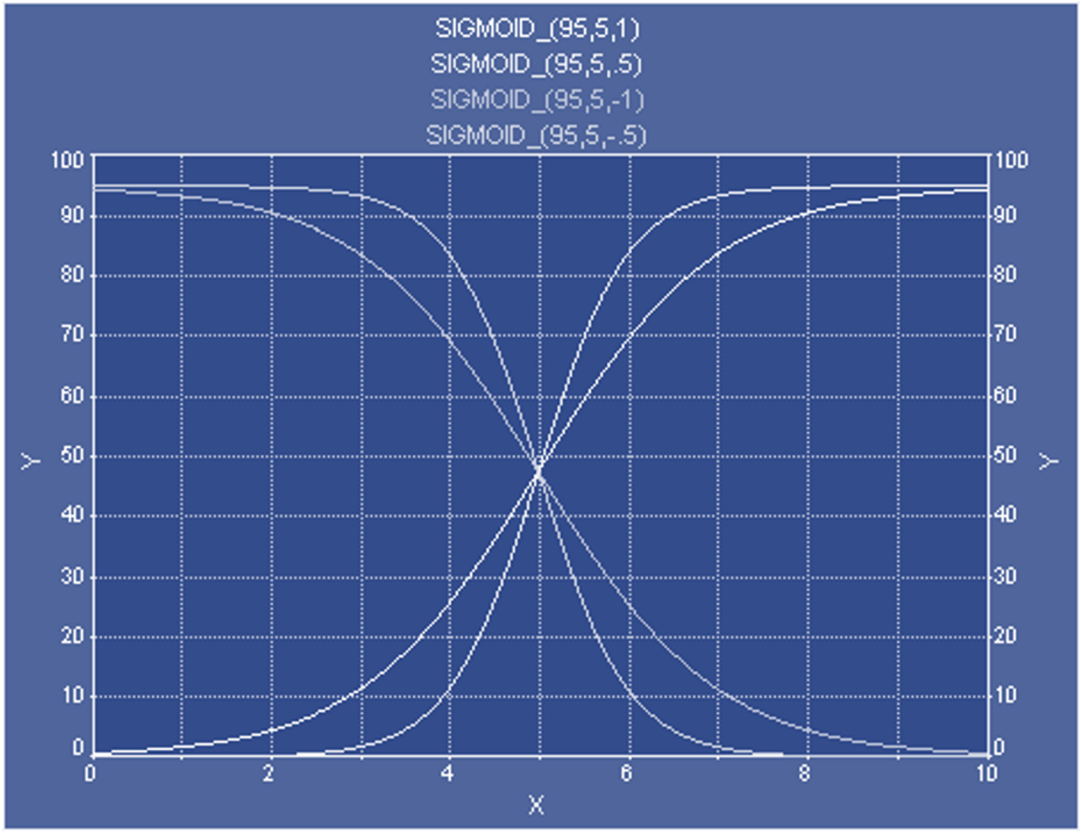
Model Families and Binary Logistic Regression



Based on
Jaeger 2008:
"Away from Anovas and towards
Logit mixed models"

based on slides by
Christoph Scheepers

$$y = \frac{a}{1 + \exp\left(-\frac{x-b}{c}\right)}$$

Vera Demberg, Saarland University

Exam info

- If you cannot sign up for the exam on HISPOS, please use the sign up form that I posted on TEAMS!

Generalized Linear Mixed Models, `glmer()`

- **Generalized Linear Mixed Models** are an extension of Linear Mixed Models that allow for the specification of ***distribution*** and ***link* functions** (via the **`family`** argument) to accommodate a variety of different data types (categorical, count, continuous, etc.)
- When no family argument is specified, `glmer()` assumes a *normal* distribution with *identity* link per default, i.e. `family=Gaussian(identity)`
- In fact, it is possible to call `lmer()` with a family argument as well, but in this case, you'll get a warning, and R will call `glmer()` instead:

  "… calling lmer with 'family' is deprecated; please use glmer() instead"

`lmer()` versus `glmer()`

- `lmer()` is for 'standard' mixed models (no transformation of parameters and assuming normality of residuals)
- `glmer()` is a generalization of `lmer()` that can be applied to a wider range of different data types (incl. binary), via appropriate **variance and link functions**
- In fact, `lm(y~x,…)` is conceptually equivalent to

```
glm(y~x,
    family=gaussian(identity),
    …)
```

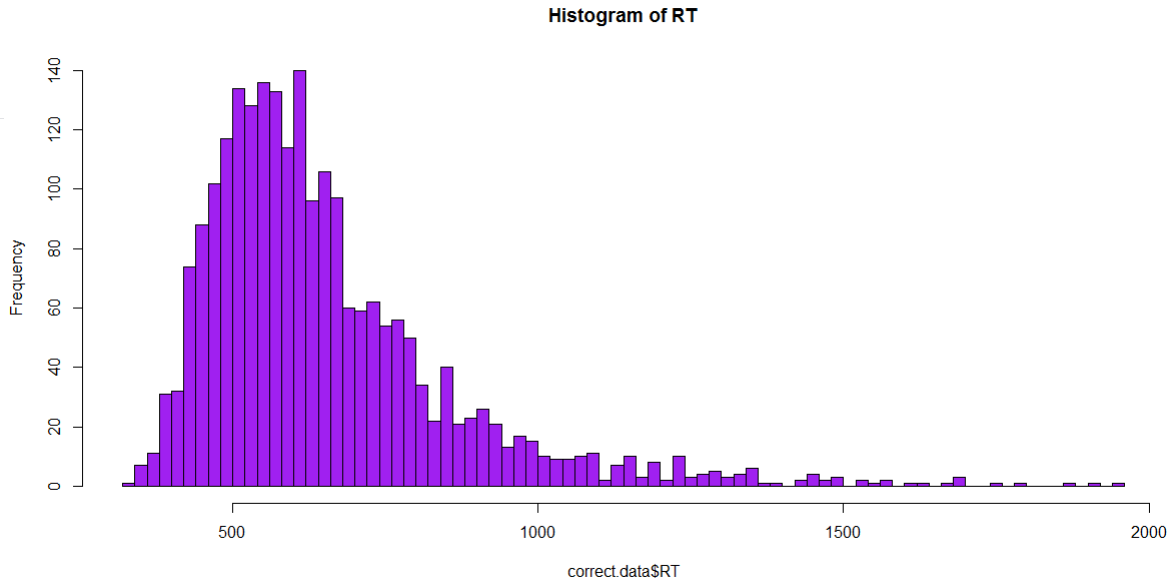
Model families available in `glmer()`:

Family	Variance	Link
gaussian	gaussian	identity
binomial	binomial	logit, probit or cloglog
poisson	poisson	log, identity or sqrt
Gamma	Gamma	inverse, identity or log
inverse.gaussian	inverse.gaussian	1/mu^2
quasi	user-defined	user-defined

- ***Variance*** concerns distribution of residuals
- ***Link*** applies a transformation to the model parameters and determines the interpretation of model coefficients (the latter will be given in 'link' units)

Example: Response Times

- RTs are hardly ever perfectly normally distributed!
- Characteristic *positive skew* in RT distributions (RTs are theoretically bounded to range from 0 to +∞)



- Some authors therefore recommend log-transforming RT data prior to analysis (*coerce* them into Normal), or model them using a *Gamma distribution function*

Previous Example: RT as a function of Word Type and Spelling

```
# Normal distribution and identity link ('default' assumptions)
model1 <- lmer(RT ~ WT*SP +
                 (1+WT*SP||subj) +
                 (1+SP||item),
               data = correct.data)
summary(model1)
```

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)  656.643     20.358   32.25
WT           -37.856     16.905   -2.24
SP            19.339      7.477    2.59
WT:SP        -31.845     15.294   -2.08
```

Previous Example: RT as a function of Word Type and Spelling

```
# The same again, but with log-transformed RTs
correct.data$logRT <- log(correct.data$RT)
model2 <- lmer(logRT ~ WT*SP +
                 (1+WT*SP||subj) +
                 (1+SP||item),
               data = correct.data)
summary(model2)
```

```
Fixed effects:
             Estimate Std. Error t value
(Intercept)  6.445417   0.029056  221.83
WT          -0.056026   0.023218   -2.41
SP           0.025037   0.009595    2.61
WT:SP       -0.039379   0.017986   -2.19
```

Previous Example: RT as a function of Word Type and Spelling

```
# Assuming a Gamma distribution for the residuals
model3 <- glmer(RT ~ WT*SP +
                  (1+WT*SP||subj) +
                  (1+SP||item),
                family = Gamma(identity),
                data = correct.data)
summary(model3)
```

```
Fixed effects:
            Estimate Std. Error t value Pr(>|z|)
(Intercept)  692.046     14.418   48.00  < 2e-16 ***
WT           -37.592     10.305   -3.65 0.000264 ***
SP            16.709      7.323    2.28 0.022504 *
WT:SP        -24.951      8.825   -2.83 0.004694 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Previous Example: RT as a function of Word Type and Spelling

```
# Goodness of Fit (Akaike Information Criterion)
# "Smaller is better"
AIC(model1, model2, model3)
```

```
       df         AIC
model1 11 29883.0253
model2 11 29619.5879 (-263.4374)
model3 11 28923.5269 (-606.0610)
```

- **For the present RT data at least, assuming a Gamma residual distribution appears to be the best approach!**

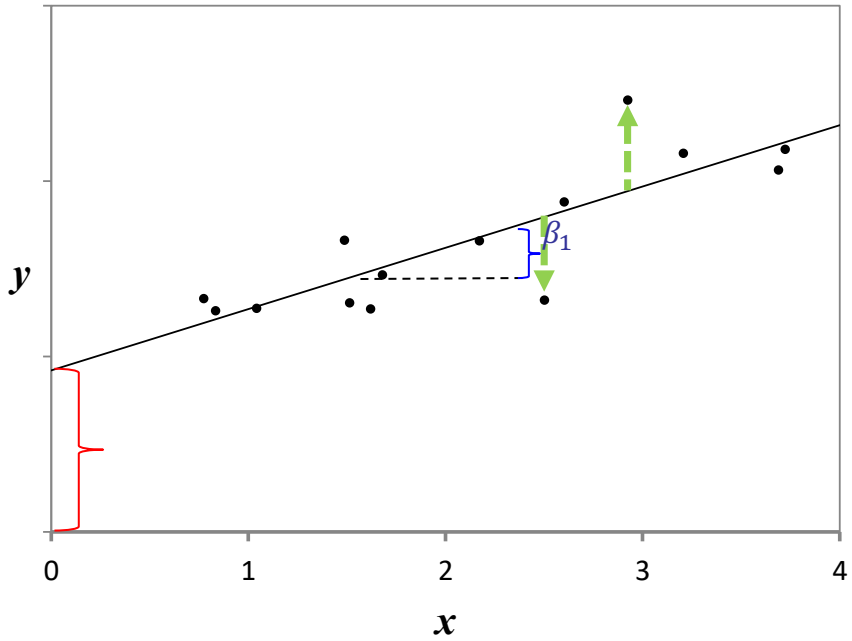
How important is the 'correct family'?

- **For *continuous* data,** the 'default' Normal distribution / Identity link assumption actually does a fairly good job in most cases
- ANOVA, for example, has been shown to be remarkably robust against violations of Normality
  - If anything, such violations are detrimental to Power, but not to Type I error rate (e.g., Khan & Rayner, 2003)
- However, other types of data require a more careful consideration of the correct model family
- A prominent example are ***binary categorical data***

Recall Simple Linear Regression

- **Goal**: Predict a continuous DV (*y*) from a continuous IV (x), assuming a *linear relationship* between the two

$$y_i = \beta_o + \beta_1 x_i + e_i$$



where

y_i = value of DV

x_i = value of the predictor variable

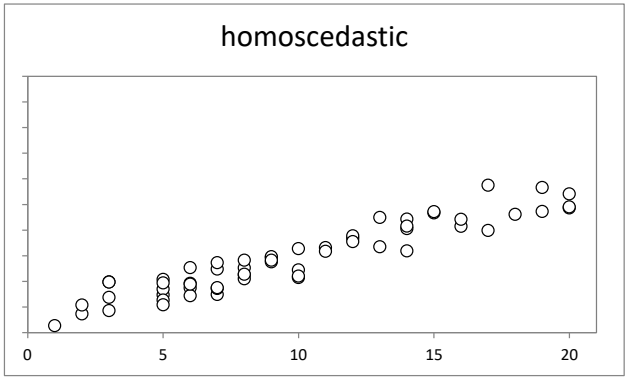
β_o = the ***intercept***

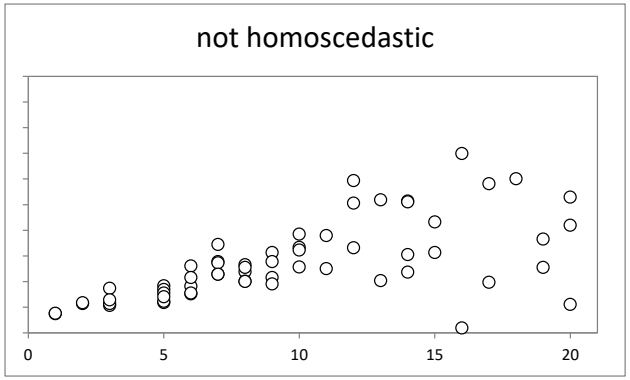
β_1 = the ***slope*** (or *regression coefficient*): the difference inassociated with a one-unit increase in x

e_i = prediction ***error*** (residuals)

Reminder: Assumptions

- Both IV and DV are measured on interval scale (continuous data)
  - Can theoretically range from $-\infty$ to $+\infty$
- Linearity / additivity
- Homoscedasticity
  - Constant variance of residuals over the entire x-range, e.g.





- Normality of residuals

Binary Categorical DVs

- Sometimes the values of the DV of interest come in only two flavours, i.e. 0 or 1
  - Female/Male, pregnant/not pregnant, correct/incorrect, … etc.
- That is, what we want to do is to somehow **predict the probability of belonging to one or the other category** as a function of our IV
- Linear regression would not work in this case
  - Binary data are nominal scale (discrete), and their probabilities are bound between 0 and 1 (with small / large x_i, linear regression might well result in y_i-values <0 or >1)
  - The relationship between x and y will **not** be **linear**
  - **Normality?** The appropriate distribution for numbers of '1s' in a sequence of independent Bernoulli (0,1) trials is actually the **binomial** distribution
  - **Heteroscedasticity** of residuals: the closer predicted probability values are to 0 or 1, the smaller the corresponding residual variances will be (error variance will be greatest when predicted probability is around .5)

Binary Categorical DVs

Illustration of Heteroscedasticity for binary DV:

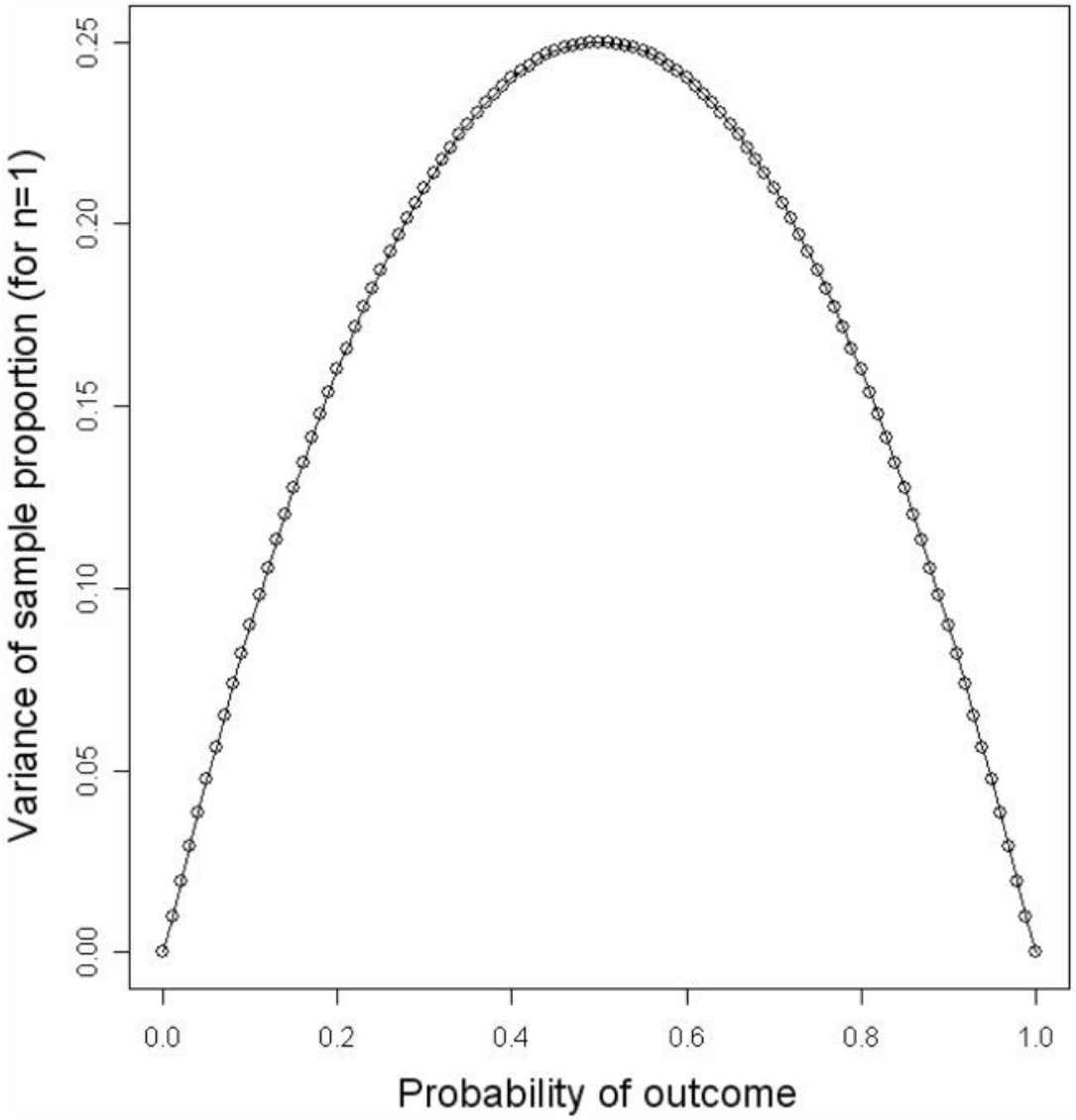
For binary variables, the population variance is p x q

(where p is the probability of the one outcome, and q is the probability of the other outcome, cf. lecture 4).

Thus, if p=0.5 and q=0.5 → var = 0.25

if p=0.1 and q=0.9 → var = 0.09

Heteroscedasticity



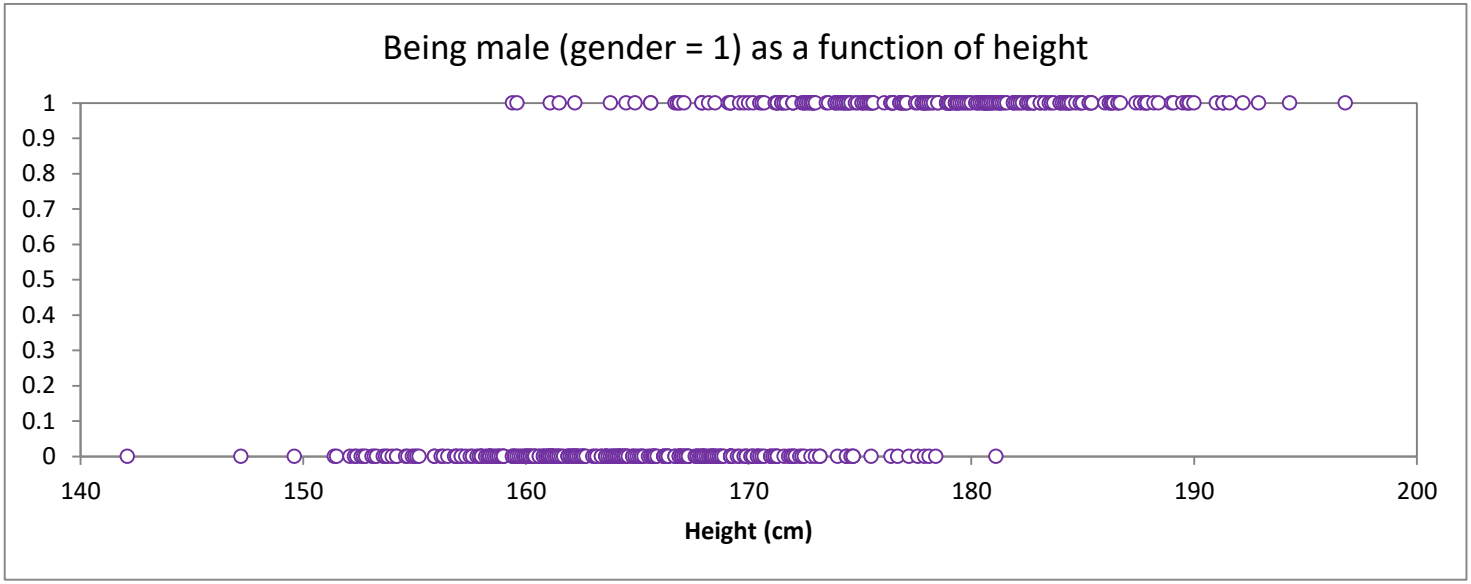
An Example

- Let's suppose we randomly sampled 500 German adults and measured their body height in cm
- Here's a generated dataset using the following parameters:
- N = 500
- Gender ~ $U(0,1)$ (male / female) => roughly 50% males
- height | gender = 0 (female) ~ $N(163.5, 6.1)$
- height | gender = 1 (male) ~ $N(178.2, 7.0)$

- **Goal**: We want to predict a person's **gender** from their body height
  - **Classification** problem with
    - Continuous IV (height in cm)
    - Binary DV (female = 0, male = 1)

An Example

subj_ID	height	gender
1	156.8	0
2	166.9	0
3	164.5	0
4	191.6	1
5	161.5	0
6	182.8	1
7	180.7	1
8	162.2	0
9	164.4	0
10	166.9	0
11	178.4	0
12	155.9	0
13	161.1	0
14	154.2	0
15	161	0
16	180.6	1
17	162.5	0
18	179.5	1
19	153.3	0
20	165.6	0
21	184.4	1
…	…	…



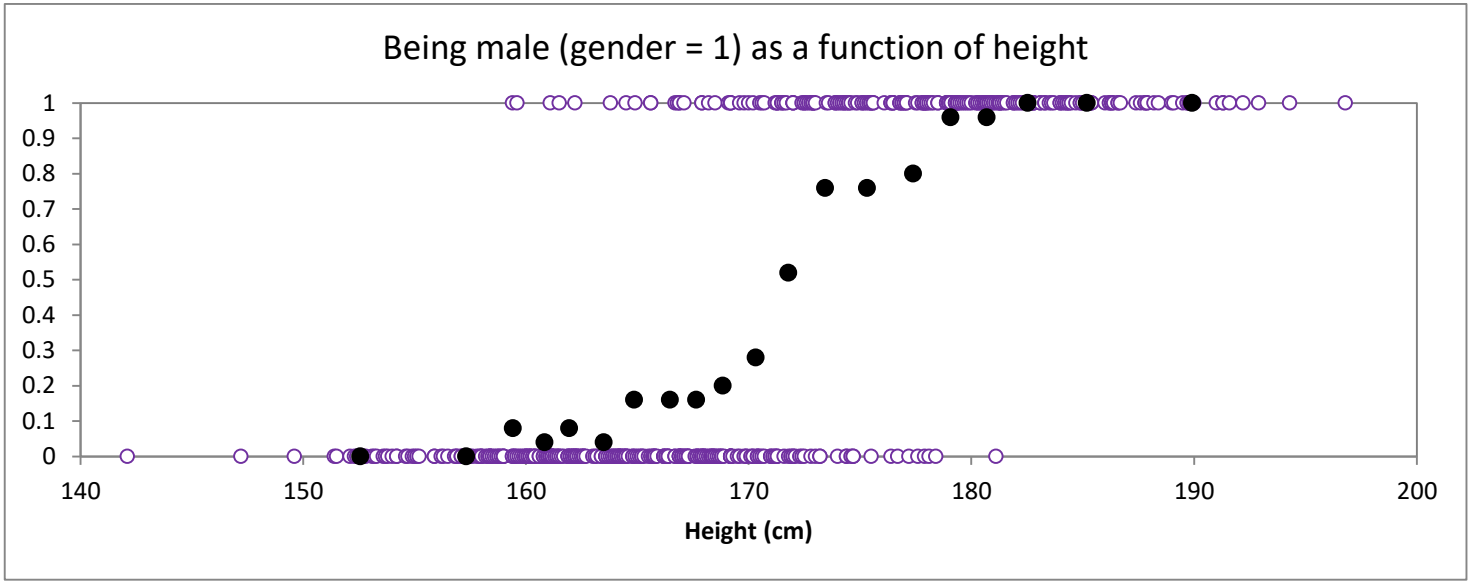
- DV (male) is coded as 0 (for female) or 1 (for male)
- If you plot the data in R using:

```
plot(gender ~ height)
```

It would look like the above

An Example

subj_ID	height	gender
1	156.8	0
2	166.9	0
3	164.5	0
4	191.6	1
5	161.5	0
6	182.8	1
7	180.7	1
8	162.2	0
9	164.4	0
10	166.9	0
11	178.4	0
12	155.9	0
13	161.1	0
14	154.2	0
15	161	0
16	180.6	1
17	162.5	0
18	179.5	1
19	153.3	0
20	165.6	0
21	184.4	1
…	…	…



- When we look at the **probability** of being male (here, for each 5% height-bin), we see that P(gender=1) as a function of height follows a roughly "S-shaped" (*sigmoid*) function
- This is a natural consequence of the two **partially overlapping** normal height-distributions (one for males and one for females)

Logistic Function

The probability of being male as a function of height – or more generally, the probability of given binary category y as a function of x – can be modeled by the following equation:

$$P(y_i) = \frac{e^{\beta_0 + \beta_i x_i}}{1 + e^{\beta_0 + \beta_i x_i}}$$

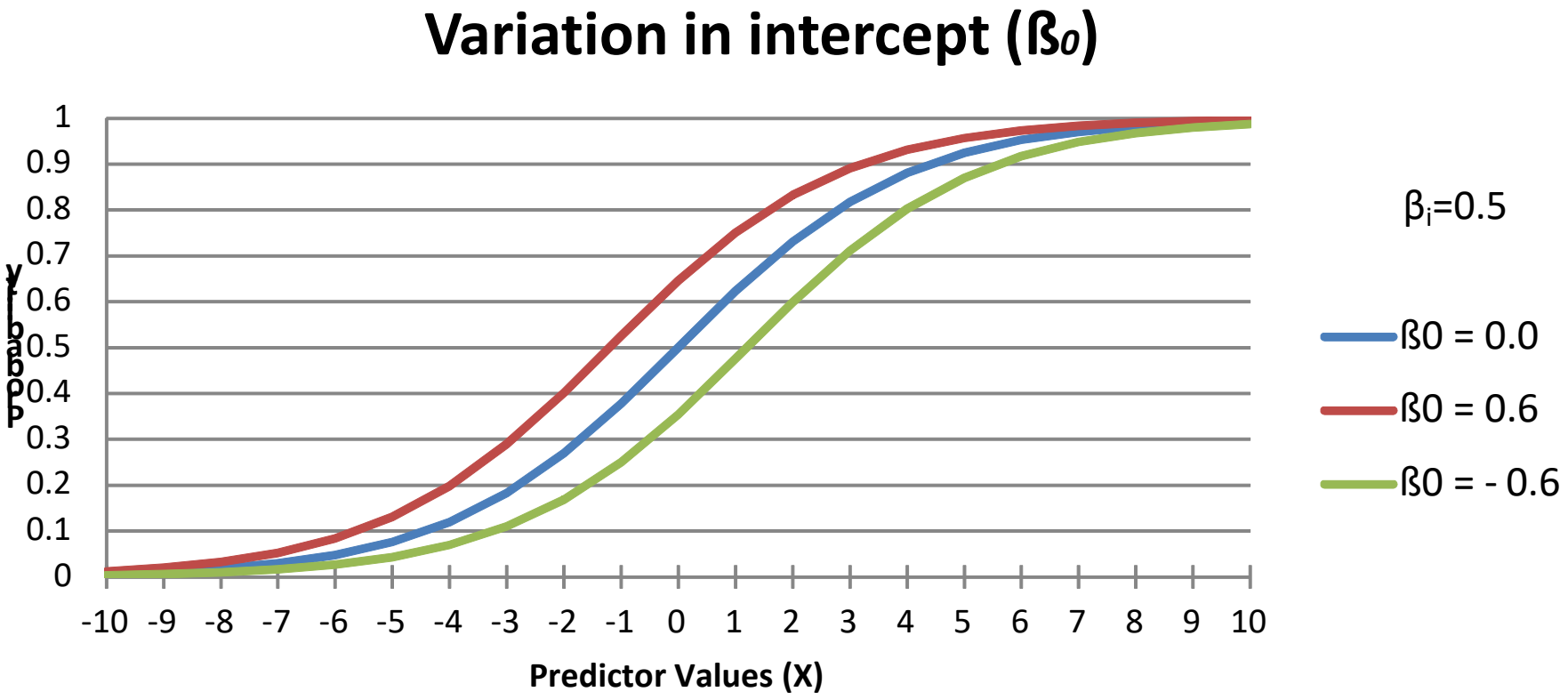
Looks suspiciously like our good old *linear regression* model
- More later!

or

$$P(y_i) = \frac{1}{1 + e^{-(\beta_0 + \beta_i x_i)}}$$

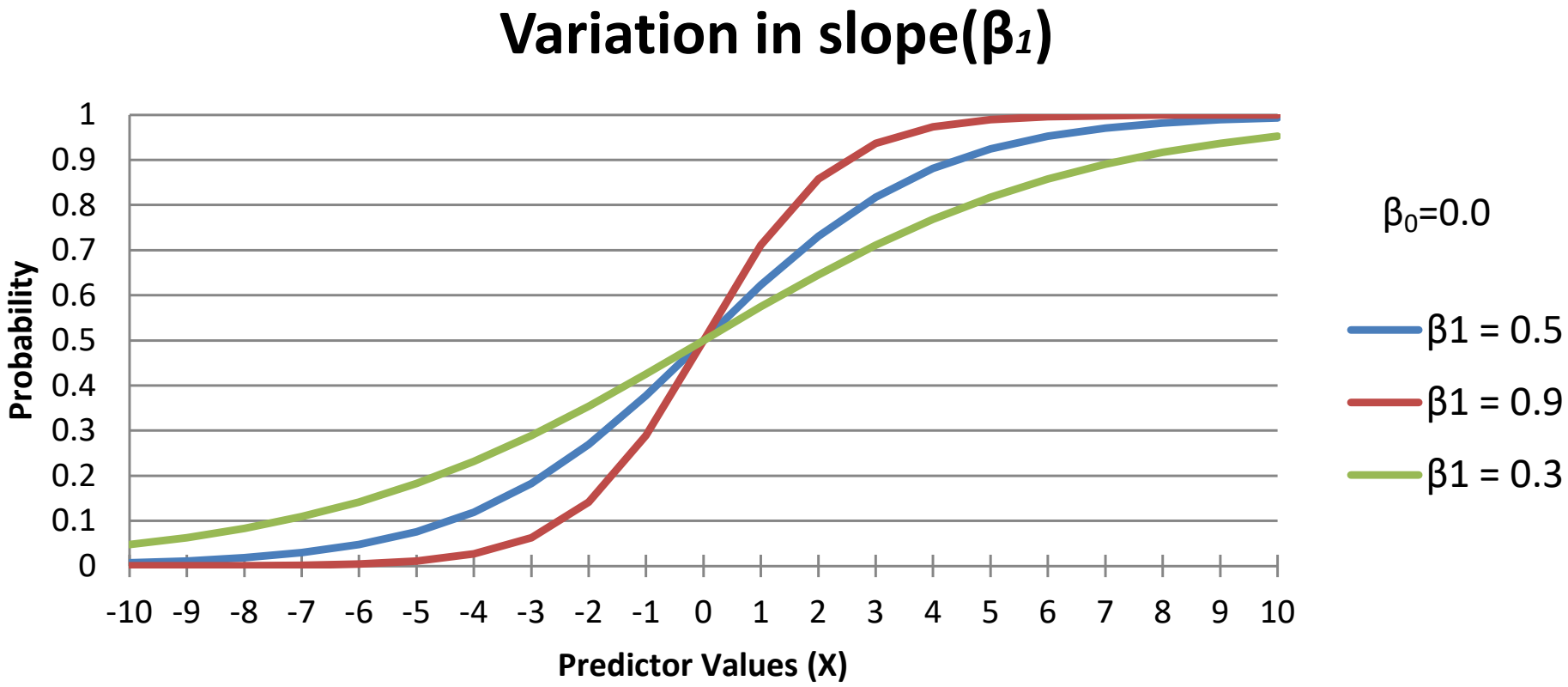
Logistic Function

$$P(y_i) = \frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}} \quad or \quad P(y_i) = \frac{1}{1+e^{-(\beta_0+\beta_i x_i)}}$$



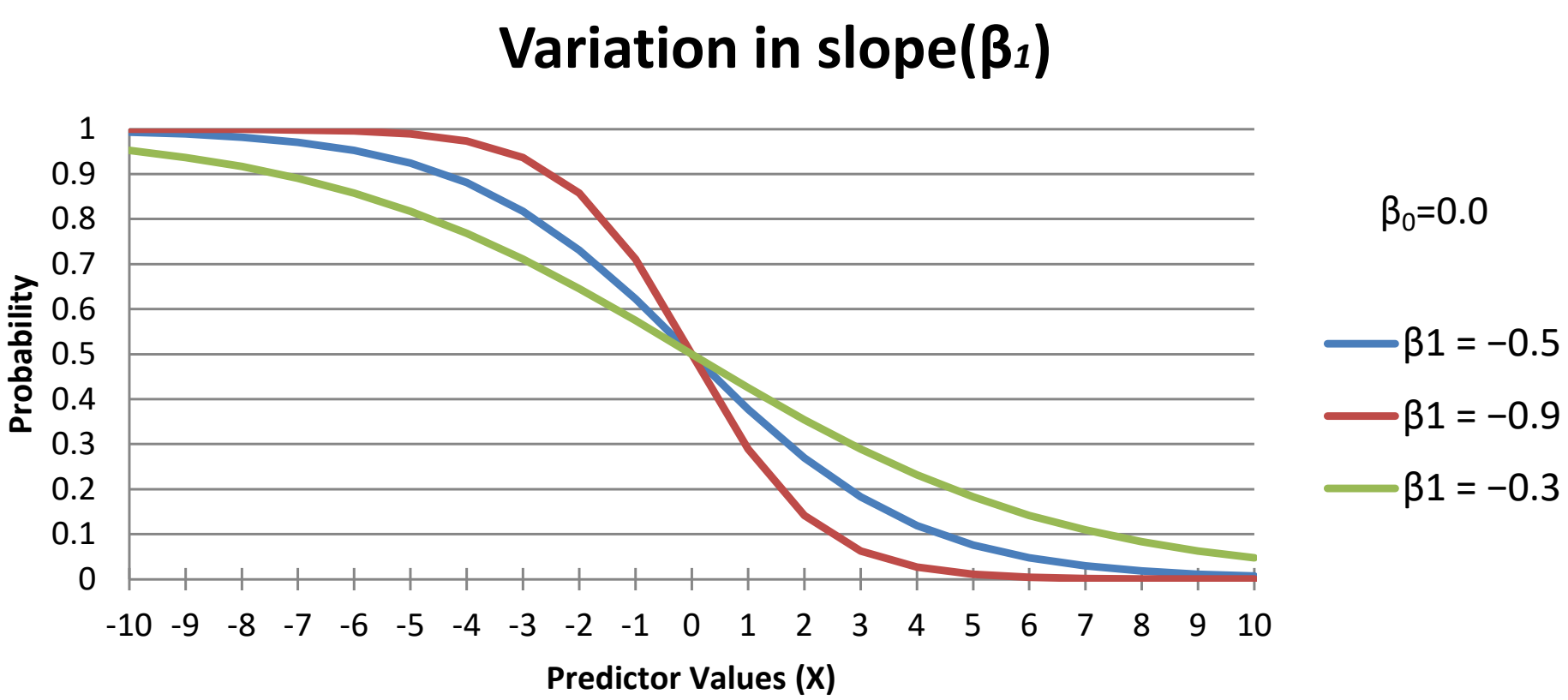
Logistic Function

$$P(y_i) = \frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}} \quad or \quad P(y_i) = \frac{1}{1+e^{-(\beta_0+\beta_i x_i)}}$$



Logistic Function

$$P(y_i) = \frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}} \quad or \quad P(y_i) = \frac{1}{1+e^{-(\beta_0+\beta_i x_i)}}$$



Odds

- Instead of probabilities, we could also conceptualize the problem in terms of odds
- What are the odds of being male given a certain probability of being male?
  Answer: $odds(male) = \frac{P(male)}{P(female)} = \frac{P(male)}{1-P(male)}$
  More generally: $odds(y) = \frac{P(y)}{1-P(y)}$
- Say, if in a given sample the probability of being male is .6, the odds of being male are $.6/.4 = 1.5$, i.e. in that sample, its 1.5 times more likely to find males than females.

Odds

$$odds(p) = \frac{p}{1-p} \quad \text{and} \quad p(odds) = \frac{odds}{1+odds} \tag{8}$$

Thus, odds increase with increasing probabilities, with odds ranging from 0 to positive infinity and odds of 1 corresponding to a proportion of 0.5. Differences in odds are usually described multiplicatively (i.e. in terms of x-fold increases or decreases). For example, the odds of being on a plane with a drunken pilot are reported to be "1 to 117" (http://www.funny2.com/). In the notation used here, this corresponds to odds of $1 / 117 \approx 0.0086$.

Log Odds or "Logits"

The natural logarithm of $odds(y)$ is called **log odds** or **logit**.

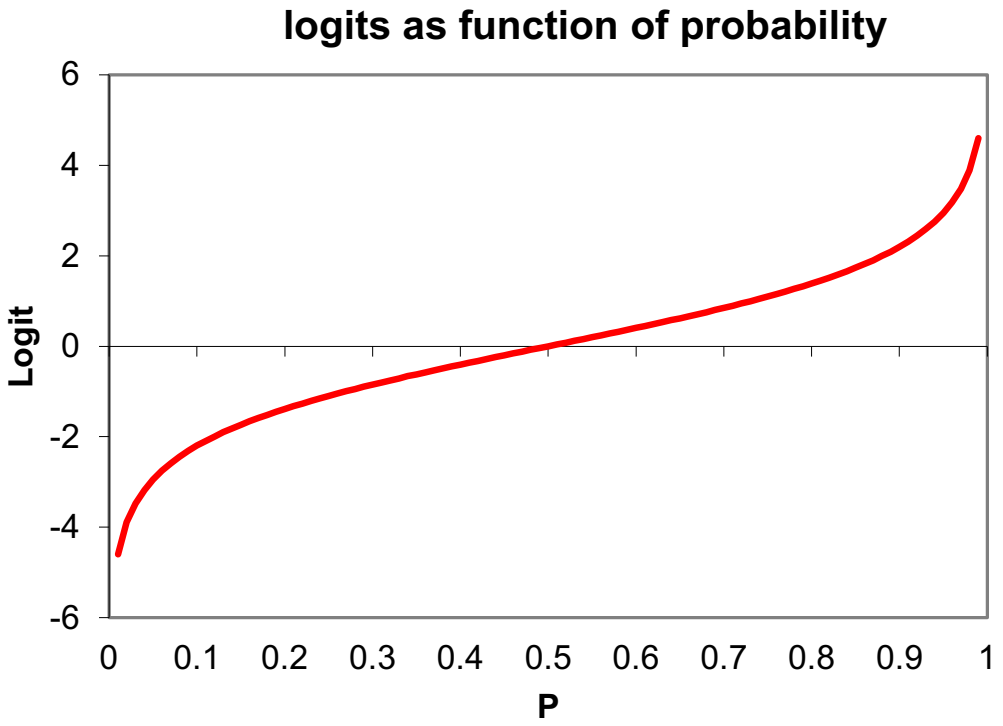
$$logit(y) = \ln(odds(y)) = \ln(\frac{P(y)}{1 - P(y)})$$

where $\ln(x)$ refers to the log based on Euler's number (ca. 2.71828182845904523536028747135 27...)

Logits have the following properties:

- If $odds(y) = 1; p = q = 0.5; logit(y) = 0$
- If $odds(y) < 1; p < 0.5; logit(y) < 0$
- If $odds(y) > 1; p > 0.5; logit(y) > 0$
- the logit transform fails if $p = 0$ or $p = 1$
- Logits range between ∞ to $+\infty$.

Logit as a function of *P*



- In the "middle" probability range, small changes in *P* imply small changes in *logit*
- When probabilities approach one of the logical boundaries (0 or 1), small changes in *P* imply large changes in *logit*
- Compensates for the heteroscedasticity problem associated with probabilities

To convert logits back into probabilities, use:

$$P = \frac{e^{logit}}{1 + e^{logit}}$$

Interesting, but why care?..

Since

$$P(y_i) = \frac{e^{\beta_0+\beta_i x_i}}{1 + e^{\beta_0+\beta_i x_i}}$$

it follows that

$$\frac{P(y_i)}{1-P(y_i)} = \frac{\frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}}{1 - \frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}} = \frac{\frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}}{\frac{1+e^{\beta_0+\beta_i x_i}-e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}} = \frac{\frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}}{\frac{1}{1+e^{\beta_0+\beta_i x_i}}} = e^{\beta_0+\beta_i x_i}$$

and therefore

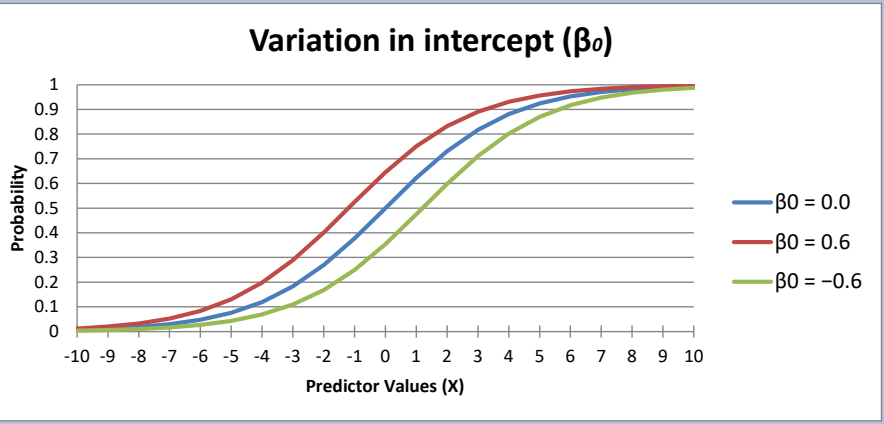
$$\ln\left(\frac{P(y_i)}{1-P(y_i)}\right) = \beta_0 + \beta_i x_i$$

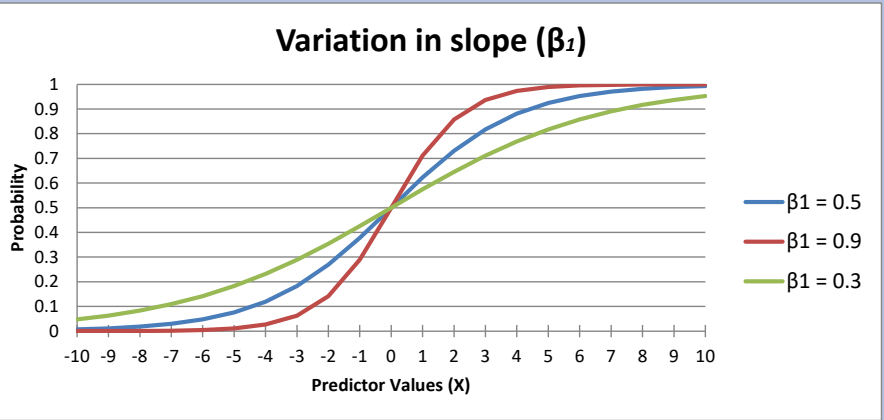
In other words: Applying a logistic function to $P(y)$ is the same as applying a linear function to the log odds (or logit) of $P(y)$.

This is what **binary logistic regression** does!

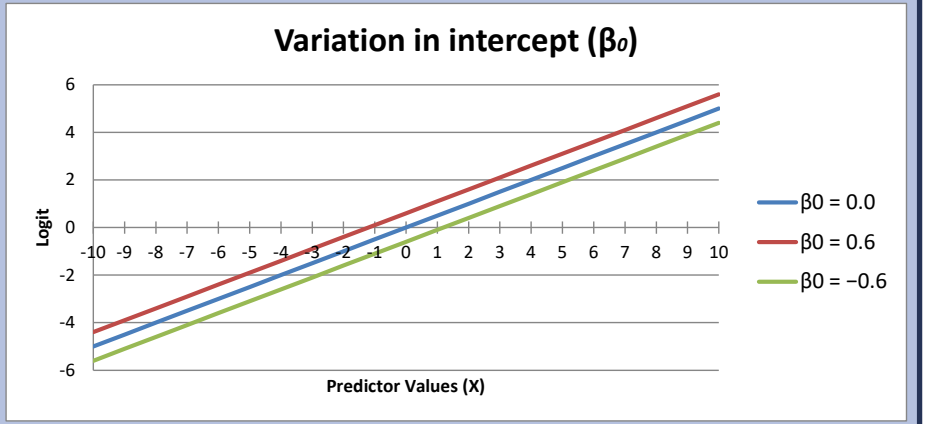
Just to confirm…

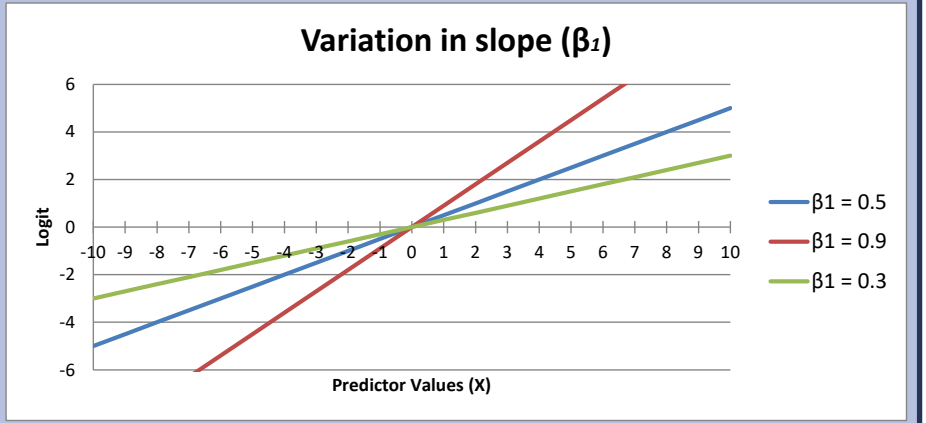
The parish of probabilities





The *land of logits*





Example

- **Real data** from a paper & pencil questionnaire study reported in Experiment 2 reported in

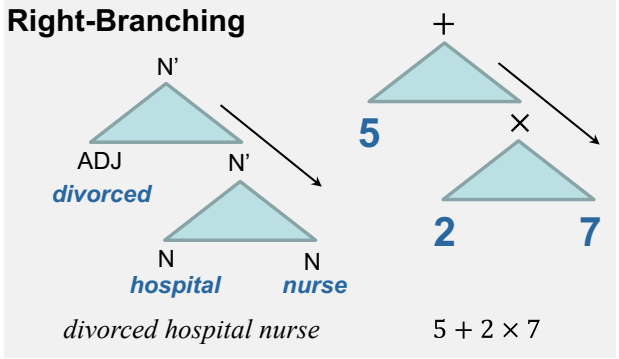
  Scheepers, C., & Sturt, P. (2014). Bidirectional syntactic priming across cognitive domains: From arithmetic to language and back. *Quarterly Journal of Experimental Psychology*, 67(8), 1643-1654 (doi: 10.1080/17470218.2013.873815)

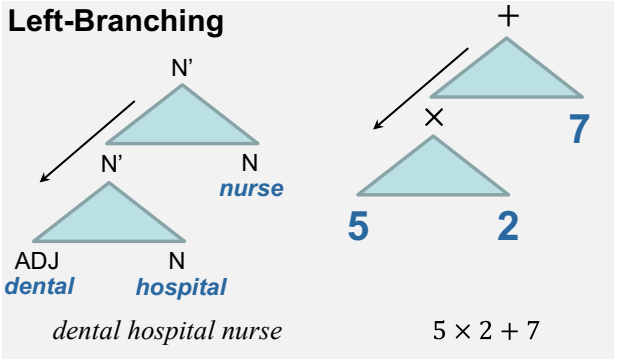
- **Question**: Is there syntactic priming (facilitation of structural processing from one trial to the next) from language to mathematics?
- **Subjects**: 36 (pretested) participants with '*less than perfect*' mathematical skills
- Stimulus materials (**items**): 24 prime-target pairs (i.e. a linguistic expression [to be rated for plausibility] followed by a simple mathematical equation [to be solved correctly]) – plus lots of 'filler items' in-between those pairs

Stimuli, Design

24 critical Items like this:

Prime Type	Target Type
RB (e.g., divorced hospital nurse)	RB (e.g., 5 + 2 * 7 =)
RB (e.g., divorced hospital nurse)	LB (e.g., 5 * 2 + 7 =)
LB (e.g., dental hospital nurse)	RB (e.g., 5 + 2 * 7 =)
LB (e.g., dental hospital nurse)	LB (e.g., 5 * 2 + 7 =)





2 * 2 factorial design, 4 conditions

- **Within subjects**
  (each participant [N=36] saw 6 items per condition)
- **Within items**
  (using a Latin square, each item*condition combo was seen by 9 subjects)

Task, DV, Hypotheses, etc.

- Rate linguistic expressions (primes) for plausibility and solve equations (targets) correctly
- Binary DV: Accuracy of target equation solving (1=correct, 0=incorrect)
- **Hypothesis**: Accuracy should be higher when prime and target have the same rather than a different structure => **Prime Type * Target Type interaction**

Trials

…

Filler equation or expression [solve / rate]

Filler equation or expression [solve / rate]

Prime: dental hospital nurse [rate 1-5]

Target: 5 + 2 × 7 = [solve]

Filler equation or expression [solve / rate]

Filler equation or expression [solve / rate]

Prime: capsized oil tanker [rate 1-5]

Target: 64 / 8 − 4 = [solve]

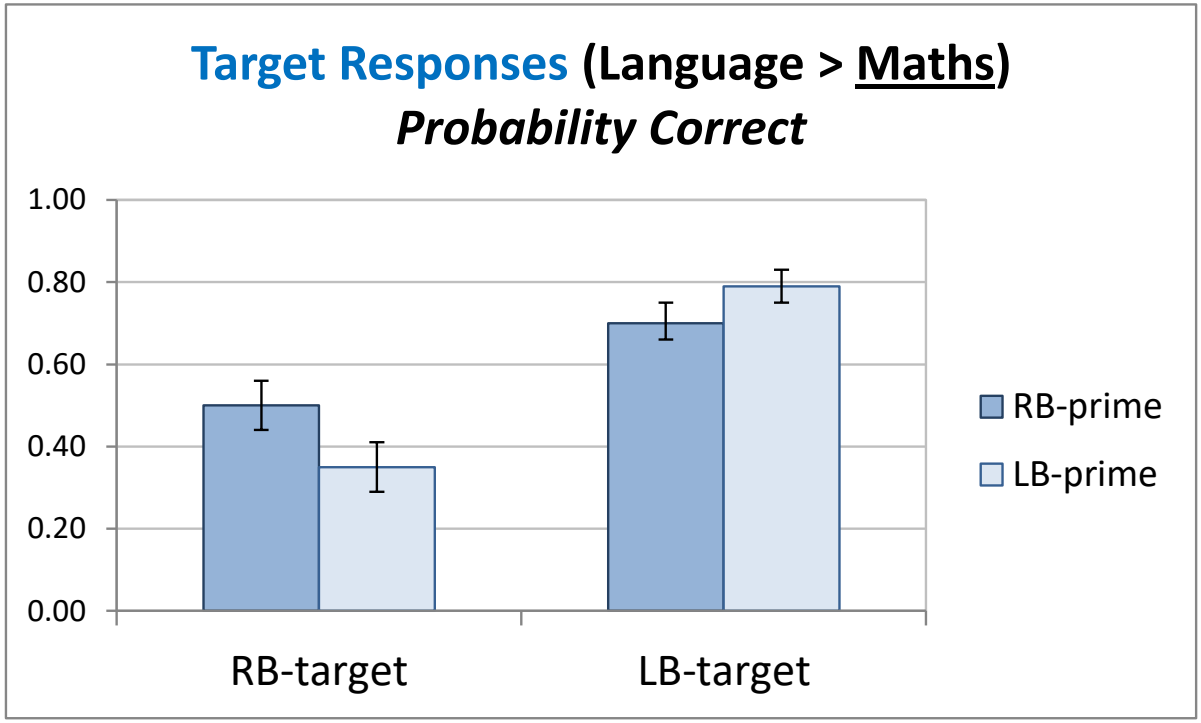
Filler equation or expression [solve / rate]

Filler equation or expression [solve / rate]

…

.

Descriptive Results



- Main effect of target type (LB > RB)
- Interaction as predicted
- **We want to analyse the data using generalized linear mixed models with 'crossed' random effects for subjects and items**

The Data

```
# Math-priming CSV file
MP.data <- read.csv("MPdata.csv")
```

```
> head(MP.data, 10)
      Sbj Itm prime target correct
1  L2M01  23     2      1       0
2  L2M01  16     1      2       1
3  L2M01  13     1      1       0
4  L2M01   7     2      1       0
5  L2M01  20     1      2       0
6  L2M01  24     1      2       0
7  L2M01  18     2      2       1
8  L2M01  21     1      1       0
9  L2M01  22     2      2       1
10 L2M01  14     2      2       1
```

Deviation coding of IVs

```
# New variable PT (prime type): dummy-coded, then "centred"
# (=> deviation coding)
MP.data$PT <- scale(ifelse(MP.data$prime==1,1,0), scale=FALSE)

# New variable TT (target type): dummy-coded, then "centred"
# (=> deviation coding)
MP.data$TT <- scale(ifelse(MP.data$target==1,1,0), scale=FALSE)
```

```
> head(MP.data)
    Sbj Itm prime target correct   PT   TT
1 L2M01  23     2      1       0 -0.5  0.5
2 L2M01  16     1      2       1  0.5 -0.5
3 L2M01  13     1      1       0  0.5  0.5
4 L2M01   7     2      1       0 -0.5  0.5
5 L2M01  20     1      2       0  0.5 -0.5
6 L2M01  24     1      2       0  0.5 -0.5
```

Running the 'maximal' model in `glmer()`

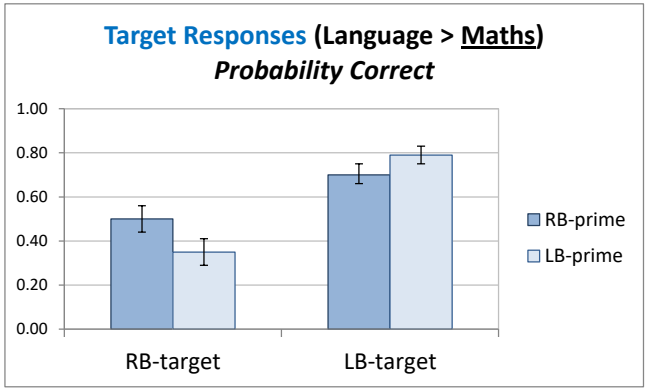
```
# You need to install the lme4 package first
library(lme4)

# Run binary logistic LME in glmer(); use full
# factorial model with design-appropriate maximal
# random effects structure(including all random
# correlations)
glmer.max <- glmer(correct ~ PT*TT +
                     (1 + PT*TT | Sbj) +
                     (1 + PT*TT | Itm),
                   family=binomial(logit),
                   data=MP.data)
summary(glmer.max)
```

The output

```
Random effects:
 Groups Name        Variance Std.Dev. Corr
 Sbj    (Intercept)  0.58168 0.7627
        PT           1.61480 1.2707   -0.36
        TT          10.80567 3.2872    0.70 -0.32
        PT:TT        3.69131 1.9213   -0.23 -0.29 -0.57
 Itm    (Intercept)  0.03187 0.1785
        PT           0.17322 0.4162   -1.00
        TT           0.40133 0.6335   -0.98  0.98
        PT:TT        0.60990 0.7810   -0.39  0.39  0.18
Number of obs: 864, groups:  Sbj, 36; Itm, 24
```

```
Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.4753     0.1938   2.453 0.014176 *
PT            0.2404     0.3561   0.675 0.499634
TT           -2.2910     0.6299  -3.637 0.000276 ***
PT:TT         2.2540     0.6510   3.463 0.000535 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Correlation of Fixed Effects:
      (Intr) PT     TT
PT    -0.459
TT     0.339 -0.111
PT:TT -0.070 -0.131 -0.415
convergence code: 0
unable to evaluate scaled gradient
Model failed to converge: degenerate  Hessian with 1 negative
eigenvalues
```



Main effect of **target type** and **prime type * target type** interaction are indeed significant (as suggested by descriptives)

However, convergence issues… would next have to simplify random effects structure.

But let's take a look also at other cases

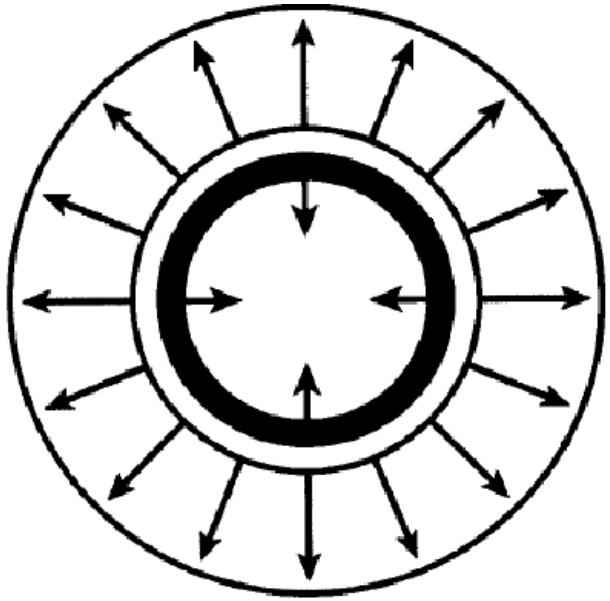
Family	Variance	Link
gaussian	gaussian	identity
binomial	binomial	logit, probit or cloglog
poisson	poisson	log, identity or sqrt
Gamma	Gamma	inverse, identity or log
inverse.gaussian	inverse.gaussian	1/mu^2
quasi	user-defined	user-defined

But let's take a look also at other cases

- We just saw what to do when the DV is a binary variable
- What if the DV is a count variable?

Example:

Pupillometric measure
"Index of Cognitive Activity"
counts the number of rapid pupil dilations per second



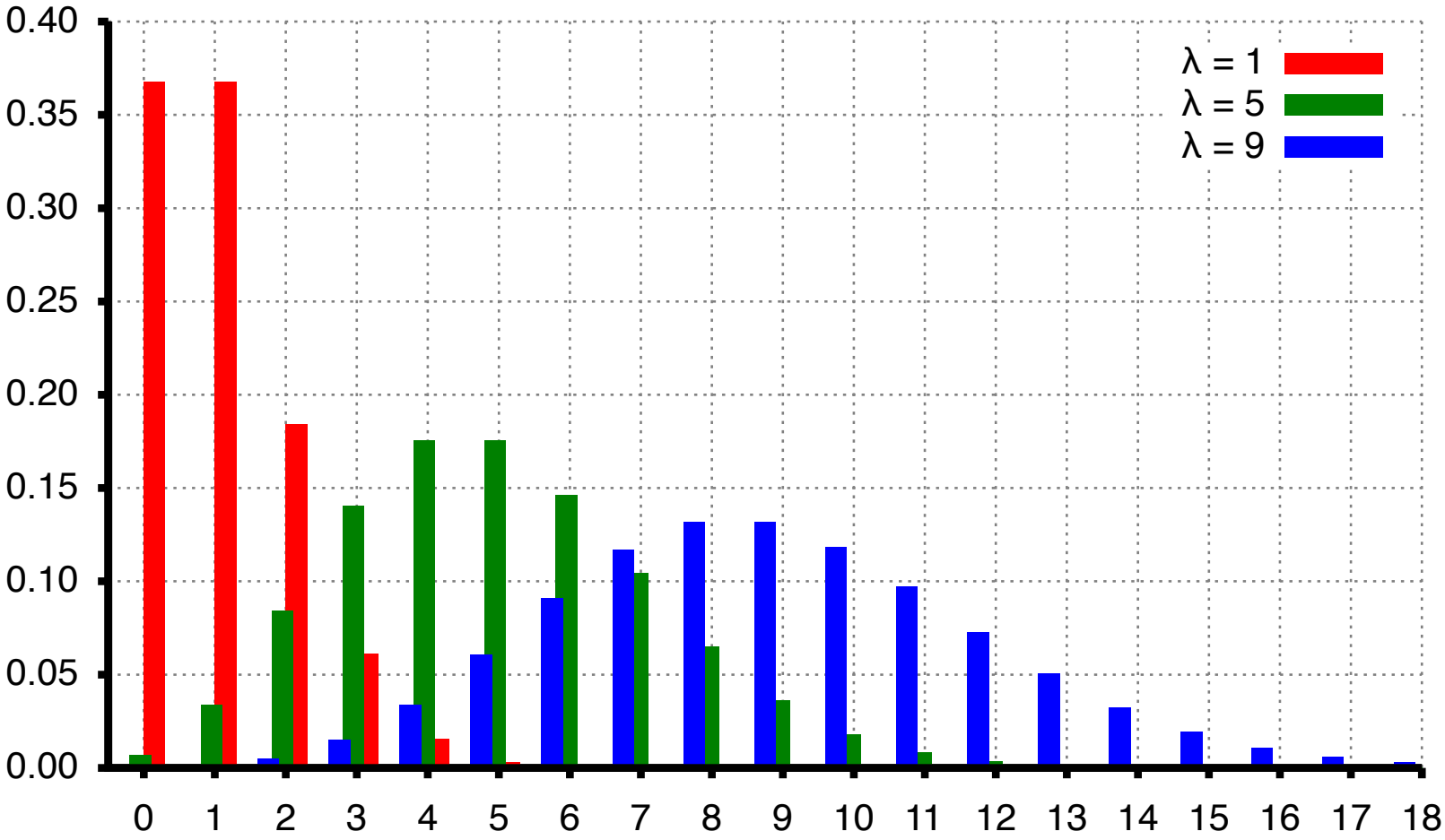
from: Beatty & Lucero-Wagoner 2000

Count variables

Can’t treat count variables like normal distribution:

- there cannot be any values < 0
- you can only have integer numbers
- the distribution usually has a heavy tail to the right (especially for cases where the number of expected counts is low).

Poisson Distribution



λ indicates the expected value.

graph taken from Wikipedia

```
> ml <- glmer(rawica ~ it + (1 +it| ITEM_ID) + (1+it| PARTICIPANT), data=subset(css,
eye=="RIGHT_ICA_EVENT"), family=poisson)
> summary(ml)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [
glmerMod]
 Family: poisson  ( log )
Formula: rawica ~ it + (1 + it | ITEM_ID) + (1 + it | PARTICIPANT)
   Data: subset(css, eye == "RIGHT_ICA_EVENT")

     AIC      BIC   logLik deviance df.resid
  4101.1   4135.9  -2042.5   4085.1      564

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.3077 -0.8459  0.0563  0.9773  3.9332

Random effects:
 Groups      Name        Variance Std.Dev. Corr
 ITEM_ID     (Intercept) 0.009788 0.09893
             it1         0.024351 0.15605  -0.85
 PARTICIPANT (Intercept) 0.033964 0.18429
             it1         0.043237 0.20793  -0.72
Number of obs: 572, groups:  ITEM_ID, 24; PARTICIPANT, 24

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.75592    0.04529   60.86   <2e-16 ***
it1          0.13342    0.05694    2.34   0.0191 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
    (Intr)
it1 -0.749
```

Conclusion

- **Generalized Linear Mixed Models (`glmer()`)** are a very powerful analysis tool indeed
- They can
  - handle data from repeated-measures designs via inclusion of design-appropriate random effect structures.
  - deal with DVs which are not normally distributed
  - even estimate 'crossed' random effects (simultaneous generalization to subject and item populations.

Useful References

Baayen, R. H., Davidson, D. J., & Bates, D. M. (2008). Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, 59, 390–412.

Barr, D.J., Levy, R., Scheepers., C., & Tily, H.J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, 68(3), 255-278.

Jaeger, T. F. (2008). Categorical data analysis: Away from ANOVAs (transformation or not) and towards logit mixed models. *Journal of Memory and Language*, 59, 434–446.