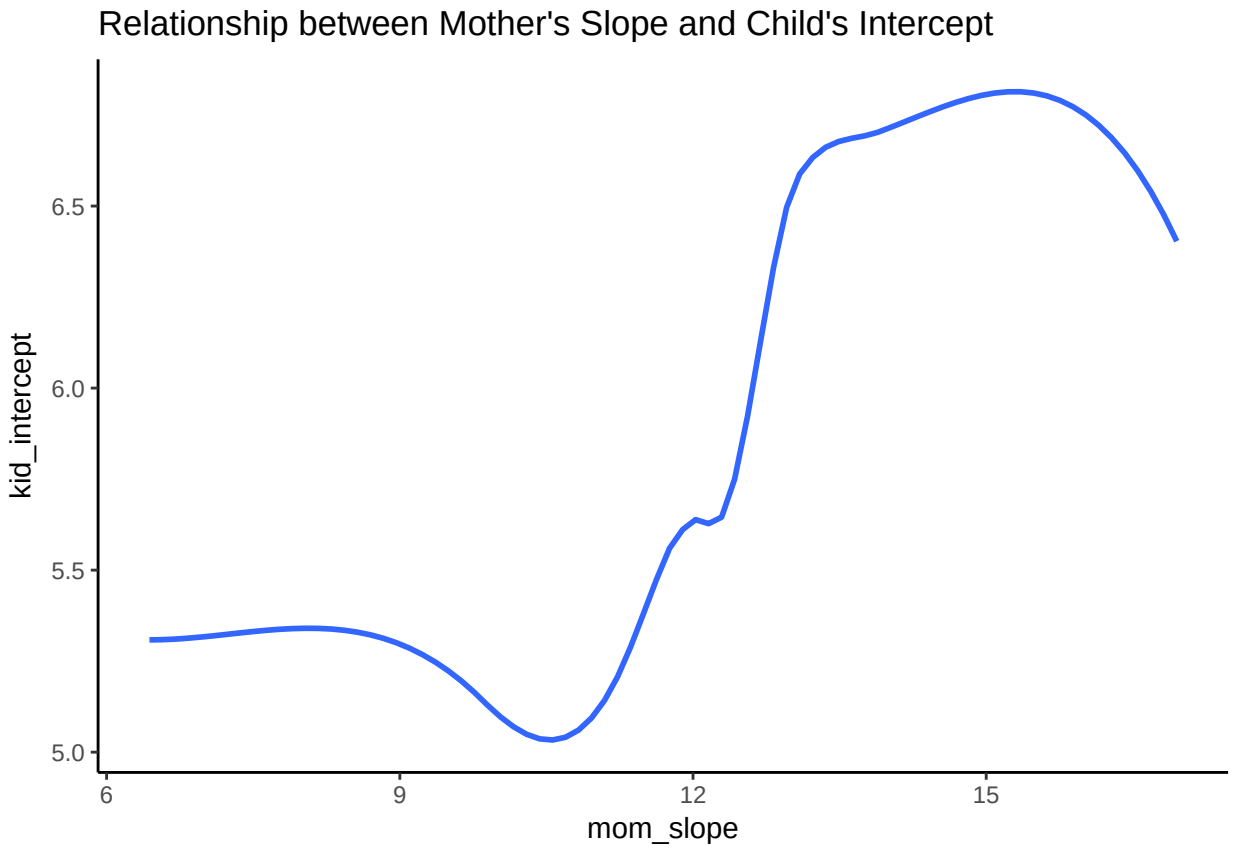
```
## summarize model
Model2 <- lm(kid_intercept ~ mom_slope + mom_intercept +
               scale(income),  data = all_data)
summary(Model2)
```

```
##
## Call:
## lm(formula = kid_intercept ~ mom_slope + mom_intercept + scale(income),
##     data = all_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8032 -0.8567  0.1035  1.1277  3.8326
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.06858    2.44720  -2.888  0.00533 **
## mom_slope      0.59955    0.13241   4.528 2.76e-05 ***
## mom_intercept  0.21533    0.04617   4.663 1.70e-05 ***
## scale(income)  0.34117    0.21961   1.554  0.12538
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.717 on 62 degrees of freedom
## Multiple R-squared:  0.3356, Adjusted R-squared:  0.3035
## F-statistic: 10.44 on 3 and 62 DF,  p-value: 1.186e-05
```

```
# Plot
ggplot(all_data, aes(x = mom_intercept, y = kid_intercept))+
  geom_smooth(se=F)+
  theme_classic()+
  labs(title = "Relationship between Intercept of Mothers and Children")
```




```
ggplot(all_data, aes(x = mom_slope, y = kid_intercept))+
  geom_smooth(se=F)+
  theme_classic()+
  labs(title = "Relationship between Mother's Slope and Child's Intercept")
```



Model3: Child's Slope as Response

```
null2 <- lm(kid_slope ~ 1, data = all_data)
full2 <- lm(kid_slope ~ kid_intercept + mom_slope +
              mom_intercept + sex + race + income, data = all_data)
stats::step(null2, scope=list(lower=null2, upper=full2), direction="both", na.rm = TRUE)
```

```
# summarize Model3
Model3 <- lm(kid_slope ~ mom_intercept, data = all_data)
summary(Model3)
```

```
##
## Call:
## lm(formula = kid_slope ~ mom_intercept, data = all_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4434 -1.1650  0.0075  0.9803  4.4668
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   14.94999    0.91655  16.311   <2e-16 ***
## mom_intercept  0.06187    0.03379   1.831   0.0717 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 1.492 on 64 degrees of freedom
## Multiple R-squared:  0.04978,    Adjusted R-squared:  0.03494
## F-statistic: 3.353 on 1 and 64 DF,  p-value: 0.07174
```

Model 4: Mother's intercept as Response

```
# Mothers talk differently to boys and girls
null3 <- lm(mom_intercept ~ 1 , data = all_data)
full3 <- lm(mom_intercept ~ kid_slope + kid_intercept +
              sex + race + income, data = all_data)
stats::step(null3, scope=list(lower=null3, upper=full3), direction="both", na.rm = TRUE)
```

```
# summarize Model 4
Model4 <- lm(formula = mom_intercept ~ kid_intercept + sex + race + kid_slope +
    income, data = all_data)
summary(Model4)
```

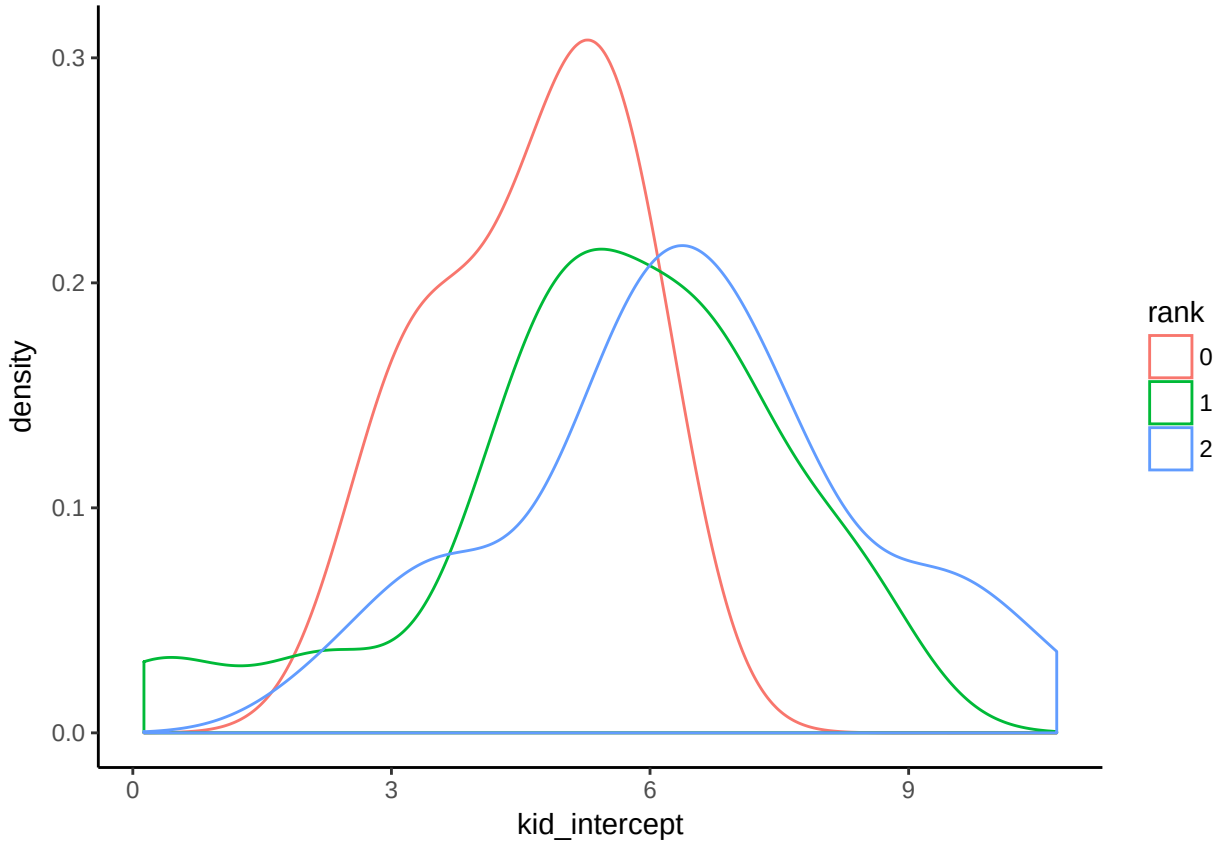
```
##
## Call:
## lm(formula = mom_intercept ~ kid_intercept + sex + race + kid_slope +
##     income, data = all_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.3774  -3.1657   0.1141   3.0533  16.3058
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    15.9420     6.8205   2.337   0.0228 *
## kid_intercept   0.4983     0.3154   1.580   0.1195
## sexF            3.4319     1.3070   2.626   0.0110 *
## raceBL         -3.3926     1.5900  -2.134   0.0370 *
## race2+         -3.1685     2.2265  -1.423   0.1600
## kid_slope       0.3269     0.4214   0.776   0.4410
## income          0.4489     0.3685   1.218   0.2279
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.869 on 59 degrees of freedom
## Multiple R-squared:  0.2828, Adjusted R-squared:  0.2099
## F-statistic: 3.878 on 6 and 59 DF,  p-value: 0.002496
```

Effect of Maternal Speech

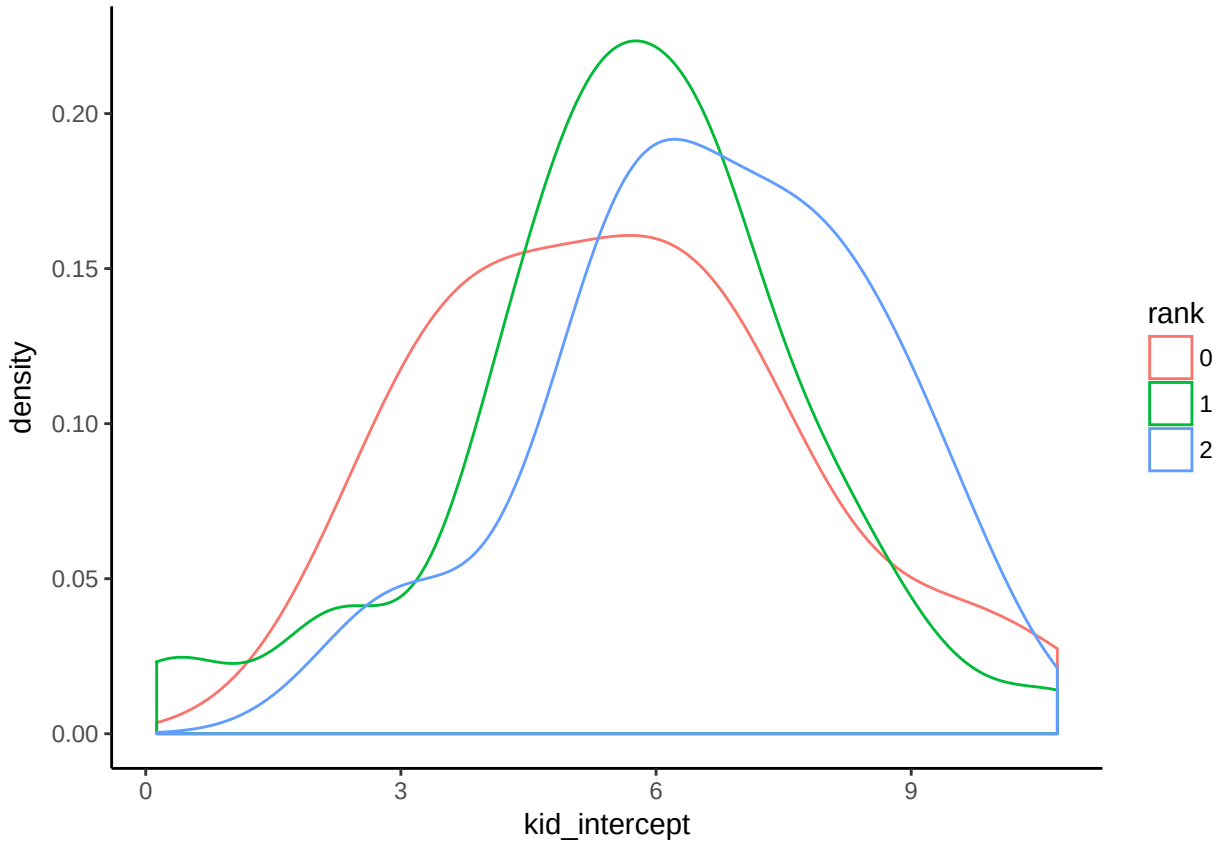
```
# How to define mother's speech as "more/less diverse"?
# Also, what effect does mother's DIVERSITY have on kid's INTERCEPT?

# 1) 1sd ± mean of mothers' intercept
## high:28 moms; median:32; low: 6;
all_data %>%
  ungroup(.) %>%
  mutate(mean = mean(mom_intercept),
```

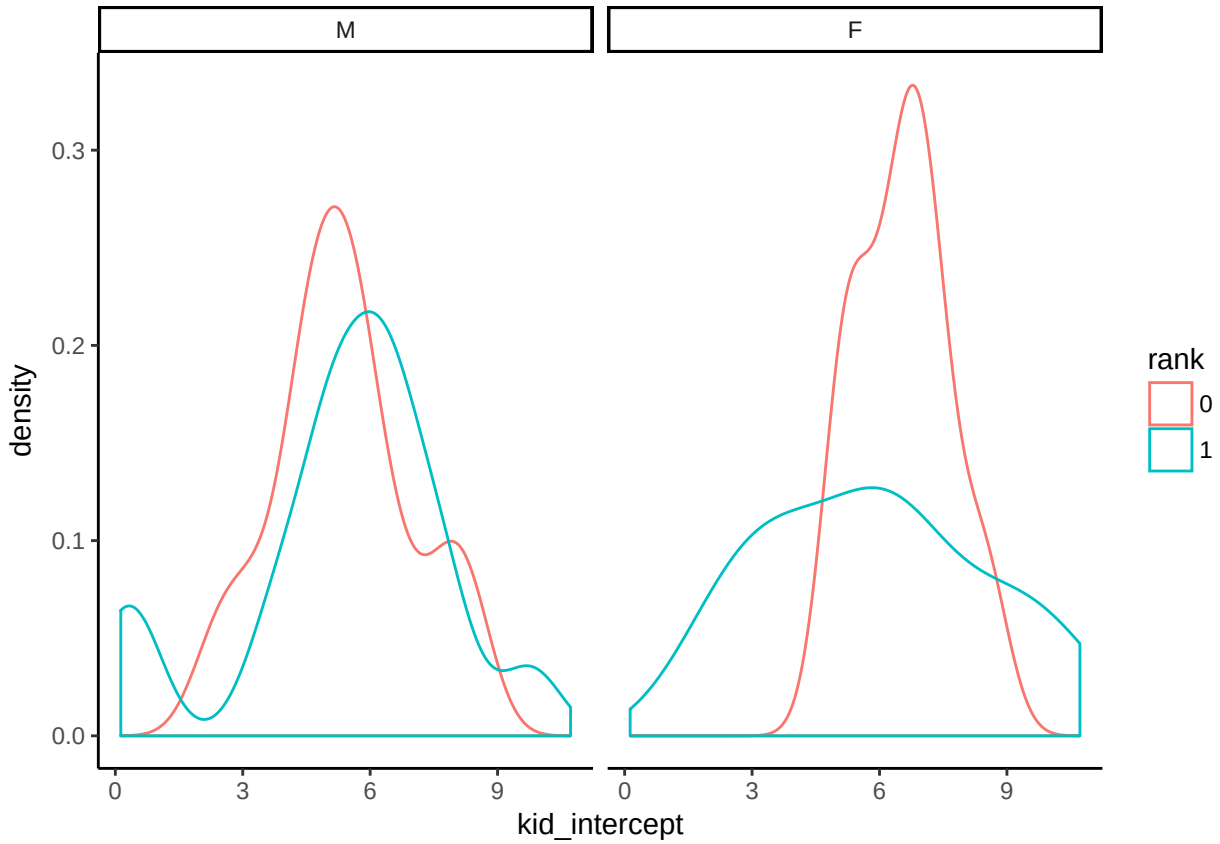
```
        sd = sd(mom_intercept)) %>%
  mutate(rank = ifelse(mom_intercept < mean-sd, 0,
                       ifelse( mom_intercept < mean, 1,2)),
        rank = as.factor(rank)) %>%
  ggplot(.,aes(x=kid_intercept, color=rank))+
  geom_density()+
  theme_classic()
```



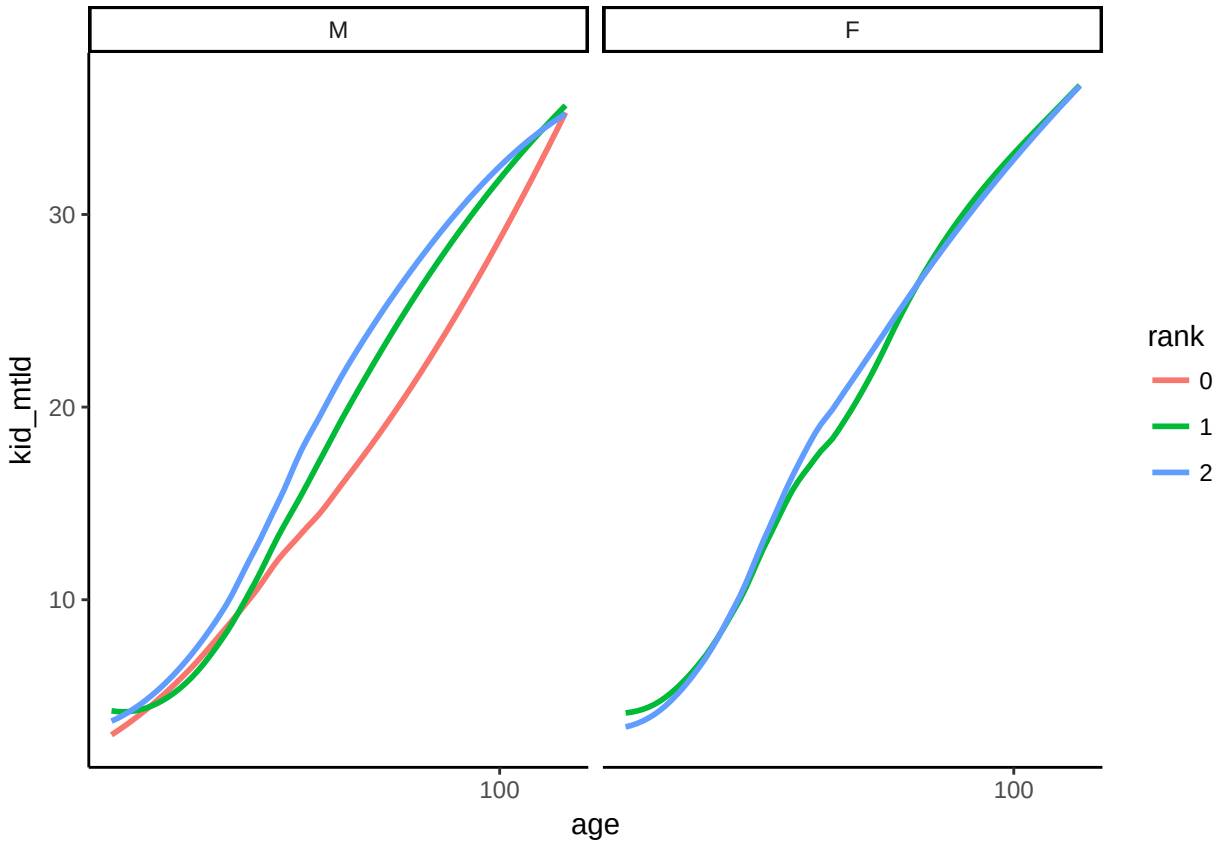

```
# 2) look at slope: to what degree mother change the way they talk
## high:10 moms; median:47; low: 9
all_data %>%
  ungroup()%>%
  mutate(mean = mean(mom_slope),
        sd = sd(mom_slope)) %>%
  mutate(rank = ifelse(mom_slope <= mean-sd, 0,
                       ifelse(mean+sd > mom_slope, 1,2)),
        rank = as.factor(rank)) %>%
  ggplot(.,aes(x=kid_intercept, color=rank))+
  geom_density()+
  theme_classic()
```




```
# 3) difference between mom's intercept and kid's intercept
## high:32 moms; lower: 34 moms
all_data %>%
  ungroup()%>%
  mutate(dif = mom_intercept - kid_intercept) %>%
  mutate(mean = mean(dif),
         rank = ifelse(dif > mean, 1, 0),
         rank = as.factor(rank)) %>%
  ggplot(.,aes(x=kid_intercept, color=rank))+
  geom_density()+
  facet_wrap(~sex)+
  theme_classic()
```




```
# General growth trajectory varies by mother's diversity of speech
# also show dif effects on boys and girls
month_data %>%
  ungroup()%>%
 mutate(mean = mean(mom_intercept),
        sd = sd(mom_intercept)) %>%
  mutate(rank = ifelse(mom_intercept < mean-sd, 0,
                       ifelse(mom_intercept < mean, 1,2)),
         rank = as.factor(rank)) %>%
  group_by(subject) %>%
  ggplot(.,aes(x=age, y=kid_mtld, color=rank))+
  geom_smooth(se=F)+
  theme_classic() +
  scale_x_log10()+
  facet_wrap(~sex)
```


M
F
30
20
10
kid_mtld
100
100
age
rank
0
1
2