Validation results

Meg Cychosz

06 January 2021

```
# get total # of clips from each recording
complete2 <- complete %>%
  group_by(id) %>%
  distinct(file_name, .keep_all = T) %>%
  mutate(num_clips = NROW(Media)*2)

clips <- complete2 %>%
  select(id, num_clips) %>%
  distinct(id, .keep_all = T)

data <- merge(clips, random, by='id')
data2 <- rbind(data, complete2)
```

```
data3 <- data2 %>%
  group_by(method, id) %>%
  mutate(num_clips_drawn = (NROW(file_name))) %>%
  mutate(percen_ofallclips_drawn=(NROW(file_name)/num_clips)*100) # sanity check - complete method shou
```

```
data_annon <- data3 %>%
 gather("addressee", "language", Adult2OtherChild, Adult2Others, Adult2TargetChild, Adult2unsure, Other
  filter(language=='Mixed' | language=='Spanish' | language=='English/Quechua' | language =='Unsure') %
  group_by(id, method) %>%
  distinct_at(., vars(file_name, language), .keep_all = T) %>% # don't record multiple speakers speakin
  mutate(total_annotations = NROW(file_name)) # N of annotations made; distinct from N of speech clips

# separately, calculate the num and % of annotated clips
data_annon_cts <- data_annon %>%
  group_by(id, method) %>%
  distinct(file_name, .keep_all = T) %>%
  mutate(speech_clips = NROW(file_name)) %>% # N of unique clips annotated - NOT the # of annotations
  mutate(percen_ofallclips_annon=(NROW(file_name)/num_clips)*100) %>% # % of total clips annotated
  select(speech_clips, percen_ofallclips_annon, id, method, file_name, num_clips_drawn, percen_ofallcli


for_speech_clips <- data_annon_cts %>%
  select(id, method, speech_clips) %>%
  distinct_at(., vars(id, method), .keep_all = T)

# first load in the complete files so we can estimate the # of available clips
all <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Meg_data/1032_config.csv')
all2 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Meg_data/1060_config.csv')
all3 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Meg_data/1075_config.csv')
```

```
all4 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Meg_data/1077_config.csv')
all5 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Meg_data/1081_config.csv')
all6 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Anele_data/179config.csv')
all7 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Anele_data/198config.csv')
all8 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Anele_data/199config.csv')
all9 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Anele_data/261config.csv')
all10 <- read.csv('/Users/megcychosz/Google Drive/biling_CDS/Anele_data/267config.csv')
config_files <- rbind(all, all2, all3, all4, all5, all6, all7, all8, all9, all10)

# calculate the num and % of all clips available for annotation
data_avbl <- config_files %>%
  group_by(id) %>%
  distinct(file_name, .keep_all = T) %>% # duplicates bc drawn with replacement
  mutate(voc = if_else(percents_voc > 0, "1", "0")) %>% # turn percents_voc binary
  filter(sleeping=='1' | researcher_present == '1' | voc == '0') %>%
  count() %>%
  rename(not_avl_clips = n) %>%
  merge(., data_annon, by=c('id')) %>%
  mutate(avbl_clips = num_clips - not_avl_clips) %>% # clips that were *available* for annotation out o
  merge(., for_speech_clips, by=c('id', 'method')) %>% # N of unique clips annotated - NOT the # of ann
  mutate(percen_avl_annon = (speech_clips / avbl_clips)*100) %>% # the # of clips annotated / # of avbl
  distinct_at(., vars(id, method), .keep_all = T) %>%
  group_by(method) %>%
  mutate(avbl_clips = paste(speech_clips, "(",round(percen_avl_annon,2),"%)")) %>%
  ungroup()%>%
  select(avbl_clips, id, method) %>%
  pivot_wider(names_from=method, values_from=c("avbl_clips"))
```

```
percen_tbl <- data_annon_cts %>%
  select(-file_name) %>%
  distinct_at(., vars(id,method), .keep_all = T) %>%
  mutate(clips_drawn = paste(num_clips_drawn,"(",round(percen_ofallclips_drawn,2),"%)")) %>%
  mutate(clips_annon = paste(speech_clips,"(",round(percen_ofallclips_annon,2),"%)")) %>%
  select(-num_clips_drawn, -percen_ofallclips_annon, -speech_clips, -percen_ofallclips_drawn) %>%
  relocate(c(id, method, clips_drawn, clips_annon)) %>%
  pivot_wider(names_from=method, values_from=c("clips_drawn", "clips_annon")) %>%
  merge(., data_avbl, by=c('id'))

percen_tbl$id <- plyr::mapvalues(percen_tbl$id,
                                 from=c('267-12mo', '261-8mo', '199', '198-9mo', '179', '1081', '1077',
           to=c('Spanish-English (267)', 'Spanish-English (261)','Spanish-English (199)',
  'Spanish-English (198)', 'Spanish-English (179)', 'Quechua-Spanish (1081)', 'Quechua-Spanish (1077)',

# actually decided to split this table and move part to the appendix
clip_annon_tbl <- percen_tbl %>%
  select(id, clips_annon_random, clips_annon_complete) %>%
  arrange(desc(id))

knitr::kable(clip_annon_tbl, caption = 'Number of clips annotated by child and annotation method.',
             booktabs=T,
             row.names = FALSE,
             col.names = c("Corpus (ID)", "Random", "Complete")) %>% # "
  kable_styling() %>%
```

```
  add_header_above(c(" " = 1, "# of clips annotated (% of total clips)" = 2)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

\begin{table}[!h]
\caption{(#tab:% drawn and annotated table)Number of clips annotated by child and annotation method.}

	# of clips annotated (% of total clips)	
Corpus (ID)	Random	Complete
Spanish-English (267)	101 (5.26 %)	274 (14.27 %)
Spanish-English (261)	92 (4.79 %)	294 (15.31 %)
Spanish-English (199)	118 (6.15 %)	467 (24.32 %)
Spanish-English (198)	81 (4.22 %)	302 (15.73 %)
Spanish-English (179)	120 (6.25 %)	633 (32.97 %)
Quechua-Spanish (1081)	92 (7.5 %)	285 (23.25 %)
Quechua-Spanish (1077)	83 (7.23 %)	355 (30.92 %)
Quechua-Spanish (1075)	81 (8.69 %)	199 (21.35 %)
Quechua-Spanish (1060)	111 (10.51 %)	405 (38.35 %)
Quechua-Spanish (1032)	97 (5.05 %)	372 (19.38 %)

\end{table}

```
clip_drawn_avbl_tbl <- percen_tbl %>%
  select(-clips_annon_random, -clips_annon_complete) %>%
  relocate(id, clips_drawn_random, clips_drawn_complete, random, complete) %>%
  arrange(desc(id))

knitr::kable(clip_drawn_avbl_tbl, caption = 'Number of clips drawn and number of clips annotated, by ch
             booktabs=T,
             row.names = FALSE,
             col.names = c("Corpus (ID)", "Random", "Complete", "Random", "Complete")) %>% # "
  kable_styling() %>%
  add_header_above(c(" " = 1, "# of clips drawn (% of total clips)" = 2, "# of clips annotated (% of ava
  kableExtra::kable_styling(latex_options = "hold_position")
```

\begin{table}[!h]
\caption{(#tab:% drawn and annotated table)Number of clips drawn and number of clips annotated, by child and annotation method.}

	# of clips drawn (% of total clips)		# of clips annotated (% of available clips)	
Corpus (ID)	Random	Complete	Random	Complete
Spanish-English (267)	345 (17.97 %)	960 (50 %)	101 (13.1 %)	274 (35.54 %)
Spanish-English (261)	290 (15.1 %)	960 (50 %)	92 (8.18 %)	294 (26.13 %)
Spanish-English (199)	192 (10 %)	960 (50 %)	118 (10.61 %)	467 (42 %)
Spanish-English (198)	284 (14.79 %)	960 (50 %)	81 (7.96 %)	302 (29.67 %)
Spanish-English (179)	192 (10 %)	960 (50 %)	120 (8.05 %)	633 (42.48 %)
Quechua-Spanish (1081)	249 (20.31 %)	613 (50 %)	92 (13.83 %)	285 (42.86 %)
Quechua-Spanish (1077)	137 (11.93 %)	574 (50 %)	83 (8.15 %)	355 (34.84 %)
Quechua-Spanish (1075)	267 (28.65 %)	466 (50 %)	81 (14.21 %)	199 (34.91 %)
Quechua-Spanish (1060)	154 (14.58 %)	528 (50 %)	111 (12.01 %)	405 (43.83 %)
Quechua-Spanish (1032)	263 (13.7 %)	960 (50 %)	97 (10.16 %)	372 (38.95 %)

\end{table}

0.0.1 Language categories across random and full methods

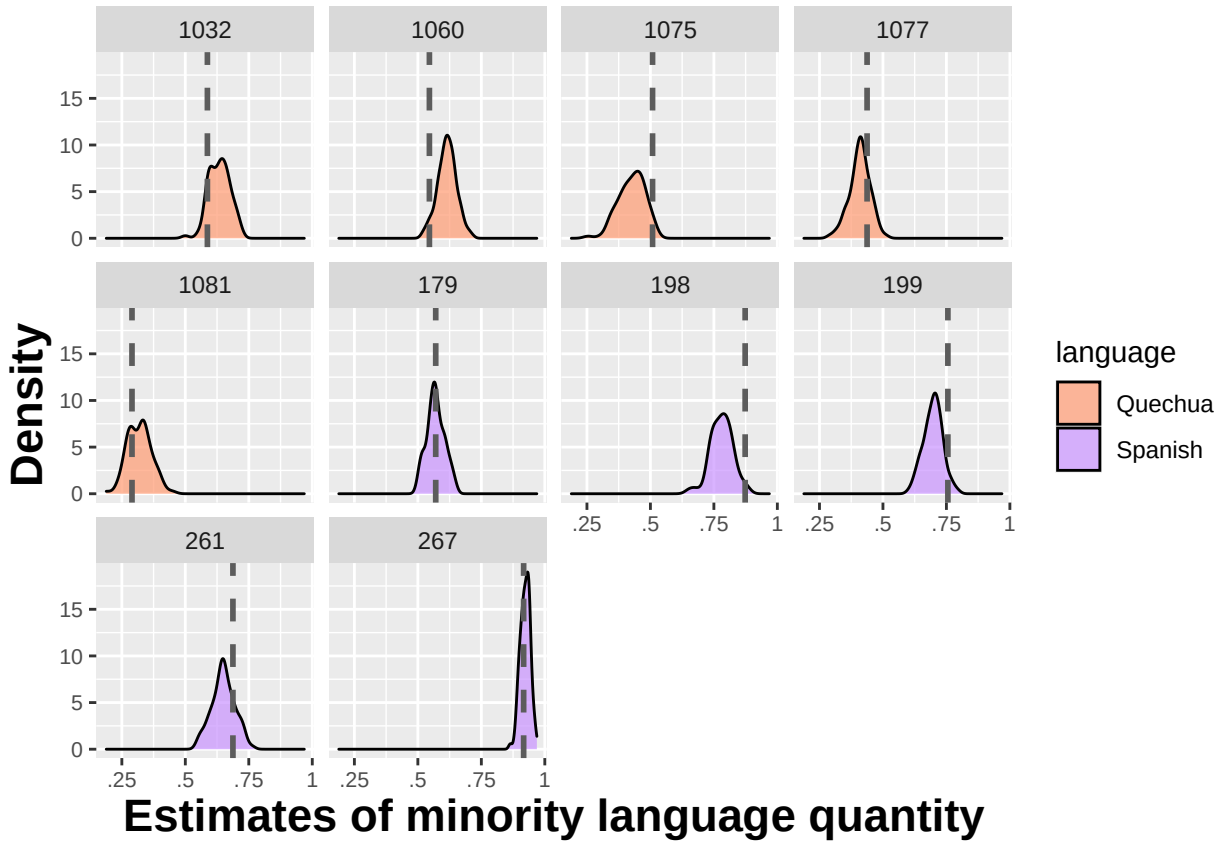
```
lang_annon <- data_annon %>%
  filter(language=='Mixed' | language=='Spanish' | language=='English/Quechua') %>% # only clips where
  group_by(id, method) %>%
  distinct_at(., vars(file_name, language), .keep_all = T) %>% # don't record multiple speakers speakin
  mutate(total_lang_annotations = NROW(file_name)) # N of language annotations made; distinct from N of

que <- lang_annon %>%
  group_by(id, method) %>%
  filter(language=='English/Quechua') %>%
  group_by(method) %>%
  distinct(file_name, .keep_all = T) %>%
  group_by(id, method) %>% # irrespective of speaker/addressee; by-child only
  mutate(n_que=n()) %>%
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_que = n_que / total_lang_annotations) # compute que/eng ratio

span <- lang_annon %>%
  group_by(id, method) %>%
  filter(language=='Spanish') %>%
  group_by(method) %>%
  distinct(file_name, .keep_all = T) %>%
  group_by(id, method) %>%
  mutate(n_span = n())  %>%
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_span = n_span / total_lang_annotations) # compute span ratio

mixed <- lang_annon %>%
  group_by(id, method) %>%
  filter(language=='Mixed') %>%
  group_by(method) %>%
  distinct(file_name, .keep_all = T) %>%
  group_by(id, method) %>%
  mutate(n_mxd = n())  %>%
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_mxd = n_mxd / total_lang_annotations) # compute mixed ratio
```

```
# now simulate 100 minority lang estimates from each child
# take however many clips were used to compute the randomly-sampled estimate
# then compute the prop. of those that are spanish/quechua
# repeat 100X
nsims=100
# total_reg_annotations refers to the # of clips used to estimate language prop
# get that variable
random_lang_clips <- lang_annon %>%
  filter(method=='random') %>%
  distinct_at(., vars(id), .keep_all = T) %>%
  ungroup() %>%
  select(id, total_lang_annotations)

sim_lang_data <- lang_annon %>%
  filter(method=='complete') %>% # we're only sampling from all-day annotations
```

```
  select(-total_lang_annotations) %>% # this is the # of all-day clips annotated and we want # of rando
  merge(., random_lang_clips, by='id') %>%
  group_by(id) %>%
  replicate(nsims, ., simplify = FALSE) %>% # simulate 100 collections of random clips
  map_dfr(~ sample_n(., total_lang_annotations), .id = "simulation") # sample the same # of clips per s

# now compute the Quechua estimate for the Bolivia corpus
que_sim_results <- sim_lang_data %>%
  filter(language=='English/Quechua' & location=='Bolivia') %>%
  group_by(id, simulation) %>%
  distinct(file_name, .keep_all = T) %>%
  mutate(n_que=n()) %>%
  distinct(id, .keep_all = T) %>%
  mutate(percen_que = n_que / total_lang_annotations) # compute que/to all else

# and the Spanish estimate for the US corpus
span_sim_results <- sim_lang_data %>%
  filter(language=='Spanish' & location=='US') %>%
  group_by(id, simulation) %>%
  distinct(file_name, .keep_all = T) %>%
  mutate(n_span=n()) %>%
  distinct(id, .keep_all = T) %>%
  mutate(percen_span = n_span / total_lang_annotations) # compute span/to all else

# now some descriptive stats from those results
# by corpus
que_sim_stats <- que_sim_results %>%
  ungroup() %>%
  summarize(mean_sim_que = round(mean(percen_que),2),
         sd_sim_que = round(sd(percen_que),2),
         min_sim_que = round(range(percen_que)[1],2),
         max_sim_que = round(range(percen_que)[2],2)) %>%
  mutate(sim_stat = paste(mean_sim_que,"(",sd_sim_que,")",min_sim_que,"-",max_sim_que)) %>%
  select(sim_stat)

span_sim_stats <- span_sim_results %>%
  ungroup() %>%
  summarize(mean_sim_span = round(mean(percen_span),2),
         sd_sim_span = round(sd(percen_span),2),
          min_sim_span = round(range(percen_span)[1],2),
         max_sim_span = round(range(percen_span)[2],2)) %>%
  mutate(sim_stat = paste(mean_sim_span,"(",sd_sim_span,")",min_sim_span,"-",max_sim_span)) %>%
  select(sim_stat)

# now the spanish estimate by individual child in US corpus
span_sim_child_stats <- span_sim_results %>%
  group_by(id) %>%
  summarize(mean_sim_span = round(mean(percen_span),2),
         sd_sim_span = round(sd(percen_span),2),
         min_sim_span = round(range(percen_span)[1],2),
         max_sim_span = round(range(percen_span)[2],2)) %>%
  mutate(sim_stat_child = paste(mean_sim_span,"(",sd_sim_span,")",min_sim_span,"-",max_sim_span)) %>%
  select(id, sim_stat_child)
```

```r
# and the quechua estimate by individual child in Bolivia corpus
que_sim_child_stats <- que_sim_results %>%
  group_by(id) %>%
  summarize(mean_sim_que = round(mean(percen_que),2),
         sd_sim_que = round(sd(percen_que),2),
         min_sim_que = round(range(percen_que)[1],2),
         max_sim_que = round(range(percen_que)[2],2)) %>%
  mutate(sim_stat_child = paste(mean_sim_que,"(",sd_sim_que,")",min_sim_que,"-",max_sim_que)) %>%
  select(id, sim_stat_child)
```

```r
span_lang_estimate <- span %>%
  filter(location=='US')
que_lang_estimate <- que %>%
  filter(location=='Bolivia') %>%
  mutate(actual_percen_minority = percen_que)
actual_lang_estimate <- span_lang_estimate %>%
  mutate(actual_percen_minority = percen_span) %>%
  rbind(., que_lang_estimate) %>%
  filter(method=='random') %>%
  select(id, actual_percen_minority)

que_sim_results2 <- que_sim_results %>%
  mutate(sim_percen_minority = percen_que)
minority_sim_results <- span_sim_results %>%
  mutate(sim_percen_minority = percen_span) %>%
  rbind(., que_sim_results2)

dropLeadingZero <- function(l){
  str_replace(l, '0(?=.)', '')
}

#minority language
lang_density <- actual_lang_estimate %>%
  merge(., minority_sim_results, by="id") %>%
  mutate(id=recode(id,"198-9mo"="198","261-8mo"="261","267-12mo"="267")) %>%
  mutate(language=recode(location, "Bolivia" = "Quechua", "US" = "Spanish")) %>%
  ggplot(., aes(x=sim_percen_minority, fill=language)) +
  geom_density(alpha=.75) +
  scale_fill_manual(values=c("lightsalmon", "#CC99FF")) +
  facet_wrap(~id, scales="fixed") +
    scale_x_continuous(breaks = seq(0, 1, .25),
                       labels = dropLeadingZero) +
  geom_vline(aes(xintercept=actual_percen_minority),
             color="gray36", linetype="dashed", size=1) +
  xlab("Estimates of minority language quantity") +
  ylab("Density") +
  theme(axis.text=element_text(size=8),
      axis.title=element_text(size=17,face="bold"))
lang_density
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/min_lang_density_plot.jpeg", height = 4
lang_density
dev.off()
```

```
## pdf
##   2
```

```
vars <- data_annon_cts %>%
  select(percen_ofallclips_drawn, id, method) %>%
  colnames(.)

final_data <- span %>%
  merge(., data_annon_cts, by=vars) %>%
  select(id, num_clips, age_YYMMDD, gender, location, method, percen_span, speech_clips, percen_ofallcli

final_data2 <-
  merge(final_data, que, by=c('id', 'method', 'percen_ofallclips_drawn', 'gender', 'location', 'num_cli
  select(id, gender, location, method, percen_span, percen_que, num_clips, percen_ofallclips_drawn, spe

plot_data <-
  merge(final_data2, mixed, by=c('id', 'method', 'percen_ofallclips_drawn', 'gender', 'location', 'num_
  select(id, gender, location, method, percen_span, percen_que, percen_mxd, num_clips, percen_ofallclips

# sanity check: calculate percen mixed + spanish + english/quechua
plot_data$total <- plot_data$percen_mxd + plot_data$percen_span + plot_data$percen_que
```

```r
# compute correlations
us_cor <- plot_data %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(method, id, percen_span, location) %>%
  spread("method", "percen_span") %>%
  filter(location=='US') %>%
  summarize(., paste("r=",round(cor.test(complete, random)$estimate,2),",","p=",round(cor.test(complete

bo_cor <- plot_data %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(method, id, percen_que, location) %>%
  spread("method", "percen_que") %>%
  filter(location=='Bolivia') %>%
  summarize(., paste("r=",round(cor.test(complete, random)$estimate,2),",","p=",round(cor.test(complete

#AW added below (correlations for all random sampling simulations and descriptive stats for them)
us_cor2 <- plot_data %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(method, id, percen_span, location) %>%
  spread("method", "percen_span") %>%
  filter(location=='US')

us_cor_sim <- span_sim_results %>%
  distinct_at(., vars(simulation, id), .keep_all = T) %>%
  select(simulation, id, percen_span, location) %>%
  spread("simulation", "percen_span") %>%
  merge(., us_cor2, by=c('id', 'location')) %>%
  select(-id, -location)

bo_cor2 <- plot_data %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(method, id, percen_que, location) %>%
  spread("method", "percen_que") %>%
  filter(location=='Bolivia')

bo_cor_sim <- que_sim_results %>%
  distinct_at(., vars(simulation, id), .keep_all = T) %>%
  select(simulation, id, percen_que, location) %>%
  spread("simulation", "percen_que") %>%
  merge(., bo_cor2, by=c('id', 'location')) %>%
  select(-id, -location)

us_corsimmatrix <- as.data.frame(round(cor(us_cor_sim),2))

us_corsimmatrix_stats <- us_corsimmatrix %>%
    summarize(mean_corsim_us = mean(us_corsimmatrix$complete[1:nsims]),
         min_corsim_us = min(us_corsimmatrix$complete[1:nsims]),
         max_corsim_us = max(us_corsimmatrix$complete[1:nsims])) %>%
  mutate(corsim_stat_us = paste(mean_corsim_us,",",min_corsim_us,"-",max_corsim_us)) %>%
  select(corsim_stat_us)

bo_corsimmatrix <- as.data.frame(round(cor(bo_cor_sim),2))
```

```
bo_corsimmatrix_stats <- bo_corsimmatrix %>%
    summarize(mean_corsim_bo = mean(bo_corsimmatrix$complete[1:nsims]),
         min_corsim_bo = min(bo_corsimmatrix$complete[1:nsims]),
         max_corsim_bo = max(bo_corsimmatrix$complete[1:nsims])) %>%
  mutate(corsim_stat_bo = paste(mean_corsim_bo,",",min_corsim_bo,"-",max_corsim_bo)) %>%
  select(corsim_stat_bo)


# calculate absolute and relative errors (we're only using absolute)
us_rel_error <- plot_data %>%
  filter(location=='US') %>%
  group_by(method, id) %>%
  summarize(avg=mean(percen_span)) %>%
  spread(key='method', value='avg') %>%
   mutate(relative_error = ((abs((random - complete)) / complete)*100),
         avg_rel_error = round(mean(relative_error),2),
         sd_rel_error = round(sd(relative_error),2)) %>%
  mutate(rel_error_stats=paste(avg_rel_error,"(",sd_rel_error,")")) %>%
  distinct(rel_error_stats)
us_abs_error <- plot_data %>%
  filter(location=='US') %>%
  distinct_at(vars(id, method, percen_span)) %>%
  mutate(percen_span = round(percen_span,2)) %>%
  spread(key='method', value='percen_span') %>%
  mutate(abs_error = round((abs(random - complete)),2))

bo_rel_error <- plot_data %>%
  filter(location=='Bolivia') %>%
  group_by(method, id) %>%
  summarize(avg=mean(percen_que)) %>%
  spread(key='method', value='avg') %>%
  mutate(relative_error = ((abs((random - complete)) / complete)*100),
         avg_rel_error = round(mean(relative_error),2),
         sd_rel_error = round(sd(relative_error),2)) %>%
  mutate(rel_error_stats=paste(avg_rel_error,"(",sd_rel_error,")")) %>%
  distinct(rel_error_stats)
bo_abs_error <- plot_data %>%
  filter(location=='Bolivia') %>%
  distinct_at(vars(id, method, percen_que)) %>%
  mutate(percen_que = round(percen_que,2)) %>%
  spread(key='method', value='percen_que') %>%
  mutate(abs_error = round((abs(random - complete)),2))

# add correlations to table - will make pretty below
us_lang_tbl2 <- cbind(us_abs_error, us_cor) %>%
  merge(., span_sim_child_stats, by='id') %>%
  mutate(id=recode(id,"179"="Spanish-English (179)", "198-9mo"="Spanish-English (198)", "199"="Spanish-
  relocate(c(id, random, complete, abs_error, sim_stat_child))

bo_lang_tbl2 <- cbind(bo_abs_error, bo_cor) %>%
  merge(., que_sim_child_stats, by='id') %>%
  mutate(id = paste("Quechua-Spanish","(",id,")")) %>%
  relocate(c(id, random, complete, abs_error, sim_stat_child))
```

```
lang_tbl <- rbind(us_lang_tbl2, bo_lang_tbl2)
```

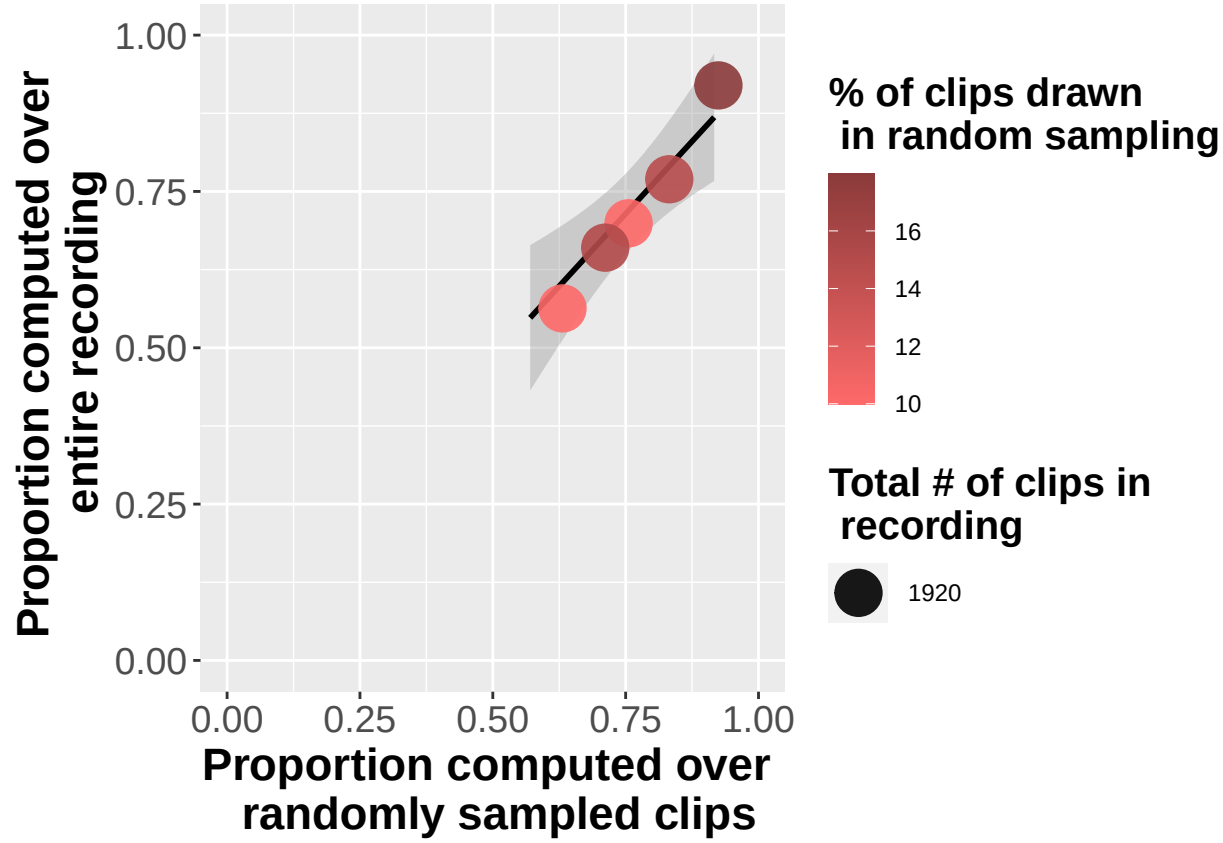
```
knitr::kable(lang_tbl, caption = 'Minority language estimates by child and annotation method.',
             booktabs=T,
             row.names = FALSE,
             col.names = c("Corpus (ID)", "Random", "All-day", "Absolute error", "n=100 simulations of r
  column_spec(1, width = "5.5cm") %>%
    column_spec(4, width = "3cm") %>%
  column_spec(5:6, width = "4cm") %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "Annotation Method" = 2, " " = 3)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

Table 1: (#tab:generate lang tables)Minority language estimates by child and annotation method.

Corpus (ID)	Annotation Method		Absolute error	n=100 simulations of random sampling Avg. (SD) Range	Within-co between ra all-day est
	Random	All-day			
Spanish-English (179)	0.57	0.57	0.00	0.57 (0.03) 0.5 - 0.64	r= 0.96 , p
Spanish-English (198)	0.87	0.78	0.09	0.78 (0.04) 0.65 - 0.89	r= 0.96 , p
Spanish-English (199)	0.76	0.70	0.06	0.7 (0.04) 0.6 - 0.79	r= 0.96 , p
Spanish-English (261)	0.69	0.65	0.04	0.65 (0.05) 0.55 - 0.76	r= 0.96 , p
Spanish-English (267)	0.92	0.92	0.00	0.92 (0.02) 0.86 - 0.97	r= 0.96 , p
Quechua-Spanish (1032)	0.59	0.64	0.05	0.63 (0.04) 0.5 - 0.72	r= 0.9 , p=
Quechua-Spanish (1060)	0.55	0.61	0.06	0.61 (0.04) 0.52 - 0.71	r= 0.9 , p=
Quechua-Spanish (1075)	0.51	0.43	0.08	0.43 (0.05) 0.25 - 0.53	r= 0.9 , p=
Quechua-Spanish (1077)	0.44	0.41	0.03	0.41 (0.04) 0.29 - 0.51	r= 0.9 , p=
Quechua-Spanish (1081)	0.29	0.32	0.03	0.32 (0.05) 0.19 - 0.45	r= 0.9 , p=

```
# for later
per_ann <- plot_data %>%
  filter(method=='random') %>%
  select(id, percen_ofallclips_drawn)

us_plot <- plot_data %>%
  filter(location=='US') %>%
  distinct_at(., vars(method, id), .keep_all = T)%>%
  select(-percen_que, -percen_ofallclips_drawn, -percen_mxd, -speech_clips, -total) %>%
  spread("method", "percen_span") %>%
  merge(., per_ann, by='id') %>%
  distinct(id, .keep_all = T) %>%
ggplot(., aes(random, complete)) +
  geom_smooth(method = "lm", color="black") +
  geom_jitter(aes(size=num_clips,color=round(percen_ofallclips_drawn,2)),alpha=.9,position = position_ji
  scale_size_continuous(range = c(5, 9)) +
  scale_colour_gradient(low='indianred1', high = 'indianred4') +
  ylab("Proportion computed over \n entire recording") +
  xlab("Proportion computed over \n randomly sampled clips") +
```

```
  ylim(0,1) +
  xlim(0,1)+
  #facet_wrap(~location, scales = "free") +
  labs(col='% of clips drawn \n in random sampling') +
       #title = 'Proportion of Spanish clips \n in U.S. corpus') +
 theme(title = element_text(size=18, face="bold"),
   axis.text=element_text(size=14),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15)) +
      guides(size=guide_legend(title="Total # of clips in \n recording"))
us_plot
```

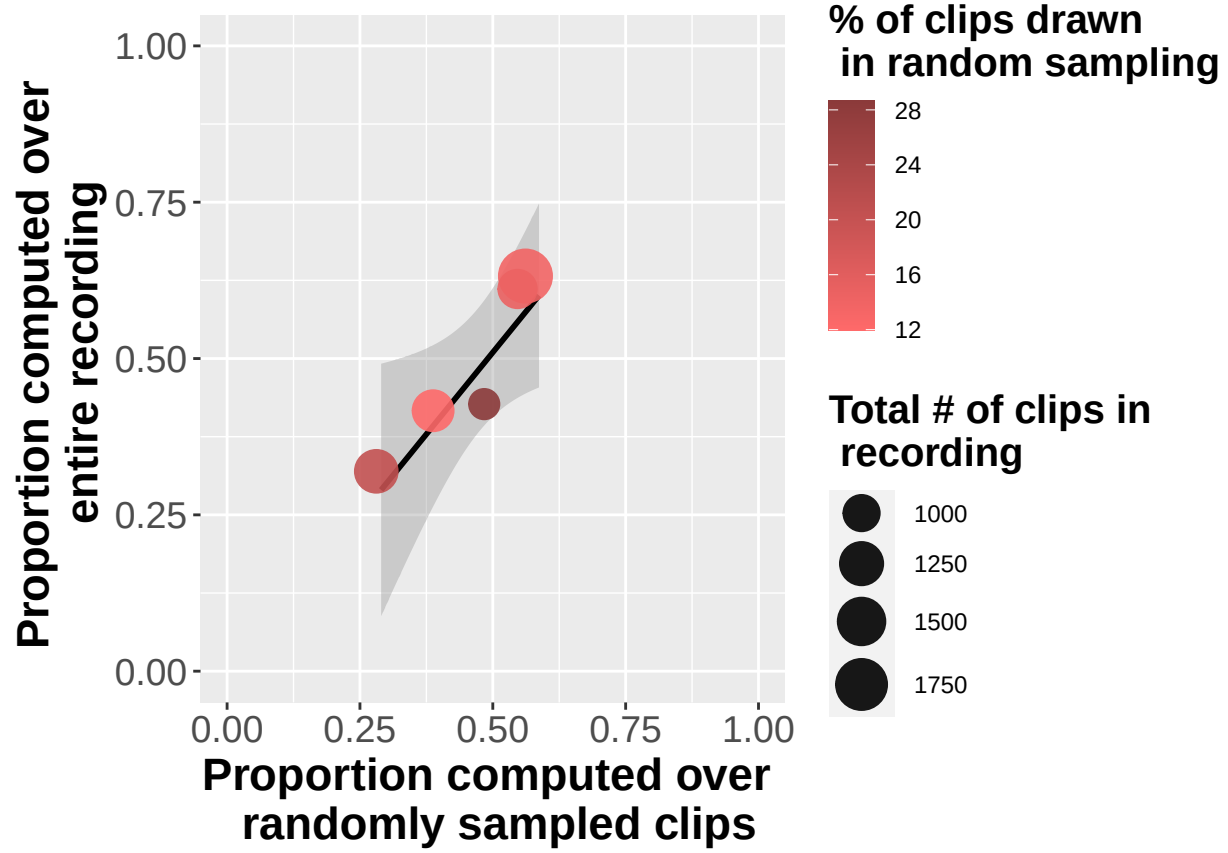



```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/us_plot.jpeg", height = 500, width = 600
us_plot
dev.off()
```

```
## pdf
##   2
```

```
bo_plot <- plot_data %>%
  filter(location=='Bolivia') %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(-percen_span, -percen_ofallclips_drawn, -percen_mxd, -speech_clips, -total) %>%
  spread("method", "percen_que") %>%
  merge(., per_ann, by='id') %>%
  distinct(id, .keep_all = T) %>%
```

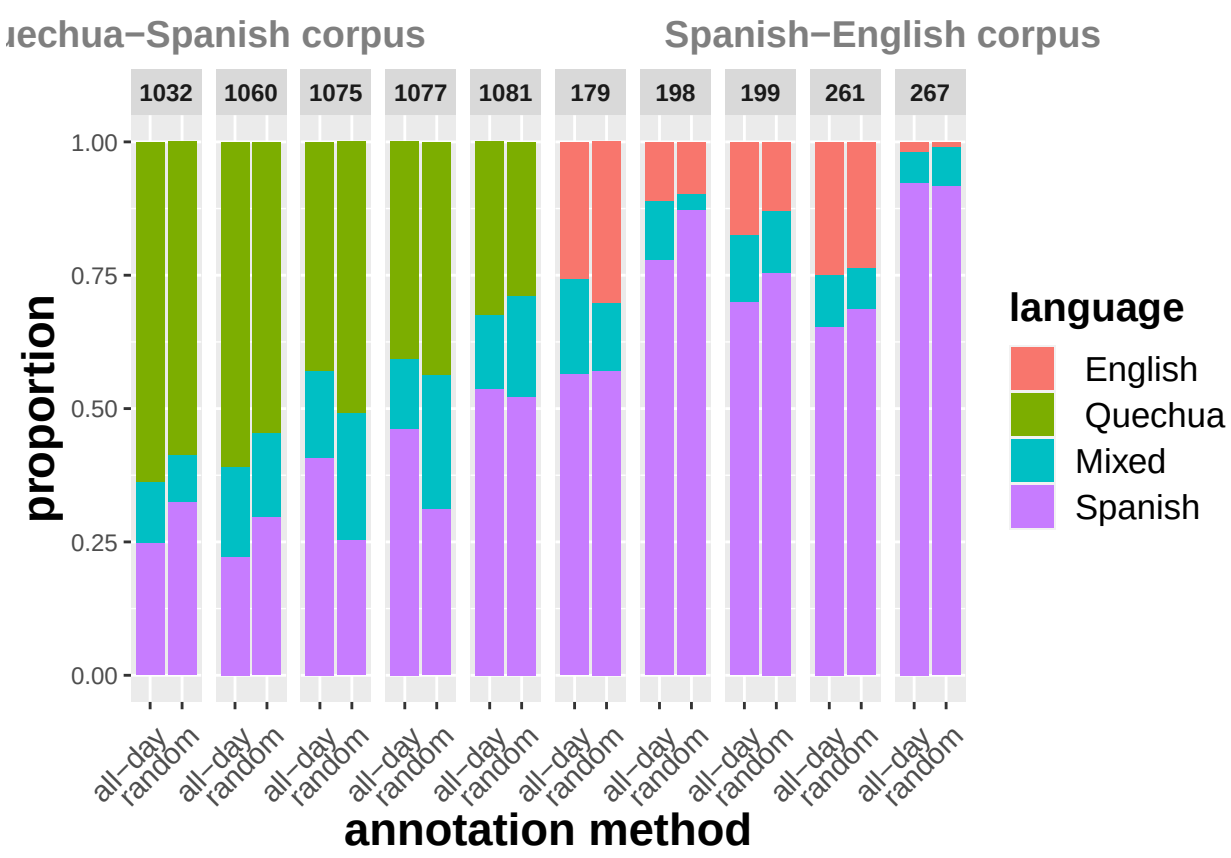
```
ggplot(., aes(random, complete)) +
  geom_smooth(method = "lm", color="black") +
  geom_jitter(aes(size=num_clips,color=round(percen_ofallclips_drawn,2)),alpha=.9,position = position_j
  scale_size_continuous(range = c(5, 9)) +
  scale_colour_gradient(low='indianred1', high = 'indianred4') +
  ylab("Proportion computed over \n entire recording") +
  xlab("Proportion computed over \n randomly sampled clips") +
  ylim(0,1) +
  xlim(0,1)+
  #facet_wrap(~location, scales = "free") +
  labs(col='% of clips drawn \n in random sampling') +
       #title = 'Proportion of Quechua clips \n in Bolivian corpus') +
 theme(title = element_text(size=18, face="bold"),
   axis.text=element_text(size=14),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15))+
       #legend.position = c(.8, .5)) +
      guides(size=guide_legend(title="Total # of clips in \n recording"))
bo_plot
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/bolivia_plot.jpeg", height = 500, width
bo_plot
dev.off()
```

```
## pdf
##   2
```

```r
# finally, we want to actuallly plot the proportions of each language category by child and annotation
lang_props <- plot_data %>%
  gather("language", "proportion", percen_span, percen_que, percen_mxd) %>%
  distinct_at(., vars(id, proportion, language), .keep_all = T) %>%
  mutate(method=plyr::mapvalues(method, "complete", "all-day"),
         id=plyr::mapvalues(id, c("198-9mo", "261-8mo", "267-12mo"), c("198", "261", "267")),
         language=case_when(language=='percen_que' & location=='Bolivia' ~ " Quechua",
                            language=='percen_que' & location=='US' ~ ' English',
                            TRUE ~ as.character(language)),
         language=plyr::mapvalues(language, c("percen_mxd", "percen_span"), c("Mixed", "Spanish"))) %>%
  ggplot(., aes(fill=language, y=proportion, x=method)) +
  geom_bar(position='stack', stat='identity') +
  facet_grid(~id) +
  xlab('annotation method') +
    labs(subtitle = "Quechua-Spanish corpus                    Spanish-English corpus") +
  #labs(title="Proportion of language categories, by child and annotation method",
   #    subtitle = "Quechua-Spanish corpus                                    Spanish-English cor
  theme(axis.text.x = element_text(angle = 45, hjust = .9, vjust=.8, size=11),
        plot.title = element_text(face="bold"),
        plot.subtitle = element_text(color='gray50',hjust = .55, face='bold', size=14),
        axis.title=element_text(size=17,face="bold"),
        legend.title = element_text(size=15,face = "bold"),
        legend.text = element_text(size=13),
        strip.text.x = element_text(size=9, face="bold"))
lang_props
```



```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/stacked_lang_plot.jpeg", height = 500, w
lang_props
dev.off()
```

```
## pdf
##   2
```

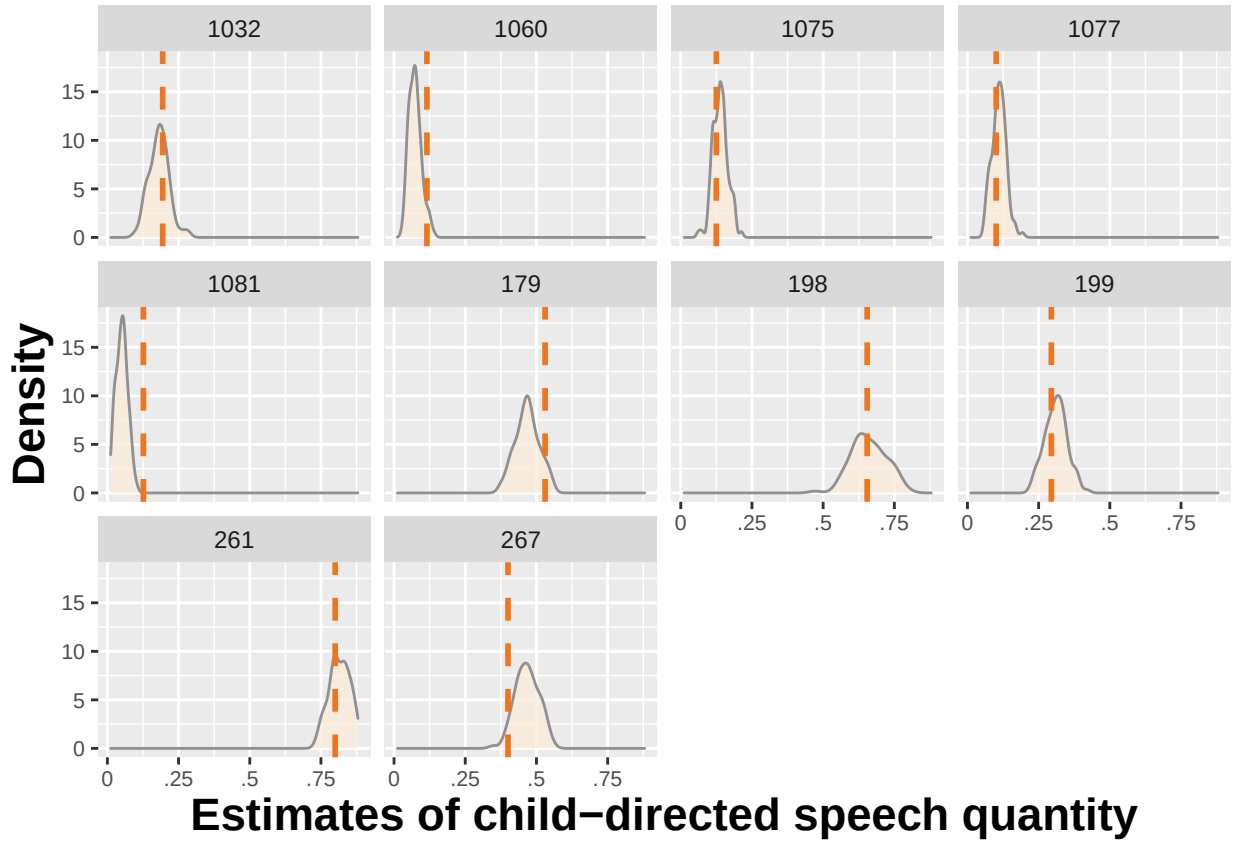
0.0.2 Child-directed speech across random and full methods

```
reg_annon <- data_annon %>%
  filter(addressee=='Adult2TargetChild' | addressee=='Otherchild2TargetChild' | addressee=='Adult2Others
  group_by(id, method) %>%
  distinct_at(., vars(file_name, addressee), .keep_all = T) %>% # don't record multiple speakers speaki
  mutate(total_reg_annotations = NROW(file_name))# N of register annotations made; distinct from N of sp

cds <- reg_annon %>%
  group_by(id, method) %>%
  filter(addressee=='Adult2TargetChild' | addressee=='Otherchild2TargetChild') %>%
  group_by(id, method) %>%
  mutate(n_cds = n()) %>% # # of CDS clips
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_cds = n_cds / total_reg_annotations) %>%
  select(id, num_clips, age_YYMMDD, gender, location, method, percen_cds, n_cds, percen_ofallclips_drawn

ads <- reg_annon %>%
  filter(addressee=='Adult2Others' | addressee=='Otherchild2adults') %>%
  group_by(id, method) %>%
  mutate(n_ads = n()) %>% # # of ADS clips
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_ads = n_ads / total_reg_annotations) %>%
  select(id, num_clips, age_YYMMDD, gender, location, method, percen_ads, n_ads, percen_ofallclips_drawn

o_child <- reg_annon %>%
  filter(addressee=='Adult2OtherChild' | addressee=='Otherchild2OtherChild') %>%
  group_by(id, method) %>%
  mutate(n_ods = n()) %>% # # of ODS clips
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_ods = n_ods / total_reg_annotations) %>%
  select(id, num_clips, age_YYMMDD, gender, location, method, percen_ods, n_ods, percen_ofallclips_drawn

o2 <- merge(cds, ads, all=T)
o3 <- merge(o2, o_child, all = T)
o3[is.na(o3)] <- 0 # one child doesn't have any ODS

# sanity check
o3$total <- o3$percen_ods + o3$percen_ads + o3$percen_cds
```

```
set.seed(1234)
# now simulate 100 CDS estimates from each child
# take however many clips were used to compute the randomly-sampled estimate
# then compute the prop. of those that are CDS
# repeat 100X
```

```r
# total_reg_annotations refers to the # of clips used to estimate speech register
# get that variable
random_clips <- reg_annon %>%
  filter(method=='random') %>%
  distinct_at(., vars(id), .keep_all = T) %>%
  ungroup() %>%
  select(id, total_reg_annotations)

sim_data <- reg_annon %>%
  filter(method=='complete') %>% # we're only sampling from all-day annotations
  select(-total_reg_annotations) %>% # this is the # of all-day clips annotated and we want # of random
  merge(., random_clips, by='id') %>%
  group_by(id) %>%
  replicate(100, ., simplify = FALSE) %>% # simulate 100 collections of random clips
  map_dfr(~ sample_n(., total_reg_annotations), .id = "simulation") # sample the same # of clips per si

# now compute the CDS estimate
cds_sim_results <- sim_data %>%
  group_by(id, simulation) %>%
  filter(addressee=='Adult2TargetChild' | addressee=='Otherchild2TargetChild') %>%
  mutate(n_cds = n()) %>% # # of CDS clips
  distinct(id, .keep_all = T) %>%
  mutate(percen_cds = n_cds / total_reg_annotations)

# now some descriptive stats from those results
cds_sim_stats <- cds_sim_results %>%
  group_by(id) %>%
  summarize(mean_sim_cds = round(mean(percen_cds),2),
         sd_sim_cds = round(sd(percen_cds),2),
         min_sim_cds = round(range(percen_cds)[1],2),
         max_sim_cds = round(range(percen_cds)[2],2)) %>%
  mutate(sim_stat = paste(mean_sim_cds,"(",sd_sim_cds,")",min_sim_cds,"-",max_sim_cds)) %>%
  select(id, sim_stat)

# now compute the ADS estimate
ads_sim_results <- sim_data %>%
  group_by(id, simulation) %>%
  filter(addressee=='Adult2Others' | addressee=='Otherchild2adults') %>%
  mutate(n_ads = n()) %>% # # of CDS clips
  distinct(id, .keep_all = T) %>%
  mutate(percen_ads = n_ads / total_reg_annotations)

# now some descriptive stats from those results
ads_sim_stats <- ads_sim_results %>%
  group_by(location) %>%
  summarize(mean_sim_ads = round(mean(percen_ads),2),
         sd_sim_ads = round(sd(percen_ads),2),
         min_sim_ads = round(range(percen_ads)[1],2),
         max_sim_ads = round(range(percen_ads)[2],2)) %>%
  mutate(sim_stat_ads = paste(mean_sim_ads,"(",sd_sim_ads,")",min_sim_ads,"-",max_sim_ads)) %>%
  select(location, sim_stat_ads)
```

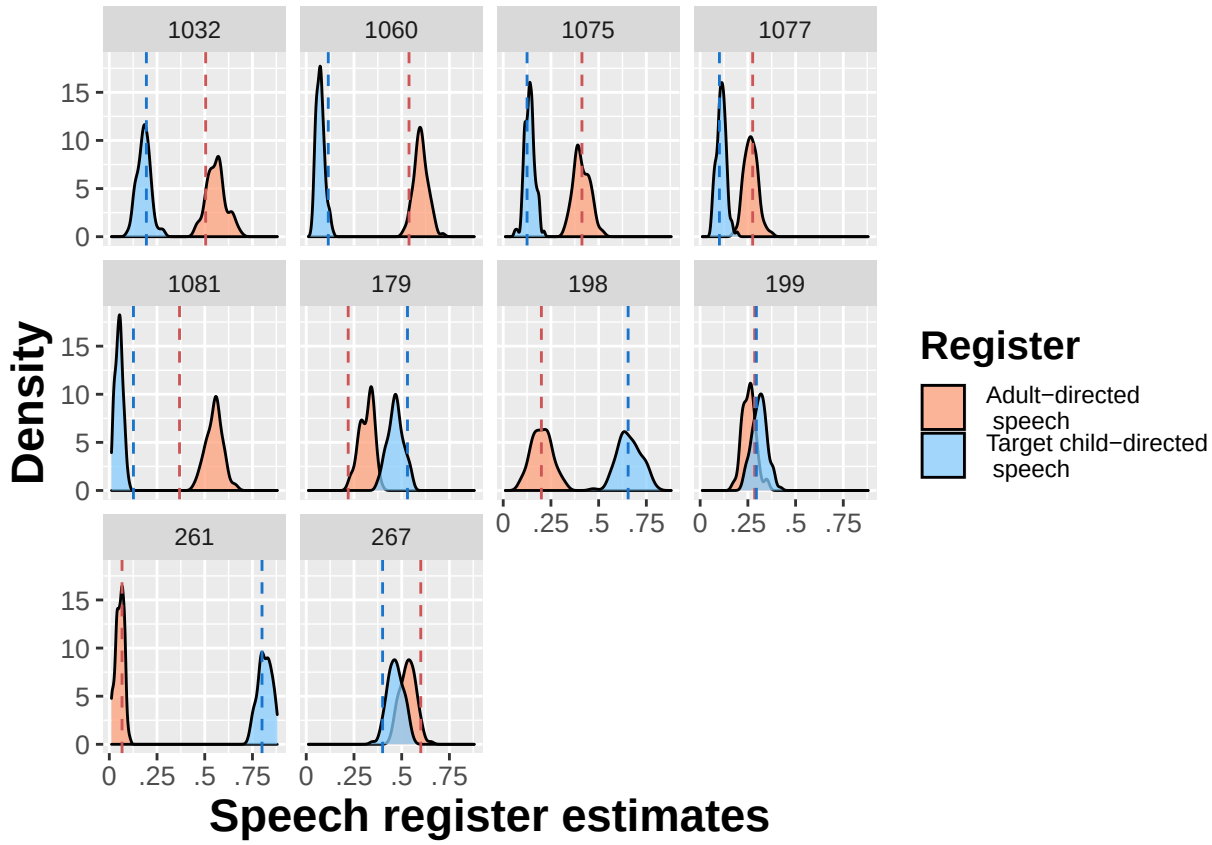
```
#cds alone
cds_density <- o3 %>%
  filter(method=='random') %>%
  mutate(random_percencds_estimate=percen_cds) %>% # get the actual cds estimate from random sampling
  select(id, random_percencds_estimate) %>%
  merge(., cds_sim_results, by="id") %>%
  mutate(id=recode(id,"198-9mo"="198","261-8mo"="261","267-12mo"="267")) %>%
  ggplot(., aes(x=percen_cds)) +
  geom_density(fill="antiquewhite", color='gray57',alpha=.75) +
  facet_wrap(~id, scales="fixed") +
    scale_x_continuous(breaks = seq(0, 1, .25),
                       labels = dropLeadingZero) +
  geom_vline(aes(xintercept=random_percencds_estimate),
             color="chocolate2", linetype="dashed", size=1) +
  xlab("Estimates of child-directed speech quantity") +
  ylab("Density") +
  theme(axis.text=element_text(size=8),
      axis.title=element_text(size=17,face="bold"))
cds_density
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/cds_density_plot.jpeg", height = 400, wi
cds_density
dev.off()
```

```
## pdf
##   2
```

```
reg_density <- o3 %>%
  filter(method=='random') %>%
  mutate(random_percencds_estimate=percen_cds) %>% # get the actual cds estimate from random sampling
  mutate(random_percenads_estimate=percen_ads) %>% # get the actual ads estimate from random sampling
  select(id, random_percencds_estimate, random_percenads_estimate) %>%
  merge(., cds_sim_results, by="id") %>%
  select(id, random_percenads_estimate, random_percencds_estimate, percen_cds, simulation) %>%
  merge(., ads_sim_results, by=c("simulation", "id")) %>%
  group_by(id) %>%
  mutate(avg_percen_cds = mean(percen_cds),
         avg_percen_ads = mean(percen_ads)) %>%
  ungroup() %>%
  gather("Register", "estimate", percen_cds, percen_ads) %>%
  mutate(Register=recode(Register, "percen_ads"='Adult-directed \n speech', "percen_cds"='Target child-
  mutate(id=recode(id,"198-9mo"="198","261-8mo"="261","267-12mo"="267")) %>%
  ggplot(., aes(x=estimate, fill=Register)) +
  geom_density(alpha=.75) +
  scale_fill_manual(values=c("lightsalmon", "skyblue1")) +
  facet_wrap(~id, scales = "fixed") +
  scale_x_continuous(breaks = seq(0, 1, .25),
                     labels = dropLeadingZero) +
  geom_vline(aes(xintercept=random_percenads_estimate),
             color="indianred3", linetype="dashed", size=.5) +
    geom_vline(aes(xintercept=random_percencds_estimate),
             color="dodgerblue3", linetype="dashed", size=.5) +
    #geom_vline(aes(xintercept=avg_percen_ads),
             #color="indianred3", linetype="solid", size=.5) +
    #geom_vline(aes(xintercept=avg_percen_cds),
             #color="dodgerblue3", linetype="solid", size=.5) +
  xlab("Speech register estimates") +
  ylab("Density") +
  theme(axis.text=element_text(size=10),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(face="bold", size=15))
reg_density
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/reg_density_plot.jpeg", height = 400, wi
reg_density
dev.off()
```

```
## pdf
##   2
```

```
# for later
percen_cds_df <- o3 %>%
  distinct_at(., vars(id, method), .keep_all = T) %>%
  filter(method=='random') %>%
  select(id, percen_ofallclips_drawn) # get the % of clips annotated for each id and method

cds_plot_data <- o3 %>%
  select(id, gender, location, num_clips, method, percen_cds) %>%
  spread("method", "percen_cds") %>%
  merge(., percen_cds_df, by='id')

# compute correlations
cds_cors <- cds_plot_data %>%
  group_by(location) %>%
  summarize(., paste("r=",round(cor.test(complete, random)$estimate,2),",","p=",round(cor.test(complete

# also do a correlation for both corpora
cds_cors_all <- cds_plot_data %>%
  summarize(., paste("r=",round(cor.test(complete, random)$estimate,2),",","p=",round(cor.test(complete
```

```r
#AW added below (correlations for cds random sampling simulations and descriptive stats for them)
cds_cor_sim <- cds_sim_results %>%
  distinct_at(., vars(simulation, id), .keep_all = T) %>%
  select(simulation, id, percen_cds, location) %>%
  spread("simulation", "percen_cds") %>%
  merge(., cds_plot_data, by=c('id', 'location'))

cds_cor_sim_us <- cds_cor_sim %>%
  filter(location=='US') %>%
  select(-id, -location, -gender, -num_clips, -percen_ofallclips_drawn)

cds_cor_sim_bo <- cds_cor_sim %>%
  filter(location=='Bolivia') %>%
  select(-id, -location, -gender, -num_clips, -percen_ofallclips_drawn)
  cds_cor_sim_bo[is.na(cds_cor_sim_bo)] <- 0 #replace NA with 0

cds_corsimmatrix_us <- as.data.frame(round(cor(cds_cor_sim_us),2))
cds_corsimmatrix_bo <- as.data.frame(round(cor(cds_cor_sim_bo),2))

cds_corsimstats_us <- cds_corsimmatrix_us %>%
    summarize(mean_corsim_cds_us = mean(cds_corsimmatrix_us$complete[1:nsims]),
         min_corsim_cds_us = min(cds_corsimmatrix_us$complete[1:nsims]),
         max_corsim_cds_us = max(cds_corsimmatrix_us$complete[1:nsims])) %>%
  mutate(corsim_stat_cds_us = paste(mean_corsim_cds_us,",",min_corsim_cds_us,"-",max_corsim_cds_us)) %>%
  select(corsim_stat_cds_us)

cds_corsimstats_bo <- cds_corsimmatrix_bo %>%
    summarize(mean_corsim_cds_bo = mean(cds_corsimmatrix_bo$complete[1:nsims]),
         min_corsim_cds_bo = min(cds_corsimmatrix_bo$complete[1:nsims]),
         max_corsim_cds_bo = max(cds_corsimmatrix_bo$complete[1:nsims])) %>%
  mutate(corsim_stat_cds_bo = paste(mean_corsim_cds_bo,",",min_corsim_cds_bo,"-",max_corsim_cds_bo)) %>%
  select(corsim_stat_cds_bo)

#reg_tbl <- o3 %>%
#  group_by(method, location) %>%
#  summarize(avg=round(mean(percen_cds),2),
#            sd=round(sd(percen_cds),2)) %>%
#  mutate(stats=paste(avg,"(",sd,")")) %>%
#  select(-avg, -sd) %>%
#  spread(key='method', value = "stats")

# calculate absolute and relative errors
cds_rel_error <- o3 %>%
  group_by(id) %>%
  #summarize(avg=mean(percen_cds)) %>%
  select(id,method,percen_cds,location) %>%
  spread(key='method', value='percen_cds') %>%
  mutate(relative_error = round(((abs(random - complete) / complete)*100),2)) %>%
  #mutate(avg_rel_error = round(mean(relative_error),2),
  #       sd_rel_error = round(sd(relative_error),2),
  #       rel_error_stats=paste(avg_rel_error,"(",sd_rel_error,")")) %>%
  distinct(relative_error, .keep_all = T)
cds_abs_error <- o3 %>%
```

```
  group_by(id) %>%
  #summarize(avg=mean(percen_cds)) %>%
  select(id,method,percen_cds,location) %>%
  spread(key='method', value='percen_cds') %>%
  mutate(abs_error = round(abs(random - complete),2)) %>%
  #mutate(avg_rel_error = round(mean(relative_error),2),
  #       sd_rel_error = round(sd(relative_error),2),
  #       rel_error_stats=paste(avg_rel_error,"(",sd_rel_error,")")) %>%
  distinct(abs_error, .keep_all = T)



# add correlations and simulated stats to table - will make pretty below
final_reg_tbl <- cds_abs_error %>%
  merge(., cds_cors, by='location') %>%
  merge(., cds_sim_stats, by='id')

final_reg_tbl$location <-
  plyr::mapvalues(final_reg_tbl$location,
                  from = c("Bolivia", "US"),
                  to =c("Quechua-Spanish", "Spanish-English"))

final_reg_tbl2 <- final_reg_tbl %>%
  mutate(random = round(random,2),
         complete = round(complete,2)) %>%
  mutate(corpus_id = paste(location,"(",id,")")) %>%
  select(-location, -id) %>%
  relocate(corpus_id, random, complete, abs_error, sim_stat)



knitr::kable(final_reg_tbl2, caption = 'Target child-directed speech estimates by child and annotation m
             booktabs=T,
             row.names = FALSE,
             col.names = c("Corpus (ID)", "Random", "All-day", "Absolute error", "n=100 simulations of
  column_spec(1, width = "4cm") %>% # force column headers onto two rows
  column_spec(4, width = "3cm") %>%
  column_spec(5:6, width = "4cm") %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "Annotation Method" = 2, " " = 3)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

```
ads_plot_data <- o3 %>%
  #filter(location=='Bolivia') %>%
  select(id, gender, location, num_clips, method, percen_ads) %>%
  spread("method", "percen_ads") %>%
  merge(., percen_cds_df, by='id')

# compute correlations
ads_cors <- ads_plot_data %>%
  group_by(location) %>%
  summarize(., paste("r=",round(cor.test(complete, random)$estimate,2),",","p=",round(cor.test(complete
```

Table 2: (#tab:cds proportion stats)Target child-directed speech estimates by child and annotation method.

Corpus (ID)	Annotation Method		Absolute error	n=100 simulations of random sampling Avg. (SD) Range	Within-corpus corre between random and all-day estimates
	Random	All-day			
Quechua-Spanish (1032)	0.19	0.17	0.02	0.18 (0.03) 0.1 - 0.28	r= 0.64 , p= 0.24
Quechua-Spanish (1060)	0.12	0.07	0.04	0.07 (0.02) 0.03 - 0.14	r= 0.64 , p= 0.24
Quechua-Spanish (1075)	0.12	0.14	0.02	0.14 (0.03) 0.06 - 0.21	r= 0.64 , p= 0.24
Quechua-Spanish (1077)	0.10	0.11	0.01	0.11 (0.02) 0.06 - 0.19	r= 0.64 , p= 0.24
Quechua-Spanish (1081)	0.13	0.05	0.07	0.05 (0.02) 0.01 - 0.1	r= 0.64 , p= 0.24
Spanish-English (179)	0.53	0.47	0.06	0.47 (0.04) 0.38 - 0.55	r= 0.97 , p= 0.01
Spanish-English (198-9mo)	0.65	0.66	0.01	0.66 (0.06) 0.47 - 0.8	r= 0.97 , p= 0.01
Spanish-English (199)	0.29	0.31	0.02	0.31 (0.04) 0.22 - 0.42	r= 0.97 , p= 0.01
Spanish-English (261-8mo)	0.80	0.82	0.02	0.82 (0.04) 0.73 - 0.88	r= 0.97 , p= 0.01
Spanish-English (267-12mo)	0.40	0.47	0.07	0.47 (0.04) 0.35 - 0.56	r= 0.97 , p= 0.01

```
reg_tbl <- o3 %>%
  group_by(method, location) %>%
  summarize(avg=round(mean(percen_ads),2),
            sd=round(sd(percen_ads),2)) %>%
  mutate(stats=paste(avg,"(",sd,")")) %>%
  select(-avg, -sd) %>%
  spread(key='method', value = "stats")

# calculate absolute and relative errors
ads_rel_error <- o3 %>%
  group_by(method, location, id) %>%
  summarize(avg=mean(percen_ads)) %>%
  spread(key='method', value='avg') %>%
  group_by(id) %>%
  mutate(relative_error = ((abs(random - complete) / complete)*100)) %>%
  ungroup() %>%
  group_by(location) %>%
  mutate(avg_rel_error = round(mean(relative_error),2),
         sd_rel_error = round(sd(relative_error),2),
         rel_error_stats=paste(avg_rel_error,"(",sd_rel_error,")")) %>%
  distinct(rel_error_stats)
ads_abs_error <- o3 %>%
  group_by(method, location, id) %>%
  summarize(avg=mean(percen_ads)) %>%
  spread(key='method', value='avg') %>%
  mutate(abs_error = (abs(random - complete))) %>%
  ungroup() %>%
  group_by(location) %>%
  mutate(avg_abs_error = round(mean(abs_error),2),
         sd_abs_error = round(sd(abs_error),2),
```

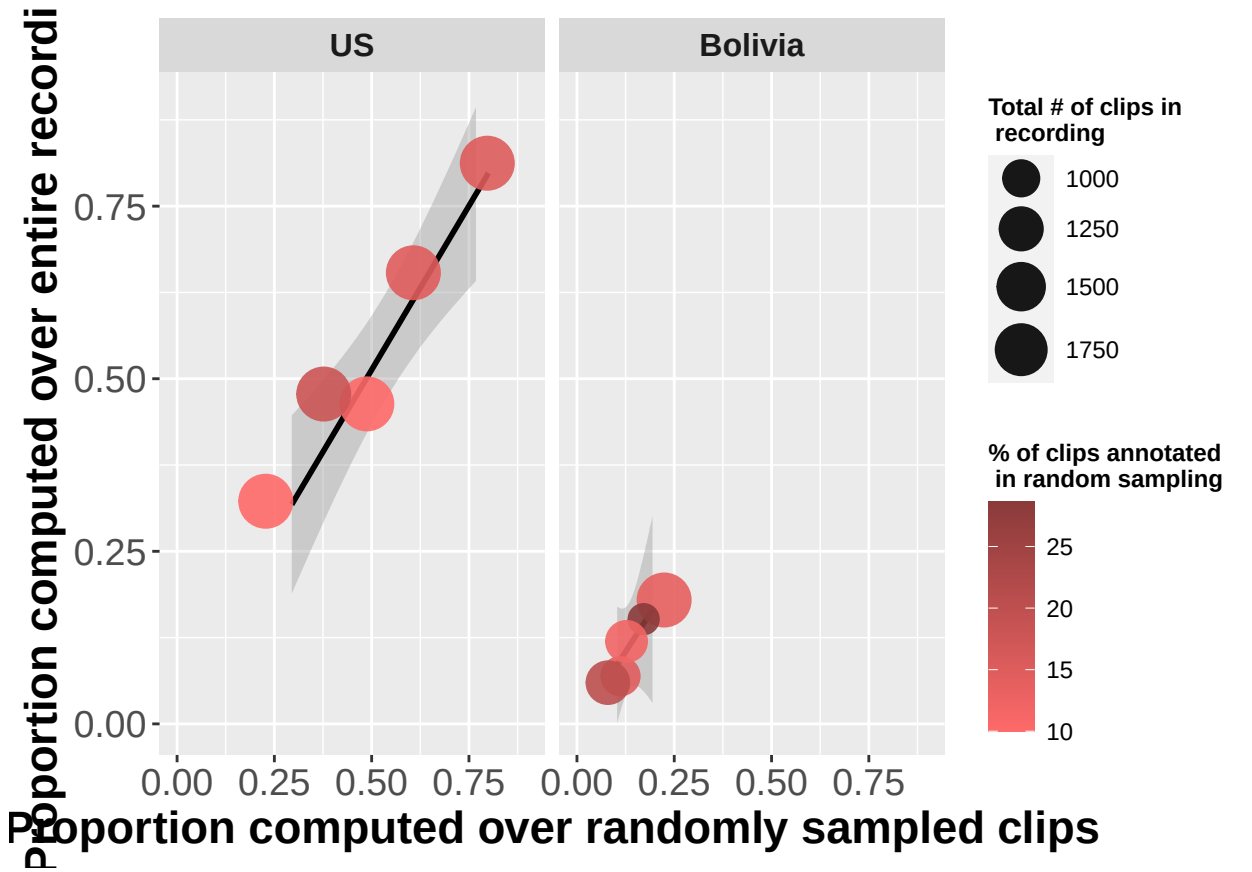
```
        abs_error_stats=paste(avg_abs_error,"(",sd_abs_error,")")) %>%
  distinct(abs_error_stats)

# add correlations to table - will make pretty below
final_reg_tbl <- reg_tbl %>%
  merge(., ads_cors, by='location') %>%
  merge(., ads_abs_error, by='location') %>%
  merge(., ads_sim_stats, by='location') %>%
  relocate(location, random, complete, abs_error_stats, sim_stat_ads)

final_reg_tbl$location <-
  plyr::mapvalues(final_reg_tbl$location,
                  from = c("Bolivia", "US"),
                  to =c("Quechua-Spanish", "Spanish-English"))

knitr::kable(final_reg_tbl, caption = 'Average adult-directed speech estimates by corpus and annotation
             booktabs=T,
             row.names = FALSE,
             col.names = c("Corpus", "Random", "All-day", "Average absolute error (SD)", "n=100 simulat
  column_spec(1, width = "3.5cm") %>%
  column_spec(4, width = "3cm") %>%
  column_spec(5:6, width = "4cm") %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "Annotation Method" = 2, " " = 3)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

Table 3: (#tab:ads proportion stats)Average adult-directed speech estimates by corpus and annotation method.

Corpus	Annotation Method		Average absolute error (SD)	n=100 simulations of random sampling Avg. (SD) Range	Correlation betw estimates
	Random	All-day			
Quechua-Spanish	0.42 (0.11)	0.48 (0.14)	0.07 (0.07)	0.48 (0.13) 0.18 - 0.72	r= 0.84 , p= 0.0
Spanish-English	0.27 (0.2)	0.27 (0.17)	0.04 (0.04)	0.27 (0.16) 0.01 - 0.65	r= 0.95 , p= 0.0

```
# reorder location variable
cds_plot_data$location <- factor(cds_plot_data$location, levels = c("US", "Bolivia"))

cds_plot <- ggplot(cds_plot_data, aes(random, complete)) +
  geom_smooth(method = "lm", color="black") +
  geom_jitter(aes(size=num_clips,color=round(percen_ofallclips_drawn,2)),alpha=.9,position = position_j
  scale_size_continuous(range = c(5, 9)) +
  scale_colour_gradient(low='indianred1', high = 'indianred4') +
  ylab("Proportion computed over entire recording") +
  xlab("Proportion computed over randomly sampled clips") +
  ylim(0,0.9) +
  xlim(0,0.9)+
  facet_wrap(~location, scales = "fixed") +
  labs(col='% of clips annotated \n in random sampling') +
```

```
      #title = 'Proportion of child-directed speech clips \n in U.S. and Bolivian corpora') +
 theme(title = element_text(size=18, face="bold"),
   axis.text=element_text(size=14),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=9),
      #legend.position = c(.85, .55),
      strip.text.x = element_text(size=12, face="bold")) +
      guides(size=guide_legend(title="Total # of clips in \n recording"))
cds_plot
```

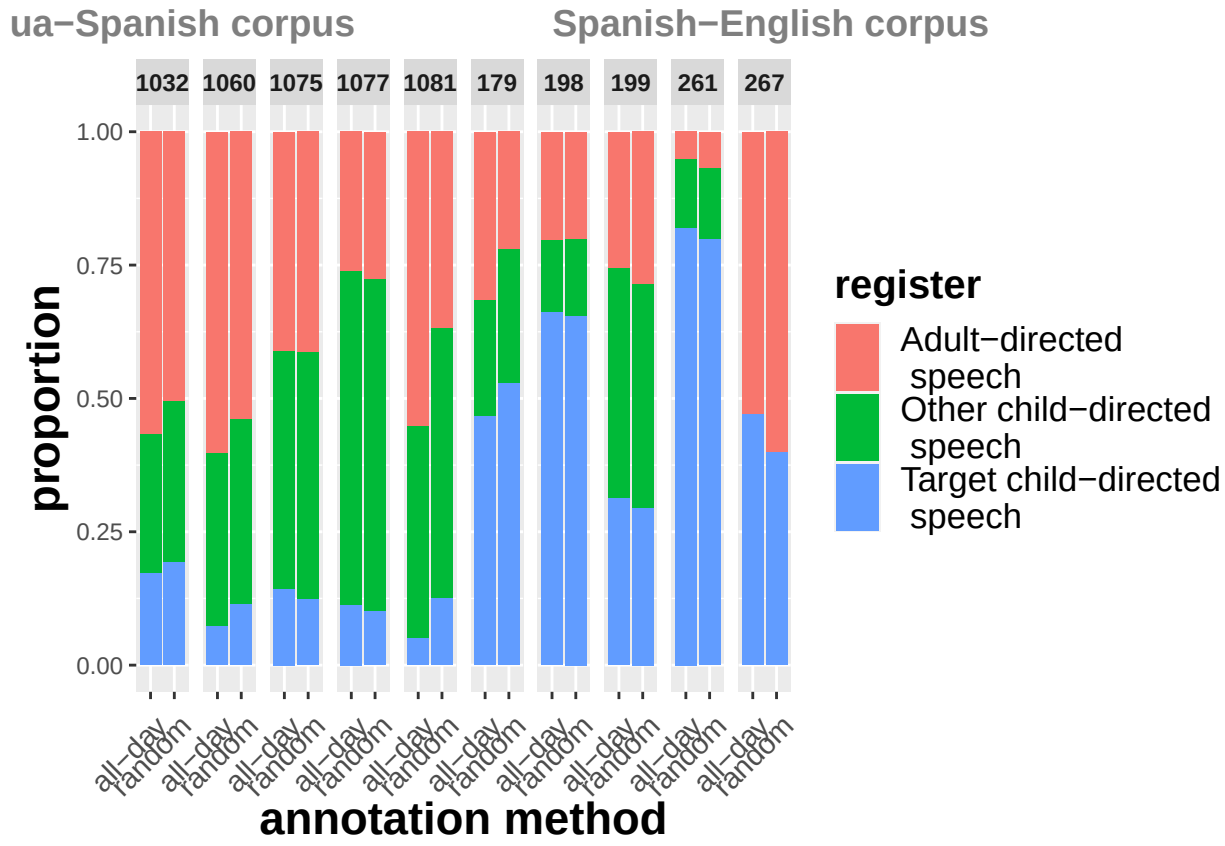



```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/cds_plot.jpeg", height = 500, width = 5
cds_plot
dev.off()
```

```
## pdf
##   2
```

```
# finally, we want to actuallly plot the proportions of each speech register category by child and anno
reg_props <- o3 %>%
  gather("register", "proportion", percen_cds, percen_ods, percen_ads) %>%
  distinct_at(., vars(id, proportion, register), .keep_all = T) %>%
  mutate(method=plyr::mapvalues(method, "complete", "all-day"),
         id=plyr::mapvalues(id, c("198-9mo", "261-8mo", "267-12mo"), c("198", "261", "267")),
         register=plyr::mapvalues(register, c("percen_cds", "percen_ods", "percen_ads"), c("Target chil
  ggplot(., aes(fill=register, y=proportion, x=method)) +
```

```
  geom_bar(position='stack', stat='identity') +
  facet_grid(~id) +
  xlab('annotation method') +
   labs(subtitle = "Quechua-Spanish corpus                    Spanish-English corpus") +
  #labs(title="Proportion of speech register categories, by child and annotation method",
       #subtitle = "Quechua-Spanish corpus                                    Spanish-English corpus
  theme(axis.text.x = element_text(angle = 45, hjust = .9, vjust=.8, size=11),
        plot.title = element_text(face="bold"),
        plot.subtitle = element_text(color='gray50',hjust = .55, face='bold', size=14),
        axis.title=element_text(size=17,face="bold"),
        legend.title = element_text(size=15,face = "bold"),
        legend.text = element_text(size=13),
        strip.text.x = element_text(size=9, face="bold"))
reg_props
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/stacked_register_plot.jpeg", height = 5
reg_props
dev.off()
```

```
## pdf
##   2
```

0.0.3 Part III: language across random and questionnaire methods

```r
# enter questionnaire estimates
ques <- data.frame("id"=c("179", "198-9mo", "199", "261-8mo", "267-12mo"),
                   "ques_est"=c(".74", ".55", ".95", ".73", ".87"))

ques_tbl <- plot_data %>%
  filter(location=='US') %>%
  merge(., ques, by='id') %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(-percen_ofallclips_drawn, -percen_mxd, -percen_que, -speech_clips, -total, -gender, -location,
  mutate(percen_span = round(percen_span,2)) %>%
  spread("method", "percen_span") %>%
  merge(., span_sim_child_stats, by='id') %>%
  relocate(id, random, complete, sim_stat_child)

# compute correlations
ques_random_cors <- ques_tbl %>% # random sampling
  mutate(ques_est = as.numeric(ques_est)) %>%
  summarize(., paste("r=",round(cor.test(ques_est, random)$estimate,3),",","p=",round(cor.test(ques_est
ques_complete_cors <- ques_tbl %>% # all-day sampling
  mutate(ques_est = as.numeric(ques_est)) %>%
  summarize(., paste("r=",round(cor.test(ques_est, complete)$estimate,2),",","p=",round(cor.test(ques_es

ques_random_cors_4 <- ques_tbl %>% # random sampling excluding participant 179 who completed at 28 mont
  filter(id!='179') %>%
  mutate(ques_est = as.numeric(ques_est)) %>%
  summarize(., paste("r=",round(cor.test(ques_est, random)$estimate,2),",","p=",round(cor.test(ques_est
ques_complete_cors_4 <- ques_tbl %>% # all-day sampling excluding participant 179 who completed at 28 m
  filter(id!='179') %>%
  mutate(ques_est = as.numeric(ques_est)) %>%
  summarize(., paste("r=",round(cor.test(ques_est, complete)$estimate,2),",","p=",round(cor.test(ques_es

# create table
knitr::kable(ques_tbl, caption = 'Spanish language estimates in U.S. corpus, by child and estimation me
             booktabs=T,
             row.names = FALSE,
             col.names = c("Child ID", "Random", "All-day", "n=100 simulations of random sampling Avg.
  column_spec(4:5, width = "4cm") %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "From daylong recording" = 3, " " = 1)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

```r
# we also want to know what the results are for the combination of CDS*Spanish, not just Spanish
reg_annon2 <- data_annon %>%
  filter(addressee=='Adult2TargetChild' | addressee=='Otherchild2TargetChild') %>% # only CDS clips
  group_by(id, method) %>%
  distinct_at(., vars(file_name, addressee), .keep_all = T) %>% # don't record multiple speakers speaki
  mutate(total_cds_annotations = NROW(file_name))#

span_cds_tbl <- reg_annon2 %>%
  group_by(id, method) %>%
  filter(addressee=='Adult2TargetChild' | addressee=='Otherchild2TargetChild' & location=='US') %>% # o
  merge(., ques, by='id') %>%
  filter(language=='Spanish') %>% # only Spanish clips
```

Table 4: (#tab:make table for questionnaire method)Spanish language estimates in U.S. corpus, by child and estimation method.

Child ID	From daylong recording			Parental Questionnaire
	Random	All-day	n=100 simulations of random sampling Avg. (SD) Range	
179	0.57	0.57	0.57 (0.03) 0.5 - 0.64	.74
198-9mo	0.87	0.78	0.78 (0.04) 0.65 - 0.89	.55
199	0.76	0.70	0.7 (0.04) 0.6 - 0.79	.95
261-8mo	0.69	0.65	0.65 (0.05) 0.55 - 0.76	.73
267-12mo	0.92	0.92	0.92 (0.02) 0.86 - 0.97	.87

```
  group_by(id, method) %>%
  mutate(n_span_cds = n()) %>% # # of CDS clips where Spanish was spoken
  distinct_at(., vars(id, method), .keep_all = T) %>%
  mutate(percen_span_cds = round(n_span_cds / total_cds_annotations,2)) %>%
  select(method, percen_span_cds, id, ques_est) %>%
  spread("method", "percen_span_cds") %>%
  relocate(id, random, complete)

# compute correlations
cor.test(as.numeric(span_cds_tbl$ques_est), span_cds_tbl$complete)
```

```
##
##  Pearson's product-moment correlation
##
## data:  as.numeric(span_cds_tbl$ques_est) and span_cds_tbl$complete
## t = 0.73817, df = 3, p-value = 0.5139
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7494380  0.9468201
## sample estimates:
##       cor
## 0.3920602
```

```
cor.test(as.numeric(span_cds_tbl$ques_est), span_cds_tbl$random)
```

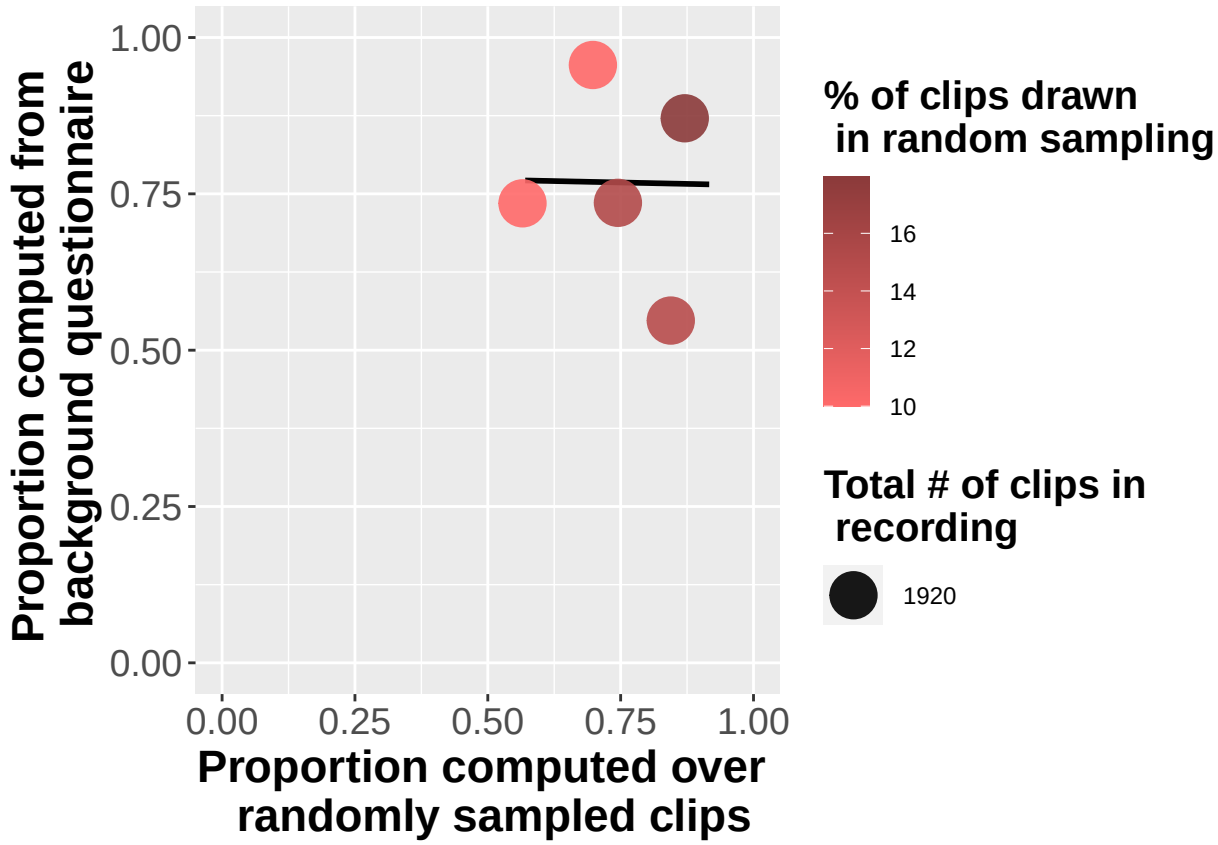
```
##
##  Pearson's product-moment correlation
##
## data:  as.numeric(span_cds_tbl$ques_est) and span_cds_tbl$random
## t = -0.12222, df = 3, p-value = 0.9105
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8969526  0.8656354
## sample estimates:
##         cor
## -0.07038707
```

```r
# simulate 100 estimates from random samples
# total_cds_annotations refers to the # of clips used to estimate CDS*spanish
# what prop. of totalCDS are spoken in Spanish?
random_cds_clips <- reg_annon2 %>%
  filter(method=='random' & location=='US') %>%
  distinct_at(., vars(id), .keep_all = T) %>%
  ungroup() %>%
  select(id, total_cds_annotations)

cdsspan_sim_data <- reg_annon2 %>%
  filter(method=='complete' & location=='US') %>% # we're only sampling from all-day annotations
  select(-total_cds_annotations) %>% # this is the # of all-day clips annotated and we want # of random
  merge(., random_cds_clips, by='id') %>%
  group_by(id) %>%
  replicate(100, ., simplify = FALSE) %>% # simulate 100 collections of random clips
  map_dfr(~ sample_n(., total_cds_annotations), .id = "simulation") # sample the same # of clips per si

# now compute the CDS*spanish estimate
cdsspan_sim_results <- cdsspan_sim_data %>%
  group_by(id, simulation) %>%
  filter(language=='Spanish') %>%
  mutate(n_cdsspan = n()) %>% # # of spanish clips amongst these CDS clips
  distinct(id, .keep_all = T) %>%
  mutate(percen_cdsspan = n_cdsspan / total_cds_annotations)

# now some descriptive stats from those results
cdsspan_sim_stats <- cdsspan_sim_results %>%
  group_by(id) %>%
  summarize(mean_sim_cdsspan = round(mean(percen_cdsspan),2),
            sd_sim_cdsspan = round(sd(percen_cdsspan),2),
            min_sim_cdsspan = round(range(percen_cdsspan)[1],2),
            max_sim_cdsspan = round(range(percen_cdsspan)[2],2)) %>%
  mutate(sim_stat_cdsspan = paste(mean_sim_cdsspan,"(",sd_sim_cdsspan,")",min_sim_cdsspan,"-",max_sim_c
  select(id, sim_stat_cdsspan)

# now combine the simulated data with the span*cds table
span_cds_tbl2 <- span_cds_tbl %>%
  merge(., cdsspan_sim_stats, by='id') %>%
  relocate(id, random, complete, sim_stat_cdsspan)


# create table
knitr::kable(span_cds_tbl2, caption = 'Spanish language in child-directed speech \n estimates in U.S. c
             booktabs=T,
             row.names = FALSE,
             col.names = c("Child ID", "Random", "All-day", "n=100 simulations of random sampling Avg.
  column_spec(1:3, width = "2cm") %>%
  column_spec(4:5, width = "4cm") %>%
  kable_styling() %>%
  add_header_above(c(" " = 1, "From daylong recording" = 3, " " = 1)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

Table 5: Spanish language in child-directed speech estimates in U.S. corpus, by child and estimation method.

Child ID	From daylong recording			Parental Questionnaire
	Random	All-day	n=100 simulations of random sampling Avg. (SD) Range	
179	0.53	0.52	0.52 (0.06) 0.41 - 0.69	.74
198-9mo	0.78	0.64	0.64 (0.07) 0.47 - 0.86	.55
199	0.64	0.66	0.66 (0.08) 0.45 - 0.85	.95
261-8mo	0.55	0.48	0.48 (0.05) 0.35 - 0.57	.73
267-12mo	0.82	0.87	0.86 (0.05) 0.74 - 0.97	.87

```
# for later
per_ann <- plot_data %>%
  filter(method=='random' & location=='US') %>%
  select(id, percen_ofallclips_drawn)

ques_plot <- plot_data %>%
  filter(location=='US') %>%
  merge(., ques, by='id') %>%
  distinct_at(., vars(method, id), .keep_all = T) %>%
  select(-percen_que, -percen_ofallclips_drawn, -percen_mxd, -speech_clips, -total) %>%
  spread("method", "percen_span") %>%
  select(-complete) %>%
  merge(., per_ann, by='id') %>%
  distinct(id, .keep_all = T) %>%
ggplot(., aes(as.numeric(random), as.numeric(ques_est))) +
  geom_smooth(method = "lm", color="black", se=FALSE) +
  geom_jitter(aes(size=num_clips,color=round(percen_ofallclips_drawn,2)),alpha=.9,position = position_ji
  scale_size_continuous(range = c(5, 9)) +
  scale_colour_gradient(low='indianred1', high = 'indianred4') +
  ylab("Proportion computed from \n background questionnaire") +
  xlab("Proportion computed over \n randomly sampled clips") +
  ylim(0,1) +
  xlim(0,1)+
  labs(col='% of clips drawn \n in random sampling') +
       #title = 'Proportion of Spanish clips \n in U.S. corpus: random sampling and background question
 theme(title = element_text(size=18, face="bold"),
   axis.text=element_text(size=14),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15)) +
      guides(size=guide_legend(title="Total # of clips in \n recording"))
ques_plot
```




```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/ques_plot.jpeg", height = 500, width = 
ques_plot
dev.off()
```

```
## pdf
##   2
```

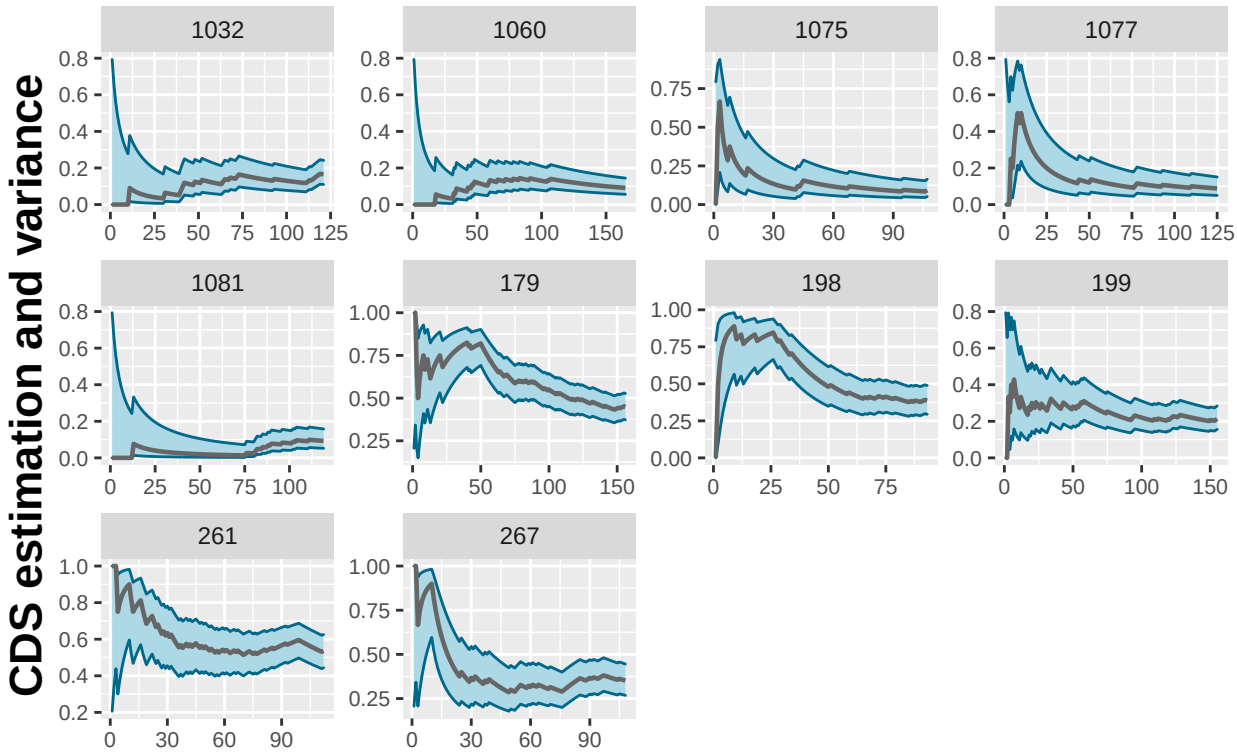
0.0.4 Part I: Running variance

```
reg_annon <- data_annon %>%
  #filter(addressee=='Adult2TargetChild' |
   #       addressee=='Otherchild2TargetChild' |
    #       addressee=='Adult2Others' |
     #       addressee=='Otherchild2adults' |
      #      addressee=='Adult2OtherChild' |
       #     addressee=='Otherchild2OtherChild') %>% # option to omit clips where there is language but 
  group_by(id, method) %>%
  distinct_at(., vars(file_name, addressee), .keep_all = T) %>% # don't record the same register 2x in 
  mutate(total_reg_annotations = NROW(file_name)) # N of register annotations made; distinct from N
# of speech clips since one clip could contain multiple distinct registers

reg_annon$id <- plyr::mapvalues(reg_annon$id,
                from=c("198-9mo", "261-8mo", "267-12mo"),
                to=c("198", "261", "267"))

cds_rolling <- reg_annon %>%
```

```
  filter(method!='complete') %>%
  group_by(id) %>%
  arrange(file_name) %>% # scramble the register annotations
  mutate(annotation_num = as.numeric(1:n())) %>% # where n is the total # of speech register annotation
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(cds_cts=recode(addressee, "Adult2TargetChild"="1", "Otherchild2TargetChild"="1",
                        "Otherchild2OtherChild"="0", "Otherchild2adults"="0",
                        "Adult2OtherChild"="0",  "Adult2Others"="0", "Otherchild2unsure"="0",
                        "Adult2unsure"="0")) %>%
  mutate(cds_cts = as.numeric(cds_cts)) %>%
  mutate(cds_running_cts = as.numeric(cumsum(cds_cts))) %>%
  mutate(roll_prop_cds = cds_running_cts / annotation_num,
         roll_mean_cds = rollmean(roll_prop_cds, k=10, fill = NA),
         roll_sd_cds = rollapply(roll_prop_cds, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
cds_rolling2 <- cds_rolling %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(cds_running_cts, annotation_num, methods = 'wilson', conf.level = .95))
  merge(., cds_rolling, by = c('id', 'annotation_num'))


# -------- now do the same for the all-day annotation  method -----------
# now do the same but for the complete data
cds_rolling_complete <- reg_annon %>%
  filter(method!='random') %>%
  group_by(id) %>%
  arrange(file_name) %>% # scramble the register annotations
  mutate(annotation_num = as.numeric(1:n())) %>% # where n is the total # of speech register annotation
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(cds_cts=recode(addressee, "Adult2TargetChild"="1", "Otherchild2TargetChild"="1",
                        "Otherchild2OtherChild"="0", "Otherchild2adults"="0",
                        "Adult2OtherChild"="0",  "Adult2Others"="0", "Otherchild2unsure"="0",
                        "Adult2unsure"="0")) %>%
  mutate(cds_cts = as.numeric(cds_cts)) %>%
  mutate(cds_running_cts = as.numeric(cumsum(cds_cts))) %>%
  mutate(roll_prop_cds = cds_running_cts / annotation_num,
         roll_mean_cds = rollmean(roll_prop_cds, k=10, fill = NA),
         roll_sd_cds = rollapply(roll_prop_cds, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
cds_rolling_complete2 <- cds_rolling_complete %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(cds_running_cts, annotation_num, methods = 'wilson', conf.level = .95))
  merge(., cds_rolling_complete, by = c('id', 'annotation_num'))
```

```
# ------------ for models, compute binomial confidence interval in 5-clip batches --------------
#cds_batches <- cds_rolling %>%
#  group_by(id) %>%
#  mutate(five_clip_batch = as.integer(gl(n(), 5, n())) * 5,
#         five_clip_batch = replace(five_clip_batch,  ave(five_clip_batch, five_clip_batch, FUN = lengt
#  ungroup %>%
#  fill(five_clip_batch) #%>%

#cds_batches2 <- cds_batches %>%
#  group_by(id, five_clip_batch) %>%
#  summarize(five_cis = binom.confint(cds_running_cts, 5, methods = 'wilson', conf.level = .95)) %>%
#  merge(., cds_batches, by = c('id', 'five_clip_batch'))
```

```
cds_var_plot <- cds_rolling2 %>%
#filter(roll_sd_cds!='NA') %>% # remove rows where variance wasn't estimated
mutate(mean_ci = cis$mean,
       upper_ci = cis$upper,
       lower_ci = cis$lower) %>%
ggplot(., aes(annotation_num, roll_prop_cds)) +
  #geom_line(aes(y=rollapply(roll_prop_cds, 10, FUN=sd, fill=NA))) +
  geom_ribbon(aes(ymax=upper_ci, ymin=lower_ci), fill='lightblue', color='deepskyblue4') +
    geom_line(aes(y=mean_ci), color='gray40', size=.8) +
  xlab("# of speech register annotations") +
  ylab("CDS estimation and variance") +
  facet_wrap(~id, scales = "free") +
  #title = 'Variance in child-directed estimation as a function of clips annotated') +
 theme(title = element_text(size=12),
   axis.text=element_text(size=8),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15)) +
  labs(caption = "'# of speech register annotations' includes 'unsure' speech register annotations.")
cds_var_plot
```



'# of speech register annotations' includes 'unsure' speech register annotations.

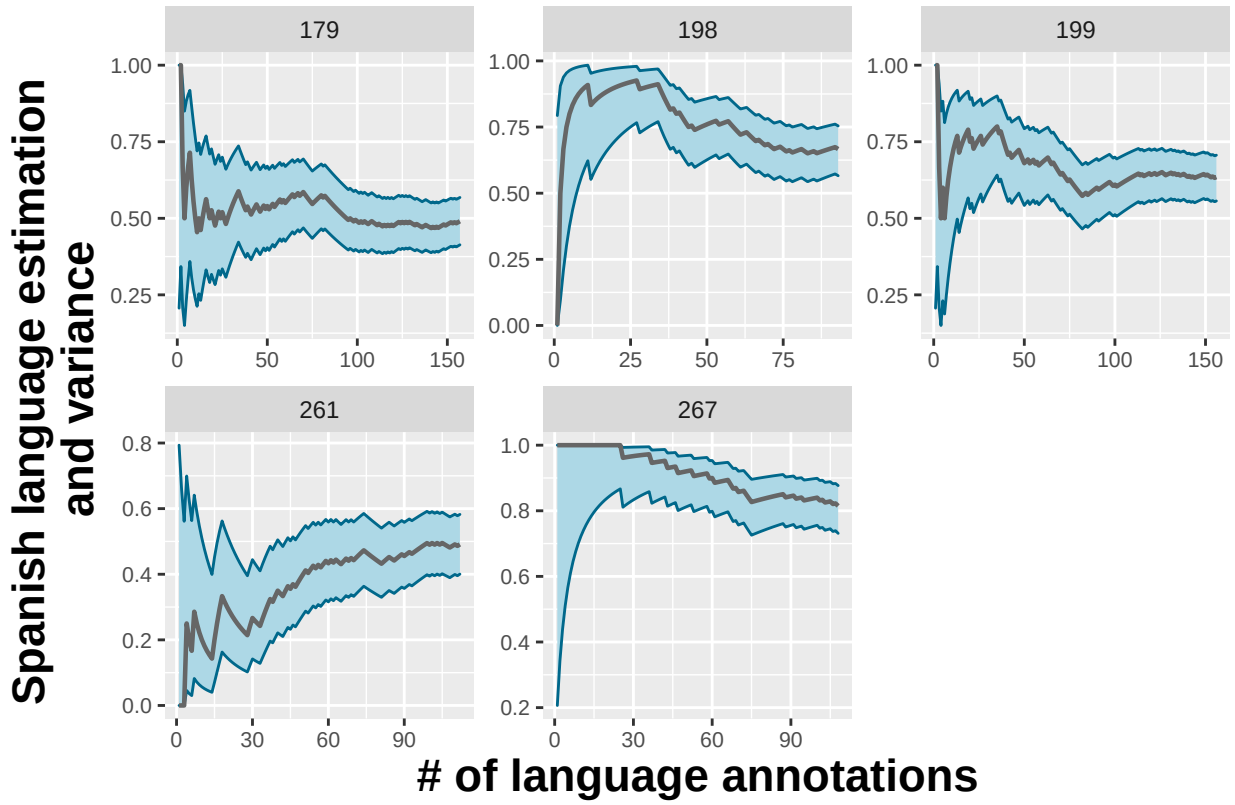
```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/cds_CI_var_plot.jpeg", height = 450, wi
cds_var_plot
dev.off()
```

```
## pdf
##   2
```

```
# now calculate rolling variances for US (Spanish)
lang_annon <- data_annon %>%
  #filter(language=='Mixed' | language=='Spanish' | language=='English/Quechua') %>% # option to omit u
  group_by(id, method) %>%
  distinct_at(., vars(file_name, language), .keep_all = T) %>% # don't record the same language 2x in t
  mutate(total_lang_annotations = NROW(file_name)) # N of language annotations made; distinct from N of

lang_annon$id <- plyr::mapvalues(lang_annon$id,
                                 from=c("198-9mo", "261-8mo", "267-12mo"),
                                 to=c("198", "261", "267"))

# the code below is just for US data, not Bolivia
# ---------------------- #

span_rolling <- lang_annon %>%
  filter(location=='US' & method=='random') %>%
  group_by(id) %>%
  arrange(file_name) %>%
  mutate(annotation_num = as.numeric(1:n())) %>%
```

```
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(span_cts=recode(language, "Unsure"="0", "English/Quechua"="0", "Mixed"="0", "Spanish"="1")) %>%
  mutate(span_cts = as.numeric(span_cts)) %>%
  mutate(span_running_cts = as.numeric(cumsum(span_cts))) %>%
  mutate(roll_prop_span = span_running_cts / annotation_num,
         roll_mean_span = rollmean(roll_prop_span, k=10, fill = NA),
         roll_sd_span = rollapply(roll_prop_span, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
span_rolling2 <- span_rolling %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(span_running_cts, annotation_num, methods = 'wilson', conf.level = .95)) %>%
  merge(., span_rolling, by = c('id', 'annotation_num'))
```

```
span_var_plot <- span_rolling2 %>%
#filter(roll_sd_span!='NA') %>% # remove rows where variance wasn't estimated
    mutate(mean_ci = cis$mean,
         upper_ci = cis$upper,
         lower_ci = cis$lower) %>%
ggplot(., aes(annotation_num, roll_prop_span)) +
  geom_ribbon(aes(ymax=upper_ci, ymin=lower_ci), fill='lightblue', color='deepskyblue4') +
  geom_line(aes(y=mean_ci), color='gray40', size=.8) +
  xlab("# of language annotations") +
  ylab("Spanish language estimation \n and variance") +
  facet_wrap(~id, scales = "free") +
  #title = 'Variance in Spanish language estimation as a function of clips drawn: US corpus') +
 theme(title = element_text(size=12),
   axis.text=element_text(size=8),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15))  +
  labs(caption = "'# of language annotations' includes 'unsure' language annotations.")
span_var_plot
```



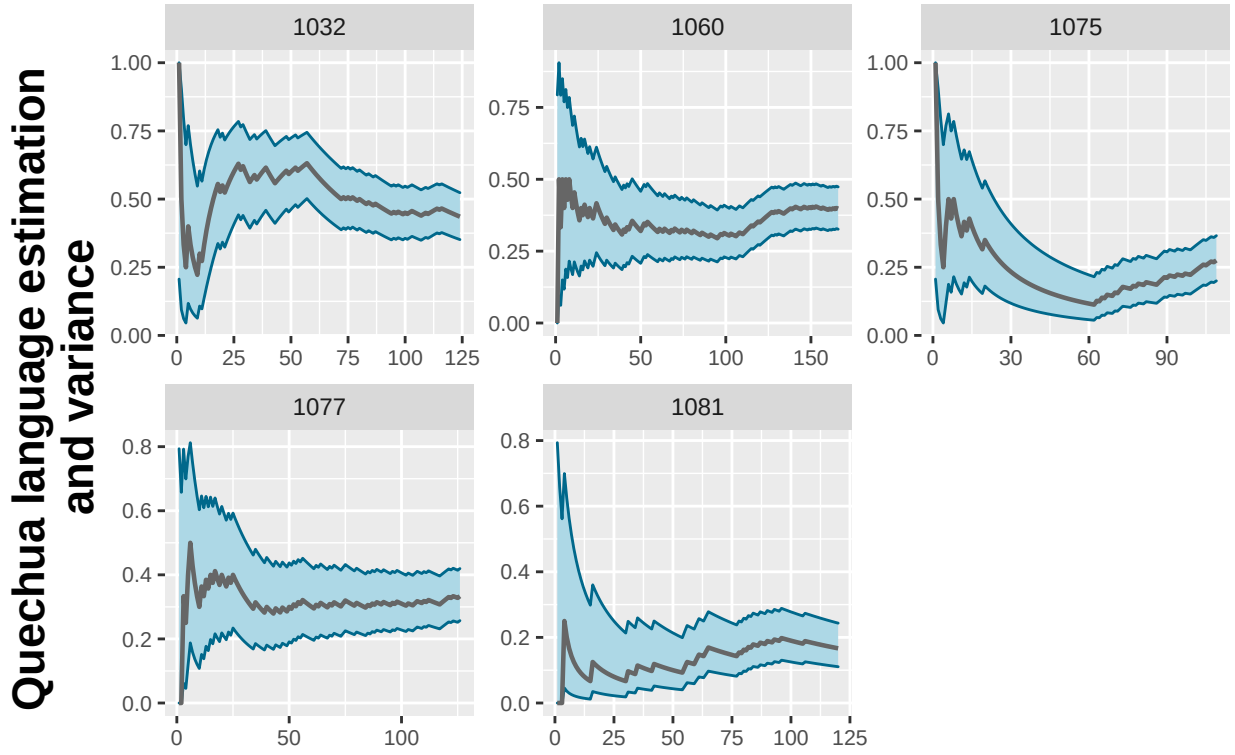
'# of language annotations' includes 'unsure' language annotations.

```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/span_CI_var_plot.jpeg", height = 450, w
span_var_plot
dev.off()
```

```
## pdf
##   2
```

```
que_rolling <- lang_annon %>%
  filter(location=='Bolivia' & method=='random') %>%
  group_by(id) %>%
  arrange(file_name) %>%
  mutate(annotation_num = as.numeric(1:n())) %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(que_cts=recode(language, "Unsure"="0", "English/Quechua"="1", "Mixed"="0", "Spanish"="0")) %>%
  mutate(que_cts = as.numeric(que_cts)) %>%
  mutate(que_running_cts = as.numeric(cumsum(que_cts))) %>%
  mutate(roll_prop_que = que_running_cts / annotation_num,
         roll_mean_que = rollmean(roll_prop_que, k=10, fill = NA),
         roll_sd_que = rollapply(roll_prop_que, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
que_rolling2 <- que_rolling %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(que_running_cts, annotation_num, methods = 'wilson', conf.level = .95))
  merge(., que_rolling, by = c('id', 'annotation_num'))
```

```
que_var_plot <- que_rolling2 %>%
#filter(roll_sd_que!='NA') %>% # remove rows where variance wasn't estimated
    mutate(mean_ci = cis$mean,
         upper_ci = cis$upper,
         lower_ci = cis$lower) %>%
ggplot(., aes(annotation_num, roll_prop_que)) +
  geom_ribbon(aes(ymax=upper_ci, ymin=lower_ci), fill='lightblue', color='deepskyblue4') +
  geom_line(aes(y=mean_ci), color='gray40', size=.8) +
  xlab("# of language annotations") +
  ylab("Quechua language estimation \n and variance") +
  facet_wrap(~id, scales = "free") +
  #title = 'Variance in Quechua language estimation as a function of clips drawn: Bolivia corpus') +
 theme(title = element_text(size=12),
   axis.text=element_text(size=8),
      axis.title=element_text(size=17,face="bold"),
      legend.title = element_text(size=15)) +
  labs(caption = "'# of language annotations' includes 'unsure' language annotations.")
que_var_plot
```



'# of language annotations' includes 'unsure' language annotations.

```
jpeg("/Users/megcychosz/Google Drive/biling_CDS/results/figures/que_CI_var_plot.jpeg", height = 450, wi
que_var_plot
dev.off()
```

```
## pdf
##   2
```

```r
# report CI ranges at 80-clip mark and when annotation stopped, by child
que_cis_table <- que_rolling2 %>%
  group_by(id) %>%
  filter(annotation_num==80 | annotation_num==NROW(id)) %>% # get values at 80-clip mark and cut-off
  mutate(ci_range = cis$upper - cis$lower)

lang_cis_table <- span_rolling2 %>%
  group_by(id) %>%
  filter(annotation_num==80 | annotation_num==NROW(id)) %>% # get values at 80-clip mark and cut-off
  mutate(ci_range = cis$upper - cis$lower) %>%
  rbind(., que_cis_table) %>%
  select(id, annotation_num, ci_range) %>%
  mutate(ci_range = round(ci_range,2)) %>%
  mutate(timept = if_else(annotation_num==80, '80-clip_lang', 'Cut-off_lang')) %>%
  select(-annotation_num) %>%
  spread("timept", "ci_range")


final_cis_table <- cds_rolling2 %>%
  group_by(id) %>%
  filter(annotation_num==80 | annotation_num==NROW(id)) %>% # get values at 80-clip mark and cut-off
  mutate(ci_range = cis$upper - cis$lower) %>%
  select(id, annotation_num, ci_range) %>%
  mutate(ci_range = round(ci_range,2)) %>%
  mutate(timept = if_else(annotation_num==80, '80-clip', 'Cut-off')) %>%
  select(-annotation_num) %>%
  spread("timept", "ci_range") %>%
  merge(., lang_cis_table, by='id')

knitr::kable(final_cis_table, caption = 'Confidence interval range for Spanish/Quechua and child-directe
             booktabs=T,
             row.names = FALSE,
             col.names = c("Child ID", "80-clip", "Cut-off", "80-clip", "Cut-off")) %>% # "
  kable_styling() %>%
  add_header_above(c(" " = 1, "Language" = 2, "Child-directed speech" = 2)) %>%
  kableExtra::kable_styling(latex_options = "hold_position")
```

```r
# cds model
# build data sets
cds_model_data_firstthird <- cds_rolling2 %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2)) %>% # for a sanity check
  filter(row_number() <= n()*.33) # get the bottom 50% of rows from each group

cds_model_data_secondthird <- cds_rolling2 %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2)) %>% # for a sanity check
  filter(row_number() < n()*.66 & row_number() > n()*.33) # get the top 50% of rows from each group

cds_model_data_thirdthird <- cds_rolling2 %>%
  group_by(id) %>%
```

Table 6: (#tab:report CI ranges)Confidence interval range for Spanish/Quechua and child-directed speech estimation, by child, after annotating 80 clips and at annotation cut-off.

	Language		Child-directed speech	
Child ID	80-clip	Cut-off	80-clip	Cut-off
1032	0.16	0.13	0.21	0.17
1060	0.15	0.09	0.20	0.15
1075	0.13	0.11	0.17	0.17
1077	0.14	0.10	0.20	0.16
1081	0.08	0.11	0.16	0.13
179	0.21	0.15	0.21	0.15
198	0.21	0.19	0.20	0.19
199	0.18	0.13	0.21	0.15
261	0.21	0.18	0.21	0.18
267	0.20	0.18	0.16	0.15

```
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2)) %>% # for a sanity check
  filter(row_number() >= n()*.66) # get the top 50% of rows from each group

cds_model_data_full <- cds_rolling_complete2 %>% # all the data from the 'complete' annotation
  group_by(id) %>%
  arrange(annotation_num)

# fit models
cds_model_firstthird <- cds_model_data_firstthird %>%
  #filter(roll_sd_cds!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>%
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

cds_model_secondthird <- cds_model_data_secondthird %>%
  #filter(roll_sd_cds!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>%
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

cds_model_thirdthird <- cds_model_data_thirdthird %>%
  #filter(roll_sd_cds!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>%
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

cds_model_complete <- cds_model_data_full %>%
  #filter(roll_sd_cds!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>%
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()
```

```
# redo data to get the Bolivia and US corpora (more power for stats)
span_rolling_all <- lang_annon %>%
  filter(method=='random') %>%
  group_by(id) %>%
  arrange(file_name) %>%
  mutate(annotation_num = as.numeric(1:n())) %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(span_cts=recode(language, "Unsure"="0", "English/Quechua"="0", "Mixed"="0", "Spanish"="1")) %>%
  mutate(span_cts = as.numeric(span_cts)) %>%
  mutate(span_running_cts = as.numeric(cumsum(span_cts))) %>%
  mutate(roll_prop_span = span_running_cts / annotation_num,
         roll_mean_span = rollmean(roll_prop_span, k=10, fill = NA),
         roll_sd_span = rollapply(roll_prop_span, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
span_rolling_all2 <- span_rolling_all %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(span_running_cts, annotation_num, methods = 'wilson', conf.level = .95)
  merge(., span_rolling_all, by = c('id', 'annotation_num'))


# now start all over to get running variance for complete data from Bolivia and US
span_rolling_all_complete <- lang_annon %>%
  filter(method=='complete') %>%
  group_by(id) %>%
  arrange(file_name) %>%
  mutate(annotation_num = as.numeric(1:n())) %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(span_cts=recode(language, "Unsure"="0", "English/Quechua"="0", "Mixed"="0", "Spanish"="1")) %>%
  mutate(span_cts = as.numeric(span_cts)) %>%
  mutate(span_running_cts = as.numeric(cumsum(span_cts))) %>%
  mutate(roll_prop_span = span_running_cts / annotation_num,
         roll_mean_span = rollmean(roll_prop_span, k=10, fill = NA),
         roll_sd_span = rollapply(roll_prop_span, width=10, FUN=sd, fill=NA))

# running binomial confidence interval (wilson)
span_rolling_all_complete2 <- span_rolling_all_complete %>%
  group_by(id, annotation_num) %>% # group by id and sample size
  arrange(annotation_num) %>%
  summarize(cis = binom.confint(span_running_cts, annotation_num, methods = 'wilson', conf.level = .95)
  merge(., span_rolling_all_complete, by = c('id', 'annotation_num'))


# fit the spanish models
# make the datasets
span_model_data_firstthird <- span_rolling_all2 %>% # Bolivia and US; running variance over estimates m
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2))  %>% # for a sanity check
  filter(row_number() <= n()*.33)
```

```
span_model_data_secondthird <- span_rolling_all2 %>%
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2)) %>% # for a sanity check
  filter(row_number() < n()*.66 & row_number() > n()*.33)

  span_model_data_thirdthird <- span_rolling_all2 %>% # Bolivia and US; running variance over estimates
  group_by(id) %>%
  arrange(annotation_num) %>%
  mutate(halfrow = as.numeric(n()/2)) %>% # for a sanity check
  filter(row_number() >= n()*.66)

span_model_data_complete <- span_rolling_all_complete2 %>% # Bolivia & US; all the data from the 'all-d
  group_by(id) %>%
  arrange(annotation_num)

# fit the models
span_model_firstthird <- span_model_data_firstthird %>%
  #filter(roll_sd_span!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>% # get the variance
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

span_model_secondthird <- span_model_data_secondthird %>%
  #filter(roll_sd_span!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>% # get the variance
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

span_model_thirdthird <- span_model_data_thirdthird %>%
  #filter(roll_sd_span!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>% # get the variance
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()

span_model_complete<- span_model_data_complete %>%
  #filter(roll_sd_span!='NA') %>%
  mutate(ci_range = cis$upper - cis$lower) %>% # get the variance
  lmer(ci_range~annotation_num + (1|id), data = .) %>%
  summary()
```