Statistics for Linguistics

Session 5

Multiple Linear Regression

Example Data

- For the following illustrations we will use data collected in a study on

Compensatory Vowel Shortening in German[1]

- Stressed vowels are shortened depending on how many segments follow within the same word
- e.g. /a:/ in /**ma:**/ is longer than in /**ma:m**/
  /a:/ in /**ma:m**/ is longer than in /**ma:ms**/
  /a:/ in /**ma:ms**/ is longer than in /**ma:ms.la**/

[1]Schmitz et al. (2018)

Example Data

- For the following illustrations we will use data collected in a study on

Compensatory Vowel Shortening in German[1]

- Independent of shortening, open vowels should be shorter than mid vowels, which in turn should be shorter than closed vowels
- i.e. /**i:**, **u:**/ < /**e:**, **o:**/ < /**a:**/

[1]Schmitz et al. (2018)

Simple Linear Regression Formula

continuous
dependent variable

predictor variable

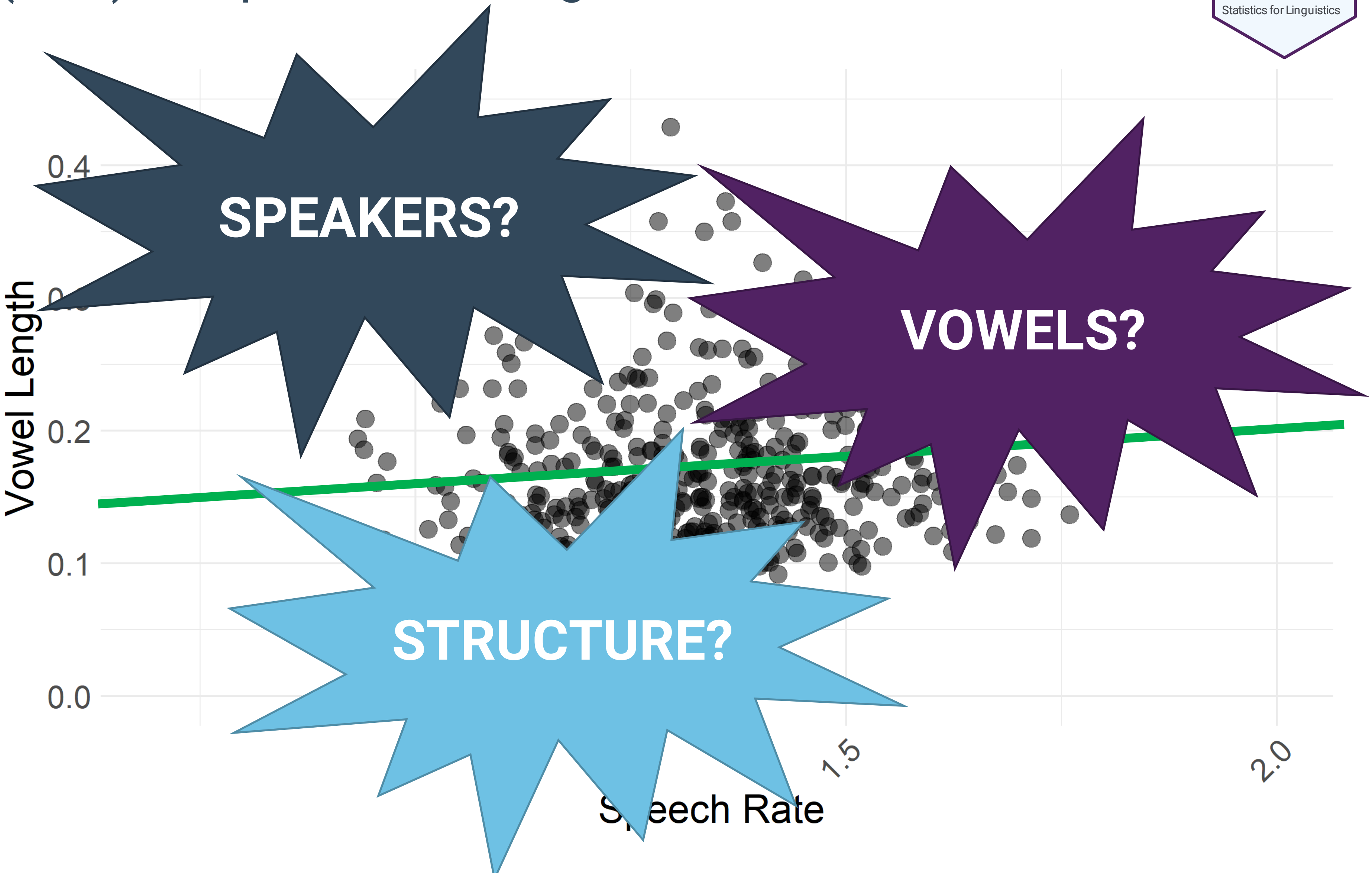
$$Y = \beta_1 + \beta_2 X + \epsilon$$

intercept

slope

error term /
residuals

(Too) Simple Linear Regression



Simple Linear Regression Formula

continuous
dependent variable

predictor variable

$$Y = \beta_1 + \beta_2 X + \epsilon$$

intercept

slope

error term /
residuals

Multiple Linear Regression Formula

continuous
dependent variable

predictor variable 2

predictor variable 1

predictor variable i

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_i X_i + \epsilon$$

intercept

slope
of variable 1

slope
of variable 2

slope
of variable i

error term /
residuals

Multiple Linear Regression in R

- More variables make the modelling procedure a little more time consuming
- Typical steps are
  1. Check dependent variable distribution
  2. Create a ‘full’ model
  3. Find the ‘best’ model
  4. Check assumptions
  5. Interpret the model

Step 1: Dependent Variable Distribution

- As seen earlier, we can check the distribution of a variable with a **Shapiro-Wilk Test**
- Our dependent variable, `duration`, is not normally distributed
- We therefore, again, use a **log-transformed** version of the duration variable, i.e. `durationLog`

Step 2: Full Model Creation

- Our dependent variable is `durationLog`
- Now, we need to consider the independent variables we wish to use in our model
- We wish to include the following independent variables:
  - `structure` i.e. coda structure
  - `vowel` i.e. vowel quality
  - `rate` i.e. speech rate
  - `number` i.e. slide number during experiment

Step 2: Full Model Creation

- Let's create the full model:

```
model = lm(durationLog ~ structure + vowel + rate +
                                number, data)
```

Step 3: Find Best Model

- In theory, we now should create models with all possible combinations of variables
- However, this is time consuming and error prone
- Luckily, R provides a function for this step

```
step(model)
```

Step 3: Find Best Model

```
> step(model)
```

Step 3: Find Best Model

```
> step(model)
Start:  AIC=-1167.31
durationLog ~ structure + vowel + rate + number
```

Akaike Information Criterion
The lower, the better the model fit

Step 3: Find Best Model

Akaike Information Criterion
The lower, the better the model fit

```
> step(model)
Start:  AIC=-1167.31
durationLog ~ structure + vowel + rate + number

            Df Sum of Sq    RSS     AIC
- number     1    0.0536 31.839 -1168.55
<none>                   31.786 -1167.31
- rate       1    0.8500 32.636 -1157.48
- vowel      4    3.4109 35.197 -1129.64
- structure  2   14.9708 46.756  -998.41
```

a model without number

a model without vowel

Step 3: Find Best Model

```
Step:  AIC=-1168.55
durationLog ~ structure + vowel + rate
```

best model found by the `step()` function and its AIC

```
              Df Sum of Sq    RSS      AIC
<none>                     31.839 -1168.55
- rate         1    0.8416 32.681 -1158.86
- vowel        4    3.4070 35.246 -1131.01
- structure    2   14.9881 46.827  -999.73
```

additional proof that further reduction is not improving model fit

Step 3: Find Best Model

best model found by the `step()` function and its call

```
Call:
lm(formula = durationLog ~ structure + vowel + rate, data = data)

Coefficients:
    (Intercept)     structureopen   structuresingle              vowele
        -1.5062            0.4340            0.1219             -0.1441
         voweli            vowelo            vowelu                rate
        -0.2374           -0.1229           -0.2365             -0.2532
```

model coefficients – we will take a closer look in step 5

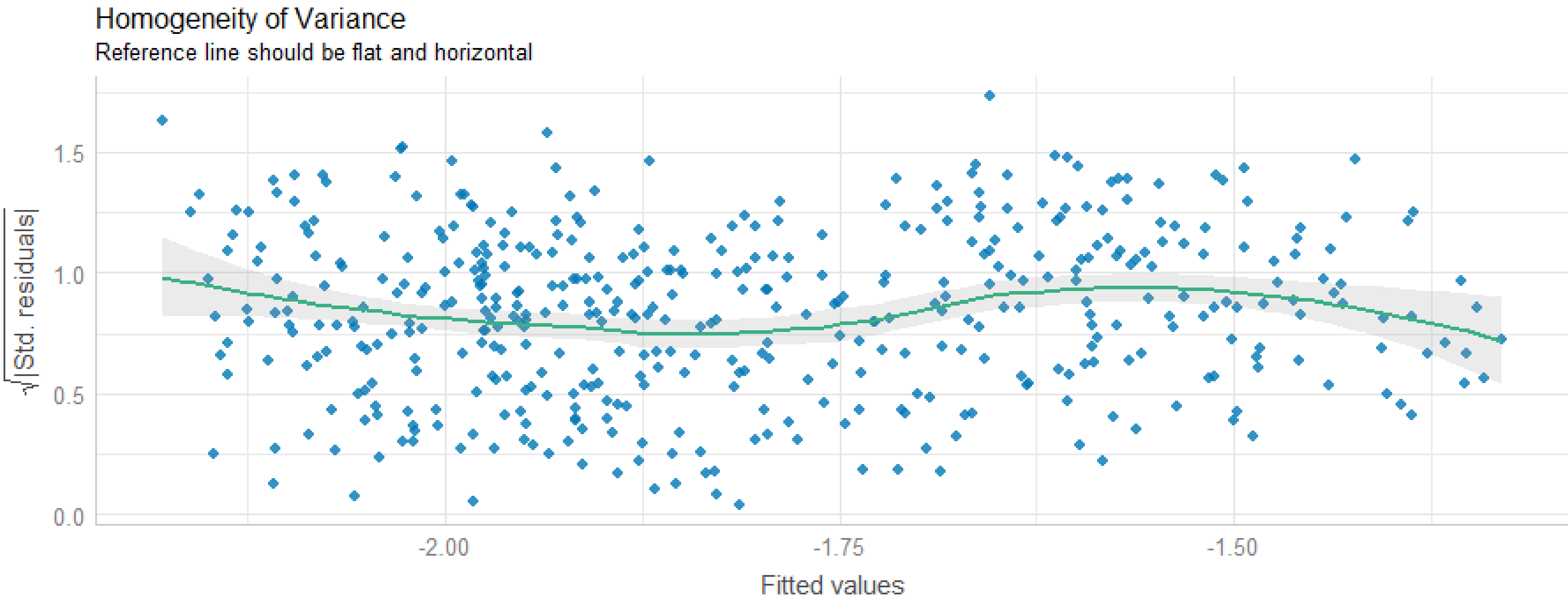
Step 4: Check Assumptions

- Multiple Linear Regression Models follow the same **assumptions** as Simple Linear Regression Models
  - Linearity
  - Homoscedasticity
  - Normality
  - Independence

Step 4: Check Assumptions

- **Linearity** Assumption:

  The relationship between X and the mean of Y is linear.

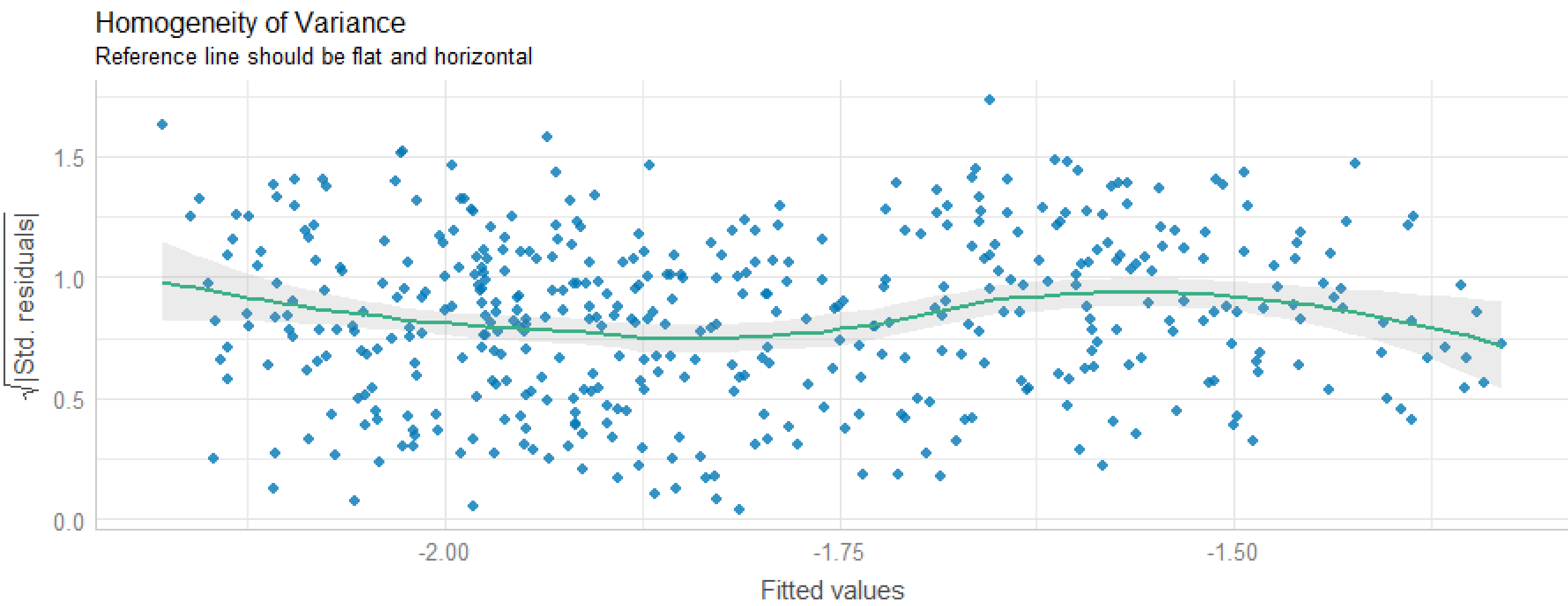


- The line should be horizontal and flat.

Step 4: Check Assumptions

- **Homoscedasticity** Assumption:

  The variance of residuals is the same for any value of X.

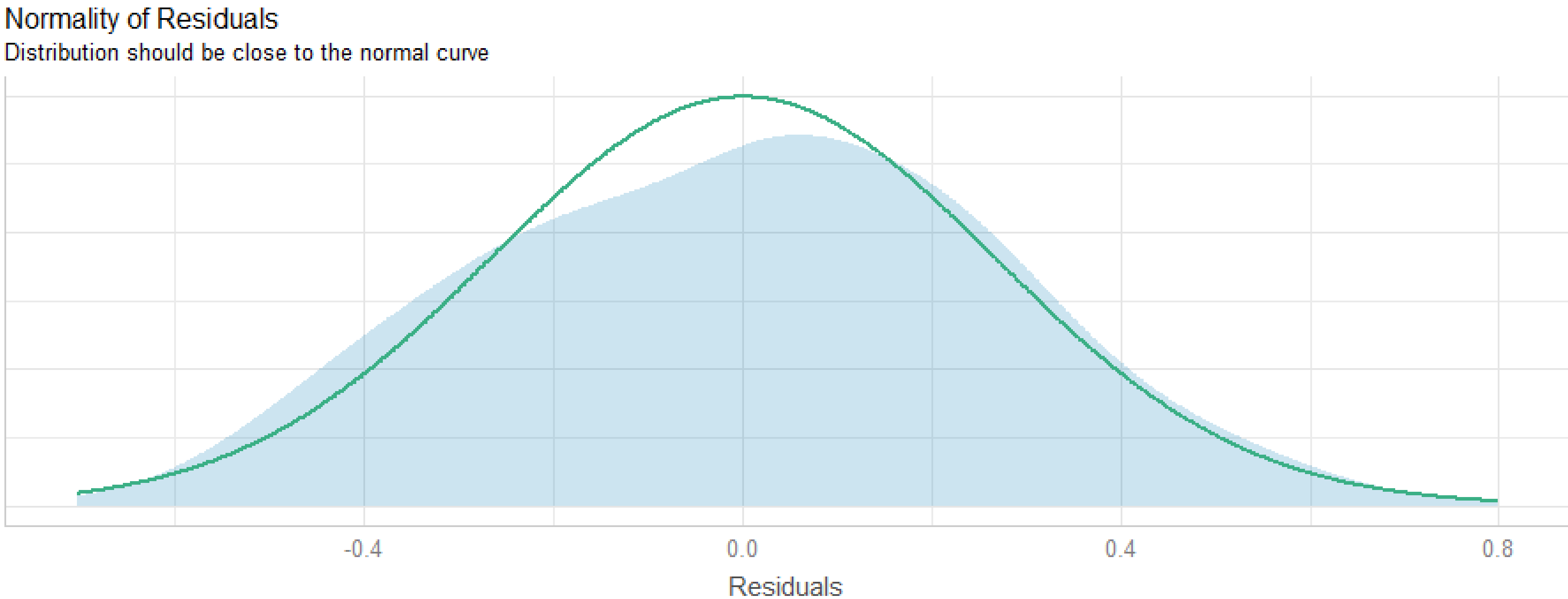


- Data should be spread equally around the line, with no obvious patterns visible.

Step 4: Check Assumptions

- **Normality** Assumption:

  For any fixed value of X, Y is normally distributed.



- The distribution of a linear model's residuals should follow a normal distribution.

Step 5: Interpretation

- In general, we are interested in two things
  1. the ***p*-values** of individual predictors
  2. the **effects** of the individual predictors

Step 5: Interpretation – *p*-Values

1. Using the `anova()` function, we can obtain ***p*-values**

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
structure	2	15.131	7.5654	104.4874	< 2.2e-16	***
vowel	4	3.507	0.8767	12.1079	2.41e-09	***
rate	1	0.842	0.8416	11.6241	0.0007112	***
Residuals	439	31.786	0.0724			

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

```
                  Estimate   Std. Error  t value  Pr(>|t|)
(Intercept)       -1.50620    0.10486   -14.364   < 2e-16 ***
structureopen      0.43395    0.03112    13.947   < 2e-16 ***
structuresingle    0.12186    0.03117     3.910  0.000107 ***
vowele            -0.14406    0.04033    -3.572  0.000393 ***
voweli            -0.23739    0.04035    -5.883  7.97e-09 ***
vowelo            -0.12292    0.04034    -3.048  0.002446 **
vowelu            -0.23653    0.04033    -5.864  8.87e-09 ***
rate              -0.25324    0.07425    -3.410  0.000708 ***
```

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.				***
vowele	-0.				***
voweli	-0.				***
vowelo	-0.				**
vowelu	-0.				***
rate	-0.				***

contains the 'baseline' levels of all factors, i.e.

structure:double
vowel:a

plus the 'starting point' of the numerical predictor(s)

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.[illegible]	[illegible]	[illegible]	[illegible]	***
vowele	-0.[illegible]	[illegible]	[illegible]	[illegible]	***
voweli	-0.[illegible]	[illegible]	[illegible]	[illegible]	***
vowelo	-0.12292	0.04034	-3.048	0.002446	**
vowelu	-0.23653	0.04033	-5.864	8.87e-09	***
rate	-0.25324	0.07425	-3.410	0.000708	***

structure:double + vowel:a + rate:start

estimated mean of `durationLog`

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

```
                  Estimate   Std. Error  t value   Pr(>|t|)
(Intercept)       -1.50620     0.10486  -14.364    < 2e-16 ***
structureopen      0.43395     0.031[illegible]   13.947    < 2e-16 ***
structuresingle    0.[illegible]                                   ***
vowele            -0.[illegible]                                   ***
voweli            -0.[illegible]                                   ***
vowelo            -0.12292     0.04034   -3.048   0.002446 **
vowelu            -0.23653     0.04033   -5.864   8.87e-09 ***
rate              -0.25324     0.07425   -3.410   0.000708 ***
```

structure:double + vowel:a + rate:start

standard error of that mean

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

structure:double + vowel:a + rate:start

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.12186	0.03117	3.910	0.000107	***
vowele	-0.14406	0.04033	-3.572	0.000393	***
voweli				09	***
vowelo				46	**
vowelu				09	***
rate				08	***

to obtain the estimated mean value of `durationLog` in structure:single words, we have to add its estimate to the intercept, i.e.

-1.50620 + 0.12186 = -1.38434

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

structure:single + vowel:a + rate:start

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.12186	0.03117	3.910	0.000107	***
vowele	-0.14406	0.04033	-3.572	0.000393	***
voweli	-0.23739	0.04035	-5.883	7.97e-09	***
vowelo					
vowelu					
rate					

to obtain the estimated mean value of `durationLog` in structure:single & vowel:i words, we have to add both estimates to the intercept, i.e.

-1.50620 + 0.12186 - 0.23739 = -1.62173

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

structure:double + vowel:a + rate:start

```
                 Estimate                                         
(Intercept)      -1.50620                                    ***
structureopen     0.43395                                    ***
structuresingle   0.12186                                    ***
vowele           -0.14406                                    ***
voweli           -0.23739                                    ***
vowelo           -0.12292                                    **
vowelu           -0.23653    0.04033    -5.864    8.87e-09   ***
rate             -0.25324    0.07425    -3.410    0.000708   ***
```

`durationLog` is

- significantly longer in open coda words
- significantly longer in simple coda words

than in complex coda words

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

structure:double + vowel:a + rate:start

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.12186	0.03117	3.910	0.000107	***
vowele	-0.14406				***
voweli	-0.23739				***
vowelo	-0.12292				**
vowelu	-0.23653				***
rate	-0.25324				***

`durationLog` is

- significantly shorter in words with all other vowels, i.e. /e, i, o, u/

than in words with /a/

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.12186	0.03117	3.910	0.000107	***
vowele	0.14406	0.04033	3.572	0.000393	***
voweli	[illegible]	[illegible]	[illegible]	7.97e-09	***
vowelo	[illegible]	[illegible]	3.048	0.002446	**
vowelu	0.23653	0.04033	-5.864	8.87e-09	***
rate	-0.25324	0.07425	-3.410	0.000708	***

the higher the speaking rate, the lower the value of `durationLog`

Step 5: Interpretation – Effects

2. Using the `summary()` function, we can take a closer look at the **effects** of the individual predictors

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.50620	0.10486	-14.364	< 2e-16	***
structureopen	0.43395	0.03112	13.947	< 2e-16	***
structuresingle	0.12186	0.03117	3.910	0.000107	***
vowele	-0.14406	0.04033	-3.572	0.000393	***
voweli	-0.23739	0.04035	-5.883	7.97e-09	***
vowelo	-0.12292	0.04034	-3.048	0.002446	**
vowelu	-0.23653	0.04033	-5.864	8.87e-09	***
rate	-0.25324	0.07425	-3.410	0.000708	***

Step 5: Interpretation – Effects

2. Using another function, we can check the differences within one predictor

```
> tukey(model = mdl_fin, predictor = structure)

                     Estimate Std. Error t value Pr(>|t|)
open - double == 0    0.43395    0.03112   13.95  < 1e-04 ***
single - double == 0  0.12186    0.03117    3.91  0.00031 ***
single - open == 0   -0.31209    0.03111  -10.03  < 1e-04 ***
```