Landmark analyses

Last updated: 2018-10-16

Contents

Setup

Load libraries

```
library(tidyverse)
library(broom)
library(knitr)
library(kableExtra)
library(lme4)
library(merTools)
library(patchwork)
```

Load data

```
learners <- read_csv("./data/landmarks_stress_la_lb_ss.csv") # learner data
heritage <- read_csv("./data/landmarks_stress_la_hs_ss.csv") # heritage data
wm_df_learners <- read_csv("./data/wm.csv")                   # wm learners
wm_df_heritage <- read_csv("./data/wm_all.csv")               # wm heritage
pstm_df <- read_csv("./data/dur_stress_background_info.csv") # phon memory
verb_freq_df <- read_csv("./data/verb_freq.csv")             # verb freq
phon_freq <- read_csv("./data/phonotactic_frequency.csv")    # phonotactic freq
```

Late learners and native controls

Do they predict above chance?

The data analyzed using a linear model with intercept removed. This makes each parameter estimate a two-sided test of independence ($H_a \neq 0$). In order to make this test one-sided ($H_a > 0$) we will take the t-values from the model and calculate the associated probability from the t-distribution for a one-sided test using the model degrees of freedom. In R this can be done with the following function:

```
pt(t_values, mod_df, lower = FALSE)
```

The p-values from the model will now be one-sided tests that the mean difference is greater than 0. Next, we need to put the target fixations (dependent variable) on the same scale. As is, chance = 50%, thus everything will be significant because target fixations are on average at 50% as a minimum. To get around this issue we can subtract 0.5 from each participants' mean target fixation at each landmark and test to see if that value is greater than 0.

For example, if at the target word onset you are fixating on the target 50% of the time (i.e., at chance), then when we subtract 0.5 from 0.5, we get 0. 0 is not greater than 0, so it wouldn't be significant. We will conduct this test for each group, at each landmark. Then we will add the 0.5 back on to the model estimates and the confidence intervals for plotting purposes.

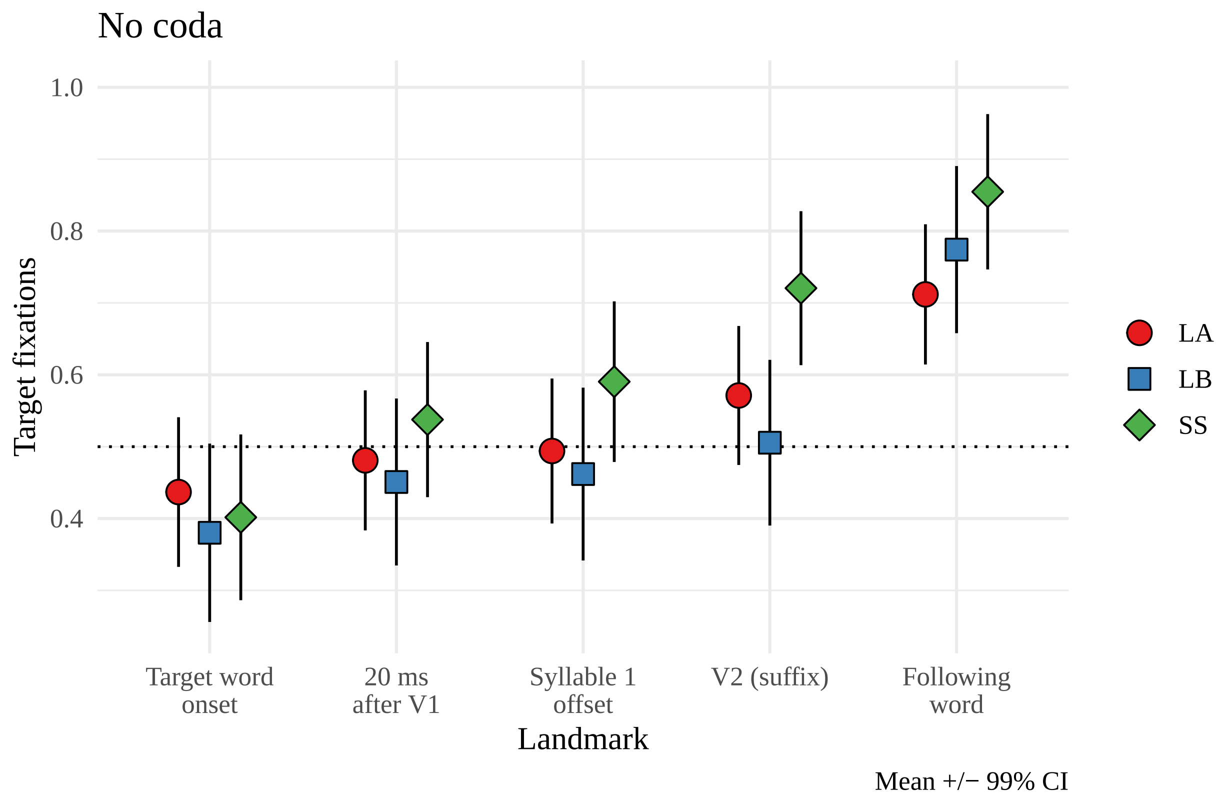
```
# Model degrees of freedom
learner_mod_df <- 65

learner_mods <- learners %>%
  filter(., !(landmark %in% c('start_sentence', 'word2_c1v1',
                              'end_sentence'))) %>%
  group_by(., participant, group, coda, landmark) %>%
  summarize(., target_fix = mean(targetProp)) %>%
  ungroup(.) %>%
  group_by(., landmark, coda) %>%
  do(tidy(lm(I(target_fix - 0.5) ~ -1 + group, data = .), conf.int = T,
          conf.level = 0.99)) %>%
  mutate(., p_adj = pt(statistic, learner_mod_df, lower = F),
            p_adj = formatC(p_adj, digits = 7, format = "f"),
            sig = if_else(p_adj < 0.05, true = "*", false = " ")) %>%
  ungroup(.) %>%
  mutate(., landmark = fct_relevel(landmark,
                                   'word3_c1v1', 'word3_20msafterv1',
                                   'word3_c2', 'word3_c3', 'word3_suffix')) %>%
  arrange(., coda, landmark)
```

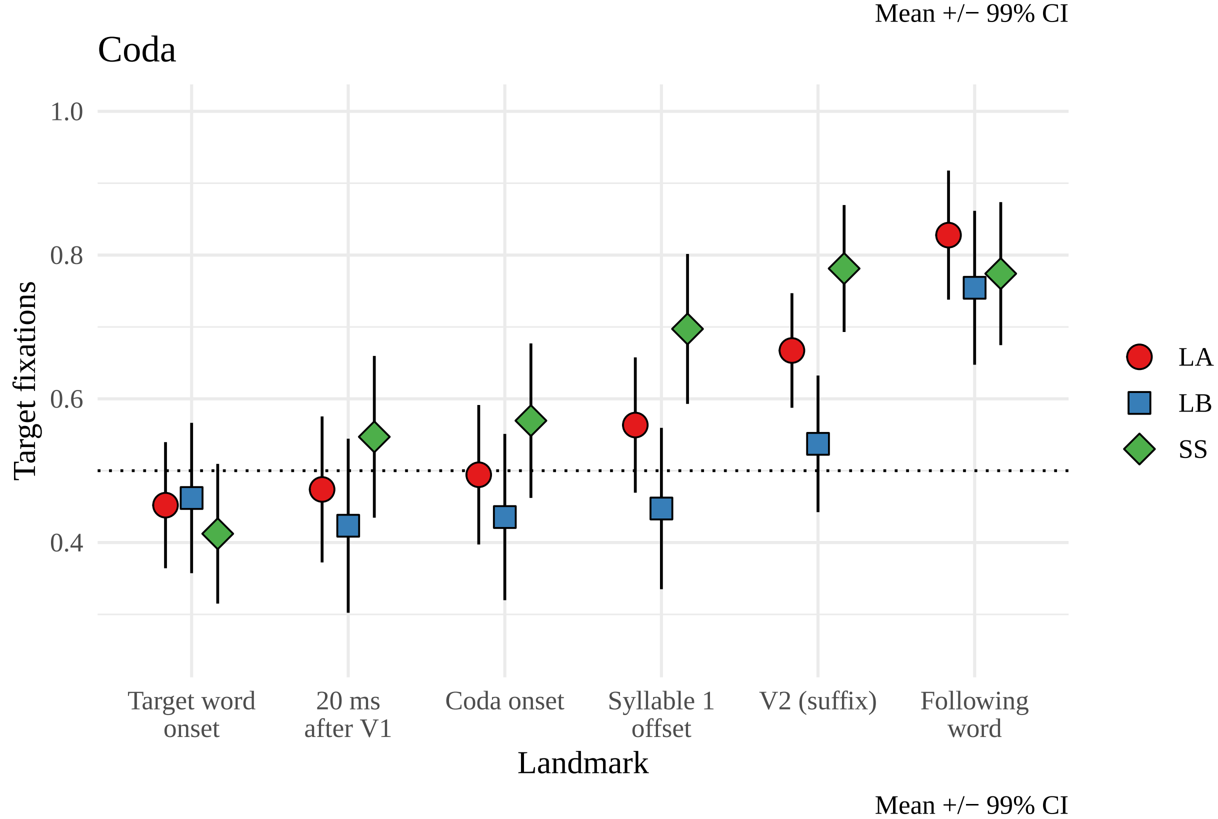
Table 1: Model output

landmark	term	estimate	std.error	statistic	conf.low	conf.high	p_adj	sig
No-coda targets								
	la	-0.06	0.04	-1.61	-0.17	0.04	0.9438315	
word3_c1v1	lb	-0.12	0.05	-2.56	-0.24	0.00	0.9936311	
	ss	-0.10	0.04	-2.26	-0.21	0.02	0.9864650	
	la	-0.02	0.04	-0.52	-0.12	0.08	0.6971842	
word3_20msafterv1	lb	-0.05	0.04	-1.12	-0.17	0.07	0.8671037	
	ss	0.04	0.04	0.93	-0.07	0.15	0.1787654	
	la	-0.01	0.04	-0.16	-0.11	0.09	0.5624449	
word3_c2	lb	-0.04	0.05	-0.84	-0.16	0.08	0.7982615	
	ss	0.09	0.04	2.15	-0.02	0.20	0.0176951	*
	la	0.07	0.04	1.96	-0.03	0.17	0.0274149	*
word3_suffix	lb	0.01	0.04	0.13	-0.11	0.12	0.4496902	
	ss	0.22	0.04	5.46	0.11	0.33	0.0000004	*
	la	0.21	0.04	5.76	0.11	0.31	0.0000001	*
word4_c1v1	lb	0.27	0.04	6.26	0.16	0.39	0.0000000	*
	ss	0.35	0.04	8.71	0.25	0.46	0.0000000	*
Coda targets								
	la	-0.05	0.03	-1.45	-0.14	0.04	0.9242526	
word3_c1v1	lb	-0.04	0.04	-0.96	-0.14	0.07	0.8308077	
	ss	-0.09	0.04	-2.40	-0.18	0.01	0.9902493	
	la	-0.03	0.04	-0.68	-0.13	0.08	0.7508646	
word3_20msafterv1	lb	-0.08	0.05	-1.68	-0.20	0.04	0.9510029	
	ss	0.05	0.04	1.11	-0.07	0.16	0.1353062	
	la	-0.01	0.04	-0.15	-0.10	0.09	0.5610294	
word3_c2	lb	-0.06	0.04	-1.48	-0.18	0.05	0.9281215	
	ss	0.07	0.04	1.72	-0.04	0.18	0.0452329	*
	la	0.06	0.04	1.79	-0.03	0.16	0.0390471	*
word3_c3	lb	-0.05	0.04	-1.24	-0.16	0.06	0.8910685	
	ss	0.20	0.04	5.02	0.09	0.30	0.0000022	*
	la	0.17	0.03	5.57	0.09	0.25	0.0000003	*
word3_suffix	lb	0.04	0.04	1.04	-0.06	0.13	0.1502988	
	ss	0.28	0.03	8.45	0.19	0.37	0.0000000	*
	la	0.33	0.03	9.68	0.24	0.42	0.0000000	*
word4_c1v1	lb	0.25	0.04	6.31	0.15	0.36	0.0000000	*
	ss	0.27	0.04	7.31	0.17	0.37	0.0000000	*

Note:
Parameter estimates show average target fixation minus 0.5.
P-values represent one-sided t-tests.
word3_c2 represents the 2nd syllable onset for no-coda targets and the coda onset for coda targets.

Landmark plots





Are working memory, word frequency, or phonotactic frequency factors?

Note:

- Phonological short-term memory is analyzed seperately because we do not have data from all three groups.
- These analyses **do not** include late beginners because they did not predict above chance at the target syllable offset (or earlier). This makes model fitting much faster.

```
## Joining, by = "participant"
```

```
## Joining, by = "target"
```

```
## Joining, by = c("target", "coda")
```

```
## Joining, by = c("participant", "group")
```

```
## Warning: Column `group` joining character vector and factor, coercing into
## character vector
```

```
## Joining, by = "target"
```

```
## Joining, by = c("target", "coda")
```

First check for homogeneity of variance for working memory.

```r
wm_df %>%
  separate(., participant, into = c('group', 'trash'), sep = 2, remove = F) %>%
  bartlett.test(wm ~ group, data = .)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  wm by group
## Bartlett's K-squared = 2.2443, df = 2, p-value = 0.3256
```

Looks good.

Now we analyze items without and with codas:

```
## Data: no_coda
## Models:
## nocoda_mod_null: cbind(targetCount, distractorCount) ~ -1 + (1 + wm_c | participant) +
## nocoda_mod_null:     (1 + wm_c + freq_sc + phon_prob_sc | target)
## nocoda_mod_wm: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## nocoda_mod_wm:     (1 + wm_c + freq_sc + phon_prob_sc | target) + group:wm_c -
## nocoda_mod_wm:     1
##                 Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## nocoda_mod_null 13 3805.2 3857.4 -1889.6   3779.2
## nocoda_mod_wm   16 3808.9 3873.1 -1888.5   3776.9 2.3147      3     0.5097
```

```
## Data: no_coda
## Models:
## nocoda_mod_null: cbind(targetCount, distractorCount) ~ -1 + (1 + wm_c | participant) +
## nocoda_mod_null:     (1 + wm_c + freq_sc + phon_prob_sc | target)
## nocoda_mod_freq: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## nocoda_mod_freq:     (1 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc -
## nocoda_mod_freq:     1
##                 Df    AIC    BIC  logLik deviance Chisq Chi Df Pr(>Chisq)
## nocoda_mod_null 13 3805.2 3857.4 -1889.6   3779.2
## nocoda_mod_freq 16 3743.1 3807.2 -1855.5   3711.1 68.16      3  1.057e-14
##
```

```
## nocoda_mod_null
## nocoda_mod_freq ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Data: no_coda
## Models:
## nocoda_mod_freq: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## nocoda_mod_freq:     (1 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc -
## nocoda_mod_freq:     1
## nocoda_mod_phon_prob: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## nocoda_mod_phon_prob:     (1 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc +
## nocoda_mod_phon_prob:     group:phon_prob_sc - 1
##                      Df    AIC    BIC  logLik deviance  Chisq Chi Df
## nocoda_mod_freq      16 3743.1 3807.2 -1855.5   3711.1
## nocoda_mod_phon_prob 19 3579.9 3656.1 -1771.0   3541.9 169.18      3
##                      Pr(>Chisq)
## nocoda_mod_freq
## nocoda_mod_phon_prob  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
##     (1 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc +
##     group:phon_prob_sc - 1
##    Data: no_coda
## Control:
## glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 3e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   3579.9   3656.1  -1771.0   3541.9      388
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -15.4161  -1.6806   0.0795   1.7696  12.1506
##
## Random effects:
##  Groups      Name         Variance Std.Dev. Corr
##  participant (Intercept)  6.1709   2.4841
##              wm_c         1.2387   1.1130   0.07
##  target      (Intercept)  3.1922   1.7867
##              wm_c         0.2852   0.5341    1.00
##              freq_sc      3.4237   1.8503    1.00  1.00
##              phon_prob_sc 2.0501   1.4318   -1.00 -1.00 -1.00
## Number of obs: 407, groups:  participant, 60; target, 8
##
## Fixed effects:
##                      Estimate Std. Error z value Pr(>|z|)
## groupss:freq_sc      -1.59469    0.40532  -3.934 8.34e-05 ***
## grouphs:freq_sc      -0.47730    0.42013  -1.136    0.256
## groupla:freq_sc      -0.09599    0.39645  -0.242    0.809
## groupss:phon_prob_sc  2.03925    0.39478   5.165 2.40e-07 ***
```

```
## groupHs:phon_prob_sc  1.58342    0.39124   4.047 5.18e-05 ***
## groupla:phon_prob_sc -0.22547    0.36490  -0.618    0.537
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             grps:_ grph:_ grpl:_ grps:__ grph:__
## grphs:frq_s  0.843
## grpl:frq_sc  0.873  0.827
## grpss:phn__ -0.906 -0.740 -0.755
## grphs:phn__ -0.804 -0.886 -0.742  0.855
## grpl:phn_p_ -0.839 -0.761 -0.888  0.875   0.828

## Data: coda
## Models:
## coda_mod_null: cbind(targetCount, distractorCount) ~ -1 + (1 + wm_c | participant) +
## coda_mod_null:     (0 + wm_c + freq_sc + phon_prob_sc | target)
## coda_mod_wm: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## coda_mod_wm:     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:wm_c -
## coda_mod_wm:     1
##               Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## coda_mod_null  9 4960.7 4998.5 -2471.3   4942.7
## coda_mod_wm   12 4965.0 5015.4 -2470.5   4941.0 1.7216      3     0.6321

## Data: coda
## Models:
## coda_mod_wm: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## coda_mod_wm:     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:wm_c -
## coda_mod_wm:     1
## coda_mod_freq: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## coda_mod_freq:     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc -
## coda_mod_freq:     1
##               Df  AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## coda_mod_wm   12 4965 5015.4 -2470.5     4941
## coda_mod_freq 12 4871 4921.5 -2423.5     4847 93.919      0  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Data: coda
## Models:
## coda_mod_freq: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## coda_mod_freq:     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc -
## coda_mod_freq:     1
## coda_mod_phon_prob: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
## coda_mod_phon_prob:     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc +
## coda_mod_phon_prob:     group:phon_prob_sc - 1
##                    Df    AIC    BIC  logLik deviance  Chisq Chi Df
## coda_mod_freq      12 4871.0 4921.5 -2423.5   4847.0
## coda_mod_phon_prob 15 4821.6 4884.7 -2395.8   4791.6 55.424      3
##                    Pr(>Chisq)
## coda_mod_freq
## coda_mod_phon_prob  5.576e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
```

```
## control$checkConv, : unable to evaluate scaled gradient

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge: degenerate Hessian with 1
## negative eigenvalues

## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: cbind(targetCount, distractorCount) ~ (1 + wm_c | participant) +
##     (0 + wm_c + freq_sc + phon_prob_sc | target) + group:freq_sc +
##     group:phon_prob_sc - 1
##    Data: coda
## Control:
## glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 3e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   4821.6   4884.7  -2395.8   4791.6      481
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -8.9618 -2.1269  0.5826  1.8559 10.4641
##
## Random effects:
##  Groups      Name         Variance Std.Dev. Corr
##  participant (Intercept)  2.900553 1.70310
##              wm_c         0.007755 0.08806  -1.00
##  target      wm_c         0.049209 0.22183
##              freq_sc      0.770465 0.87776   0.45
##              phon_prob_sc 0.619668 0.78719   0.48 -0.56
## Number of obs: 496, groups:  participant, 60; target, 10
##
## Fixed effects:
##                       Estimate Std. Error z value Pr(>|z|)
## groupss:freq_sc       0.005428   0.486499   0.011 0.991098
## groupphs:freq_sc     -0.364999   0.492130  -0.742 0.458286
## groupla:freq_sc      -0.927648   0.491681  -1.887 0.059203 .
## groupss:phon_prob_sc  0.519260   0.359096   1.446 0.148171
## groupphs:phon_prob_sc  1.253324   0.367327   3.412 0.000645 ***
## groupla:phon_prob_sc  0.614744   0.362422   1.696 0.089846 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             grps:_ grph:_ grpl:_ grps:__ grph:__
## grphs:frq_s  0.973
## grpl:frq_sc  0.980  0.973
## grpss:phn__ -0.159 -0.152 -0.145
## grphs:phn__ -0.151 -0.151 -0.150  0.954
## grpl:phn_p_ -0.161 -0.165 -0.162  0.966   0.958
```

Phonological short-term memory

Note: this is separate from main analysis because there is not data from SS group.

```
## Warning: Column `group` joining factors with different levels, coercing to
```

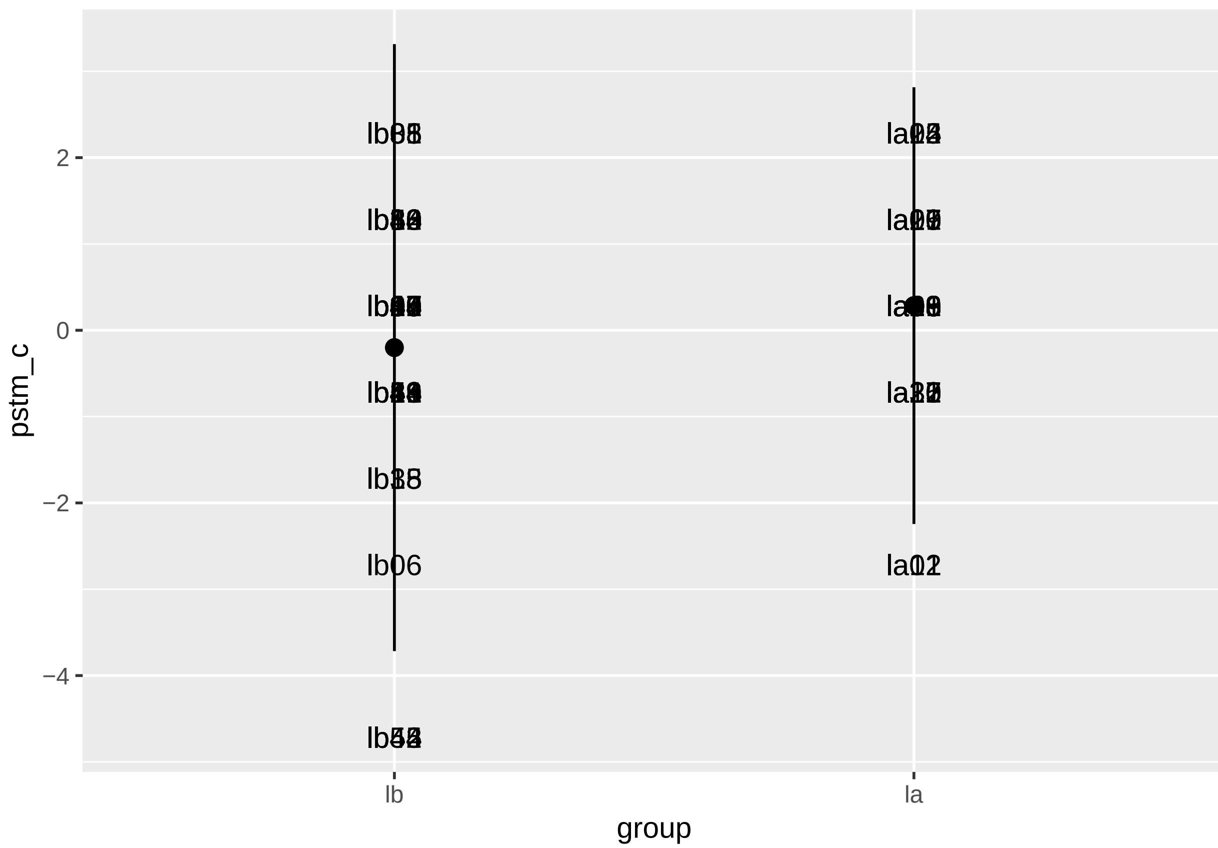
```
## character vector
```

First check for homogeneity of variance.

```
pstm_learners_clean %>%
  bartlett.test(pstm_c ~ group, data = .)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  pstm_c by group
## Bartlett's K-squared = 2.9727, df = 1, p-value = 0.08468
```

```
pstm_learners_clean %>%
  na.omit(.) %>%
  ggplot(., aes(x = group, y = pstm_c, label = participant)) +
    geom_text() +
    stat_summary(fun.data = mean_sdl, geom = 'pointrange')
```



Groups look ok. Might have to take some out.

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## cbind(targetCount, distractorCount) ~ (1 + pstm_c | participant) +
##     (1 + pstm_c | target) + group:pstm_c - 1
##    Data: lb_la_pstm_no_coda
## Control: glmerControl(optimizer = "bobyqa")
##
```

```
##      AIC      BIC   logLik deviance df.resid
##   2866.2   2894.8  -1425.1   2850.2      254
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -8.440 -2.270  0.000  2.271  5.608
##
## Random effects:
##  Groups      Name        Variance Std.Dev. Corr
##  participant (Intercept) 0.9050   0.9513
##              pstm_c      0.1122   0.3350   0.64
##  target      (Intercept) 0.2638   0.5136
##              pstm_c      0.2603   0.5102   -0.56
## Number of obs: 262, groups:  participant, 42; target, 13
##
## Fixed effects:
##                Estimate Std. Error z value Pr(>|z|)
## groupla:pstm_c   0.1915     0.2239   0.856    0.392
## grouplb:pstm_c   0.3859     0.2958   1.305    0.192
##
## Correlation of Fixed Effects:
##             grpl:p_
## grplb:pstm_ 0.267
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## cbind(targetCount, distractorCount) ~ (1 + pstm_c | participant) +
##     (1 + pstm_c | target) + group:pstm_c - 1
##    Data: lb_la_pstm_coda
## Control: glmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   4183.2   4214.7  -2083.6   4167.2      374
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -8.6595 -2.3644  0.6149  2.3642  7.4094
##
## Random effects:
##  Groups      Name        Variance Std.Dev. Corr
##  participant (Intercept) 0.7548   0.8688
##              pstm_c      0.4193   0.6476   -0.21
##  target      (Intercept) 0.4480   0.6693
##              pstm_c      0.4644   0.6815   -0.42
## Number of obs: 382, groups:  participant, 42; target, 19
##
## Fixed effects:
##                Estimate Std. Error z value Pr(>|z|)
## groupla:pstm_c   0.1981     0.3037   0.652    0.514
## grouplb:pstm_c  -0.1351     0.3518  -0.384    0.701
##
## Correlation of Fixed Effects:
```

```
##              grpl:p_
## grplb:pstm_ 0.152
```

Learners summary

Without coda:

- The native speakers fixate on targets above chance at the offset of the first syllable.
- Advanced learners fixate on targets as well by the time they have heard the target suffix.
- All three groups fixate on targets above chance by the following word.
- No effect of working memory for any groups.
- Effects of word frequency (negative) but only for native monolinguals (?)
- Effects of phonotactic probability (postive).
- No effect of phonological short-term memory for learners.

With coda:

- Native speakers fixate on targets above chance at the onset of the coda.
- Advanced learners fixate on targets above chance by the offset of the target syllable.
- All three groups fixate on targets above chance by the following word.
- No effect of working memory.
- No effect of frequency.. might be negatively correlated with target fixations for the advanced learners (p = 0.59). Lots of variability. We need to think about this.
- No effect of phonotactic probability. Positive trend for LA (p = 0.89)
- No effect of phonological short-term memory.

NOCODA

	Estimate	Std. Error	z value	p	
groupss:freq_sc	-1.59469	0.40532	-3.934	8.34e-05	***
groupla:freq_sc	-0.09599	0.39645	-0.242	0.809	
groupss:phon_prob_sc	2.03925	0.39478	5.165	2.40e-07	***
groupla:phon_prob_sc	-0.22547	0.36490	-0.618	0.537	

CODA

	Estimate	Std. Error	z value	P	
groupss:freq_sc	0.005428	0.486499	0.011	0.991098	
groupla:freq_sc	-0.927648	0.491681	-1.887	0.059203	.
groupss:phon_prob_sc	0.519260	0.359096	1.446	0.148171	
groupla:phon_prob_sc	0.614744	0.362422	1.696	0.089846	.

To think about: we might have an explanation for the SS/LA difference in coda vs. no coda targets: SS use phonotactic probability/frequency in the no coda context (which helps because apparently it is the harder context of the two). LA do not. That said, there is not effect in the coda context for natives, but that could be because they don't need it there… they have more time, cues, something?

Late vs. early bilinguals and native (monolingual) controls

Do they predict above chance?

Same analysis as previously described.

```
# Model degrees of freedom
heritage_mod_df <- 72

heritage_mods <- heritage %>%
  filter(., !(landmark %in% c('start_sentence', 'word2_c1v1',
                              'end_sentence'))) %>%
  group_by(., participant, group, coda, landmark) %>%
  summarize(., target_fix = mean(targetProp)) %>%
  ungroup(.) %>%
  group_by(., landmark, coda) %>%
  do(tidy(lm(I(target_fix - 0.5) ~ -1 + group, data = .), conf.int = T,
          conf.level = 0.99)) %>%
  mutate(., p_adj = pt(statistic, heritage_mod_df, lower = F),
            p_adj = formatC(p_adj, digits = 7, format = "f"),
            sig = if_else(p_adj < 0.05, true = "*", false = " ")) %>%
  ungroup(.) %>%
  mutate(., landmark = fct_relevel(landmark,
                                   'word3_c1v1', 'word3_20msafterv1',
                                   'word3_c2', 'word3_c3', 'word3_suffix')) %>%
  arrange(., coda, landmark)
```

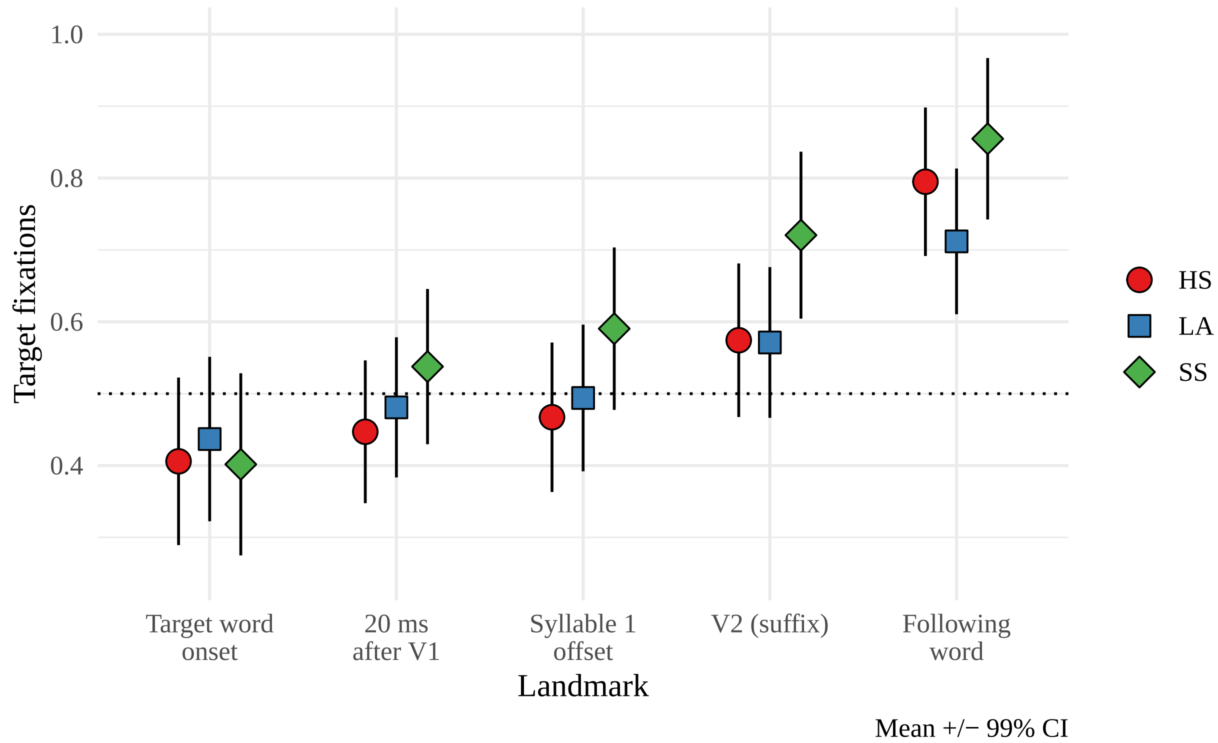
Table 4: Model output

landmark	term	estimate	std.error	statistic	conf.low	conf.high	p_adj	sig
No-coda targets								
	hs	-0.09	0.04	-2.14	-0.21	0.02	0.9819752	
word3_c1v1	la	-0.06	0.04	-1.46	-0.18	0.05	0.9257015	
	ss	-0.10	0.05	-2.05	-0.23	0.03	0.9780822	
	hs	-0.05	0.04	-1.41	-0.15	0.05	0.9188499	
word3_20msafterv1	la	-0.02	0.04	-0.52	-0.12	0.08	0.6965797	
	ss	0.04	0.04	0.92	-0.07	0.15	0.1795164	
	hs	-0.03	0.04	-0.83	-0.14	0.07	0.7965654	
word3_c2	la	-0.01	0.04	-0.16	-0.11	0.10	0.5615723	
	ss	0.09	0.04	2.12	-0.02	0.20	0.0188317	*
	hs	0.07	0.04	1.84	-0.03	0.18	0.0348610	*
word3_suffix	la	0.07	0.04	1.80	-0.03	0.18	0.0381995	*
	ss	0.22	0.04	5.02	0.10	0.34	0.0000018	*
	hs	0.29	0.04	7.55	0.19	0.40	0.0000000	*
word4_c1v1	la	0.21	0.04	5.53	0.11	0.31	0.0000002	*
	ss	0.35	0.04	8.36	0.24	0.47	0.0000000	*
Coda targets								
	hs	-0.04	0.03	-1.46	-0.12	0.04	0.9259936	
word3_c1v1	la	-0.05	0.03	-1.61	-0.13	0.03	0.9441519	
	ss	-0.09	0.03	-2.66	-0.18	0.00	0.9951562	
	hs	-0.04	0.04	-0.99	-0.13	0.06	0.8372537	
word3_20msafterv1	la	-0.03	0.04	-0.72	-0.12	0.07	0.7627565	
	ss	0.05	0.04	1.17	-0.06	0.15	0.1223290	
	hs	-0.04	0.04	-0.99	-0.14	0.06	0.8383116	
word3_c2	la	-0.01	0.04	-0.15	-0.10	0.09	0.5607754	
	ss	0.07	0.04	1.71	-0.04	0.18	0.0457271	*
	hs	0.08	0.04	1.96	-0.03	0.19	0.0271484	*
word3_c3	la	0.07	0.04	1.79	-0.03	0.18	0.0390024	*
	ss	0.20	0.04	4.56	0.08	0.32	0.0000104	*
	hs	0.21	0.04	5.71	0.11	0.30	0.0000001	*
word3_suffix	la	0.17	0.04	4.72	0.07	0.26	0.0000057	*
	ss	0.28	0.04	7.16	0.18	0.39	0.0000000	*
	hs	0.30	0.04	8.48	0.21	0.40	0.0000000	*
word4_c1v1	la	0.33	0.04	9.35	0.23	0.42	0.0000000	*
	ss	0.27	0.04	7.06	0.17	0.38	0.0000000	*

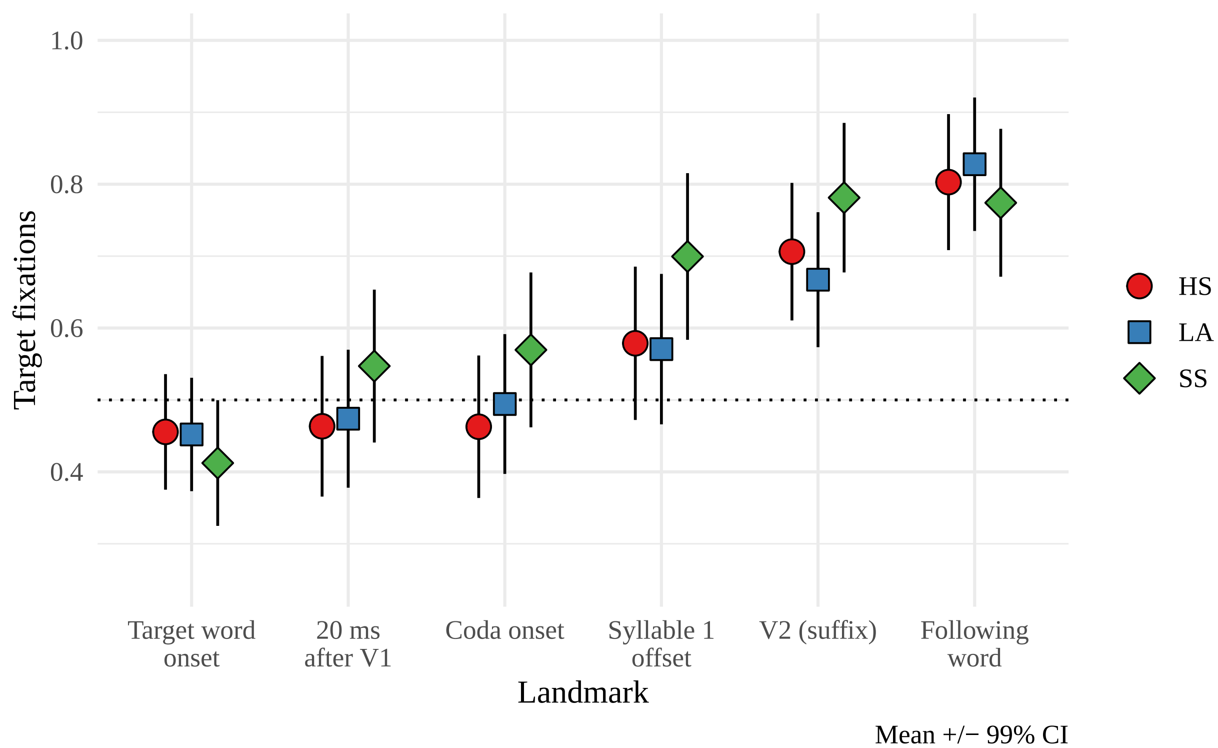
Note:
Parameter estimates show average target fixation minus 0.5.
P-values represent one-sided t-tests.
word3_c2 represents the 2nd syllable onset for no-coda targets and the coda onset for coda targets.

Landmark plots

No coda



Coda



Are working memory, frequency, or phonological frequency factors?

Check for homogeneity of variance.

```
wm_df_heritage %>%
  filter(., group %in% c("LA", "HS", "S")) %>%
  bartlett.test(WM ~ group, data = .)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  WM by group
## Bartlett's K-squared = 1.9167, df = 2, p-value = 0.3835
```

Looks good.

Phonological short-term memory

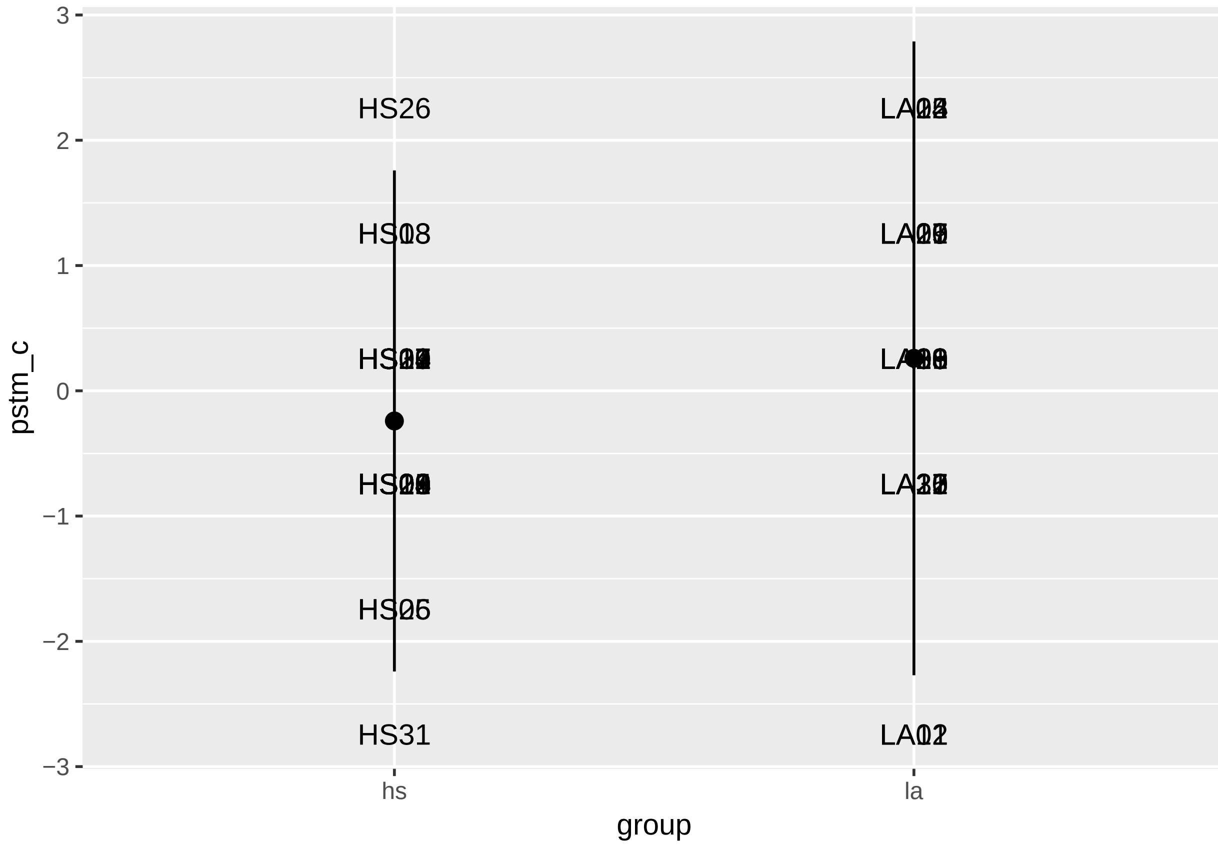
```
## Joining, by = c("participant", "group")
```

First check for homogeneity of variance.

```
pstm_clean %>%
  bartlett.test(pstm_c ~ group, data = .)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  pstm_c by group
## Bartlett's K-squared = 1.4022, df = 1, p-value = 0.2364
```

```
pstm_clean %>%
  na.omit(.) %>%
  ggplot(., aes(x = group, y = pstm_c, label = participant)) +
    geom_text() +
    stat_summary(fun.data = mean_sdl, geom = 'pointrange')
```



Groups look good.

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## cbind(targetCount, distractorCount) ~ (1 + pstm_c | participant) +
##     (1 + pstm_c | target) + group:pstm_c - 1
##    Data: hs_la_pstm_no_coda
## Control: glmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   3198.2   3228.3  -1591.1   3182.2      308
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -16.156  -2.057   0.000   2.123   8.412
##
## Random effects:
##  Groups      Name        Variance Std.Dev. Corr
##  participant (Intercept) 0.8075   0.8986
##              pstm_c      0.1939   0.4404   -0.33
##  target      (Intercept) 0.8955   0.9463
##              pstm_c      0.7035   0.8387   -0.33
## Number of obs: 316, groups:  participant, 50; target, 13
##
## Fixed effects:
```

```
##                Estimate Std. Error z value Pr(>|z|)
## groupla:pstm_c   0.3122     0.2925   1.067    0.286
## groupհs:pstm_c  -0.5472     0.4185  -1.307    0.191
##
## Correlation of Fixed Effects:
##             grpl:_
## grphs:pstm_ 0.331
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## cbind(targetCount, distractorCount) ~ (1 + pstm_c | participant) +
##     (1 + pstm_c | target) + group:pstm_c - 1
##    Data: hs_la_pstm_coda
## Control: glmerControl(optimizer = "bobyqa")
##
##      AIC      BIC   logLik deviance df.resid
##   4709.6   4742.6  -2346.8   4693.6      447
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -14.305  -2.192   0.936   2.020   6.676
##
## Random effects:
##  Groups      Name        Variance Std.Dev. Corr
##  participant (Intercept) 0.9991   0.9996
##              pstm_c      0.2931   0.5413   -0.06
##  target      (Intercept) 0.2922   0.5406
##              pstm_c      0.5636   0.7507   -0.19
## Number of obs: 455, groups:  participant, 50; target, 19
##
## Fixed effects:
##                Estimate Std. Error z value Pr(>|z|)
## groupհs:pstm_c  -0.1678     0.3536  -0.475    0.635
## groupla:pstm_c   0.1483     0.3294   0.450    0.653
##
## Correlation of Fixed Effects:
##             grph:_
## grpl:pstm_c 0.329
```

Early/late bilinguals summary

Learners summary

(note: results focus on heritage because SS and LA are discussed above)

Without coda:

- Only monolinguals fixate on target above chance at the offset of the first syllable.
- All three groups fixate on target above chance by the suffix.
- No effect of working memory (a lot of variability for heritage).
- No ffect of freq
- Effect of phonotactic probability (positive) for heritage.
- No effect of PSTM.

With coda:

- Only monolinguals fixate on targets above chance at the coda.
- All three groups fixate on targets above chance at the offset of the first syllable.
- No effect of working memory.
- No effect of freq.
- Effect (positive) of phonotactic probability for heritage.
- No effect of PSTM.

(repeat table for clarity)

NOCODA

	Estimate	Std. Error	z value	p	
groupss:freq_sc	-1.59469	0.40532	-3.934	8.34e-05	***
grouphs:freq_sc	-0.47730	0.42013	-1.136	0.256	
groupla:freq_sc	-0.09599	0.39645	-0.242	0.809	
groupss:phon_prob_sc	2.03925	0.39478	5.165	2.40e-07	***
grouphs:phon_prob_sc	1.58342	0.39124	4.047	5.18e-05	***
groupla:phon_prob_sc	-0.22547	0.36490	-0.618	0.537	

CODA

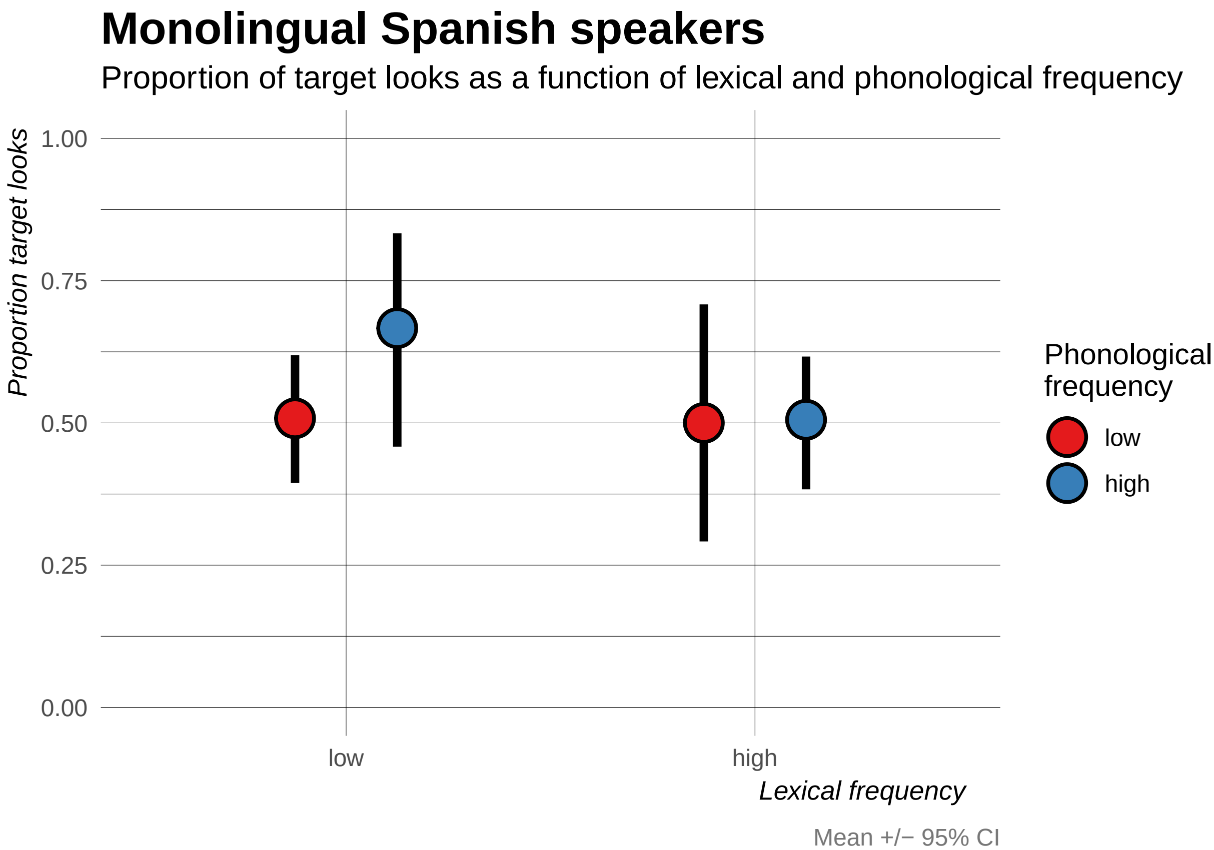
	Estimate	Std. Error	z value	P	
groupss:freq_sc	0.005428	0.486499	0.011	0.991098	
grouphs:freq_sc	-0.364999	0.492130	-0.742	0.458286	
groupla:freq_sc	-0.927648	0.491681	-1.887	0.059203	.
groupss:phon_prob_sc	0.519260	0.359096	1.446	0.148171	
grouphs:phon_prob_sc	1.253324	0.367327	3.412	0.000645	***
groupla:phon_prob_sc	0.614744	0.362422	1.696	0.089846	.

Monolingual lexical frequency/phonotactic frequency interaction

```
tidy(mono_mod_final) %>%
  slice(., 1:4) %>%
  mutate(., p.value = p.value %>% academicWriteR::round_pval(.)) %>%
  kable(., format = 'latex', booktabs = T, escape = T)
```

term	estimate	std.error	statistic	p.value	group
(Intercept)	62.02399	0.0004337	143009.53	0.001	fixed
freq_sc	-98.98400	0.0004337	-228227.66	0.001	fixed
phon_prob_sc	123.79852	0.0004337	285442.88	0.001	fixed
freq_sc:phon_prob_sc	26.42449	0.0004337	60926.95	0.001	fixed

```
#library(sjPlot)
#plot_model(mono_mod_final, type = "pred")
#
#library(ggeffects)
## dat is a data frame with marginal effects
#dat <- ggpredict(mono_mod_final, term = "freq_sc")
#plot(dat)
#
#dat <- ggpredict(mono_mod_final, term = "phon_prob_sc")
#plot(dat)
#
#mono_no_coda %>%
#  ggplot(., aes(x = phon_prob_sc, y = targetProp, color = freq_cat)) +
#    geom_jitter(height = 0.05, width = 0.5) +
#    geom_smooth(method = 'lm', fullrange = T, se = F)

mono_no_coda %>%
  mutate(., freq_cat = fct_relevel(freq_cat, "low"),
            phon_cat = fct_relevel(phon_cat, "low")) %>%
  ggplot(., aes(x = freq_cat, y = targetProp, fill = phon_cat,
                dodge = phon_cat)) +
    stat_summary(fun.data = mean_cl_boot, geom = "pointrange", pch = 21,
                 size = 1.5, position = position_dodge(0.5), color = 'black') +
    ylim(0, 1) +
    scale_fill_brewer(name = "Phonological\nfrequency", palette = "Set1") +
    labs(title = "Monolingual Spanish speakers",
         subtitle = "Proportion of target looks as a function of lexical and phonological frequency",
         caption = "Mean +/- 95% CI", y = "Proportion target looks",
         x = "Lexical frequency") +
    my_theme()
```



What it means:

At the offset of the first syllable (before the suffix), the monolingual Spanish speakers look at the target approx. 59% of the time. If we do a median split of the lexical and phonological frequencies (low, high), we see that the items with low lexical frequency, but high phonological frequency, are driving the "above chance" difference. In other words, if it wasn't for the high phonological frequency items, they likely would not be predicting above chance at this point in the time course.

What we need:

Another plot that shows this same data for all the landmarks.