Course Business

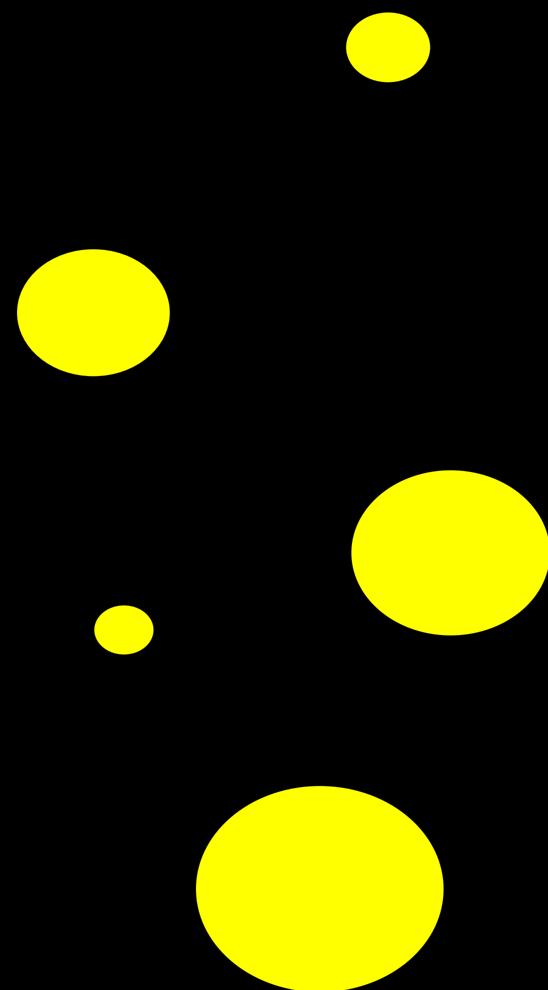
- New R package to install: `misty`
- On Canvas for this week:
  - New dataset we'll cover in lecture: `numerosity`
  - Lab materials
- Follow-up on last Wednesday
  - Thanks for working with change in schedule
  - Apologies for not having a larger lab available

Week 7.1: Centering & Transformations

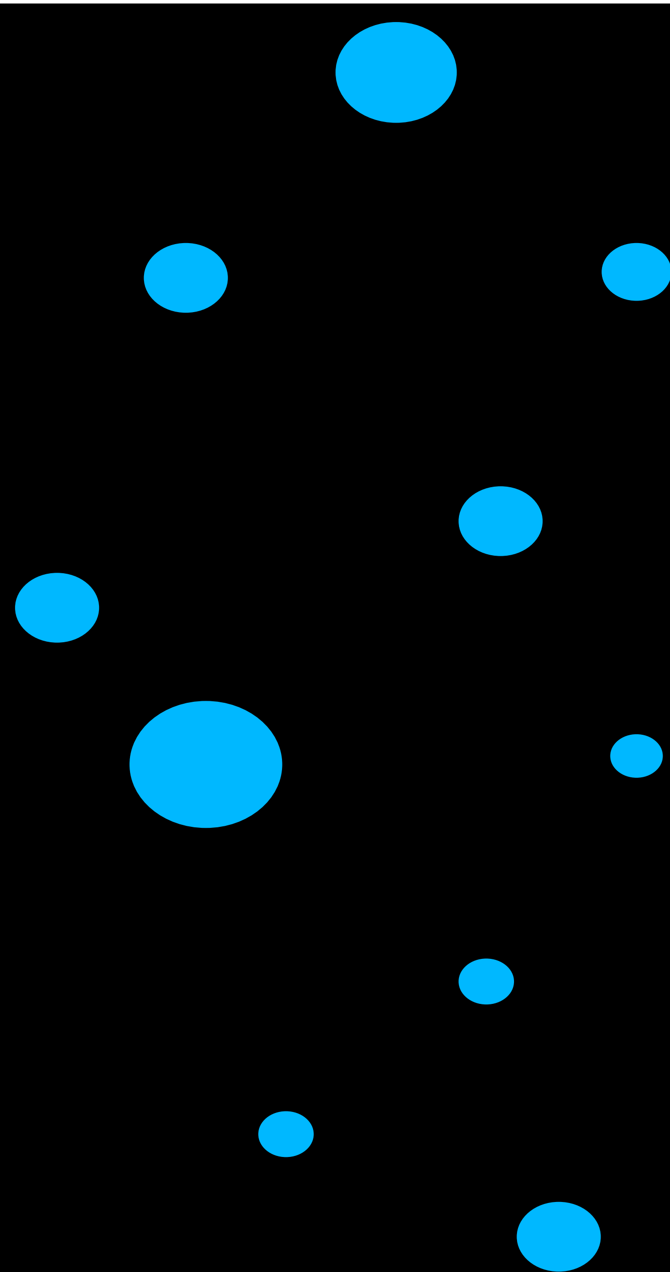
- Centering
  - Today’s Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab

Today's Dataset

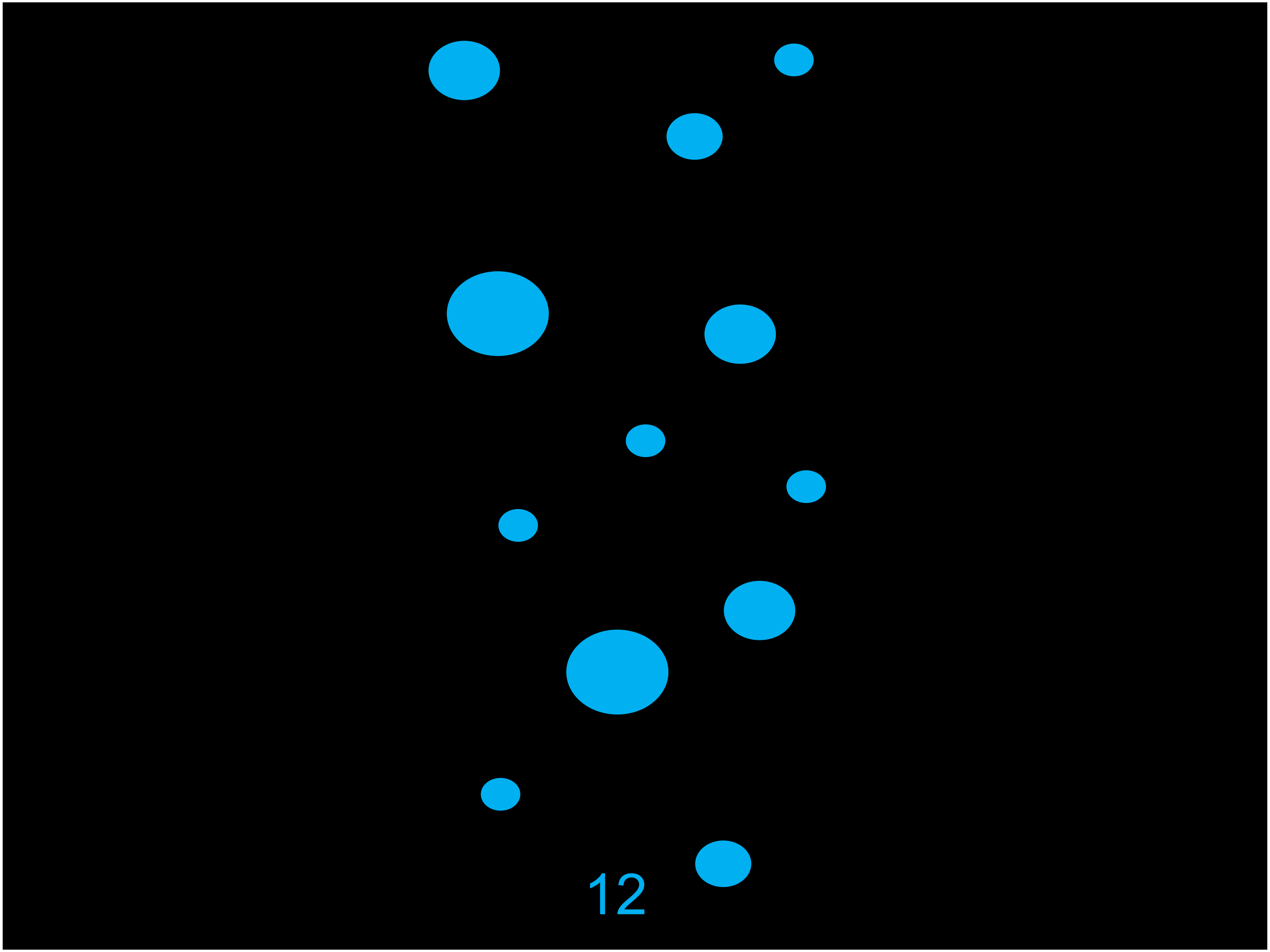
- Brain warmup – count the number of dots

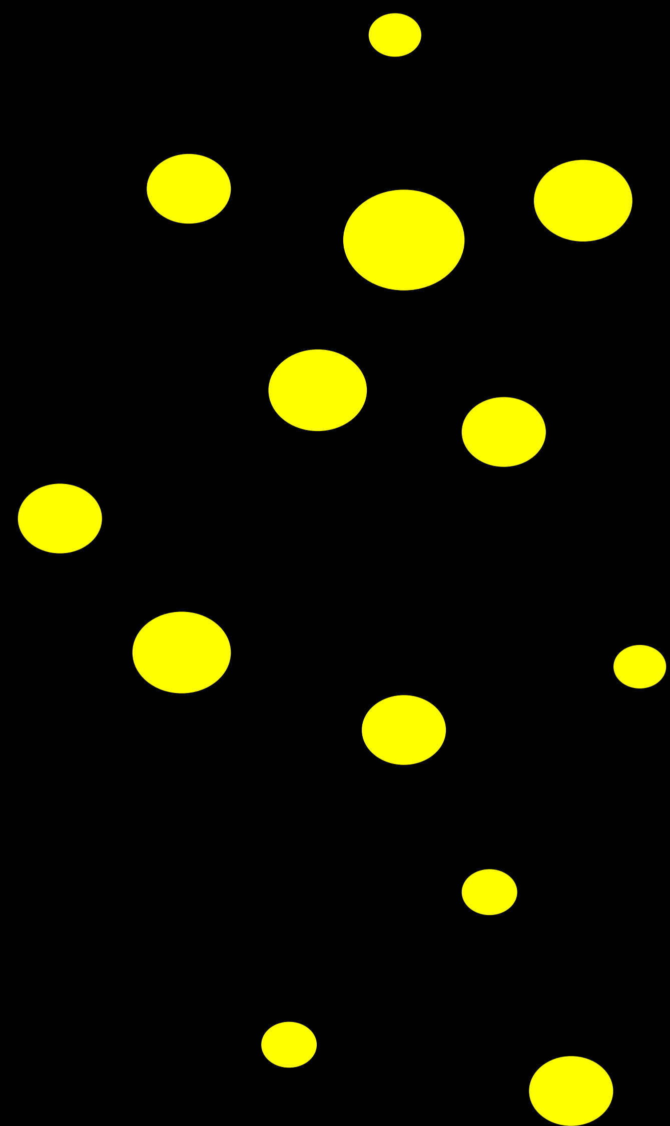


5



10





13

Today's Dataset

- `numerosity.csv`: Looking at relation of **math anxiety** to basic number skills
  - Participants log in once a day on their smartphone to complete a dot-counting task
  - And, rate their math anxiety each time
  - 30 trials (one per day for a month)
- Measures:
  - `RT` for each trial
  - `NumDots` in each display
  - Math `Anxiety` on a scale of 1 to 7

```
   Subject       NumDots           Anxiety             RT
 S1     : 30   Min.   : 16.00   Min.   :1.000   Min.   : 100
 S10    : 30   1st Qu.: 58.00   1st Qu.:1.000   1st Qu.:1682
 S11    : 30   Median : 66.50   Median :2.000   Median :2291
 S12    : 30   Mean   : 67.33   Mean   :2.236   Mean   :2323
 S13    : 30   3rd Qu.: 85.00   3rd Qu.:3.000   3rd Qu.:2962
 S14    : 30   Max.   :102.00   Max.   :6.000   Max.   :5296
 (Other):720
```

Week 7.1: Centering & Transformations

- Centering
  - Today's Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab

Interpreting Intercepts

- Let's start with examining the effect of numerosity (`NumDots`) on RT in the dot-counting task
  - At this stage, we don't care about `Anxiety`
    - Common that spreadsheet contains extra, irrelevant columns

- What would the maximal model for this be?
- `dotModel.Maximal <- lmer(RT ~`
  `1 + NumDots +`
  SUBJECT RANDOM EFFECTS
  `data = numerosity)`

- *Hint #1*: `NumDots` is a within-subjects variable

- *Hint #2:* Could there be a different effect of `NumDots` for each subject?

Interpreting Intercepts

- Let's start with examining the effect of numerosity (`NumDots`) on RT in the dot-counting task
  - At this stage, we don't care about `Anxiety`
    - Common that spreadsheet contains extra, irrelevant columns

- What would the maximal model for this be?
- ```
  dotModel.Maximal <- lmer(RT ~
  1 + NumDots +
  (1 + NumDots|Subject) +
  data = numerosity)
  ```
  - Numerosity is manipulated **within subjects** (each subject sees *several different* display sizes)
    - Possible to calculate each subject's personal NumDots effect (slope)—some subjects could count faster than othes
    - **Include** random slope
  ```

Interpreting Intercepts

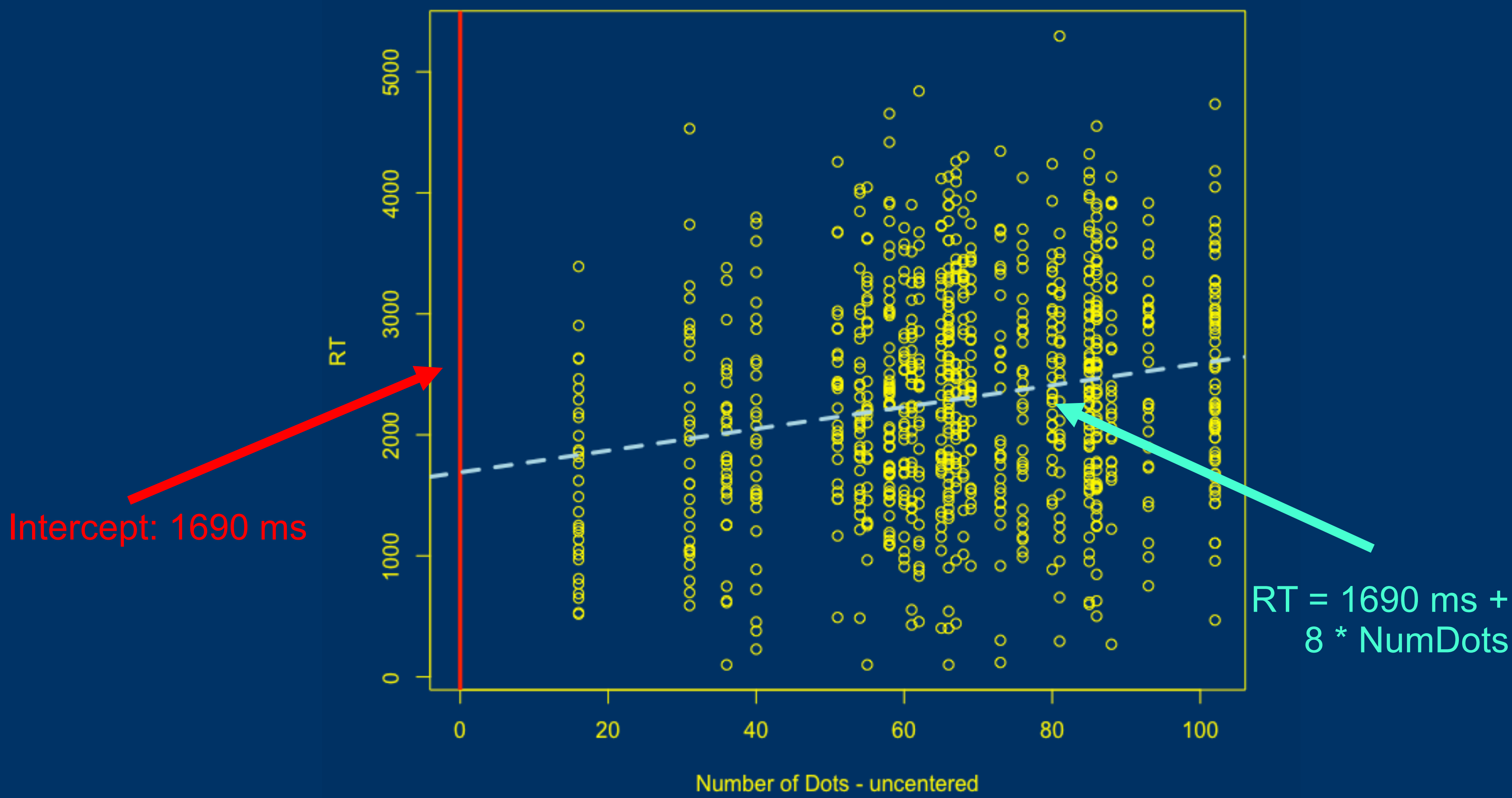
- Let's start with examining the effect of numerosity (`NumDots`) on RT in the dot-counting task
  - At this stage, we don't care about `Anxiety`
    - Common that spreadsheet contains extra, irrelevant columns
- Results:

```
Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)    
(Intercept) 1690.421    152.547  24.800  11.081 4.28e-11 ***
NumDots        9.401      1.178  27.562   7.984 1.21e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$$y = 1690 + 9.4 * \text{NumDots}$$

- Intercept: RT is 1690 ms when number of dots is 0
- NumDots effect: +9.4 ms for each dot
  - But, display size of 0 is *impossible*. Odd to talk about.

Interpreting Intercepts



- Let’s change the model so that 0 *means something*

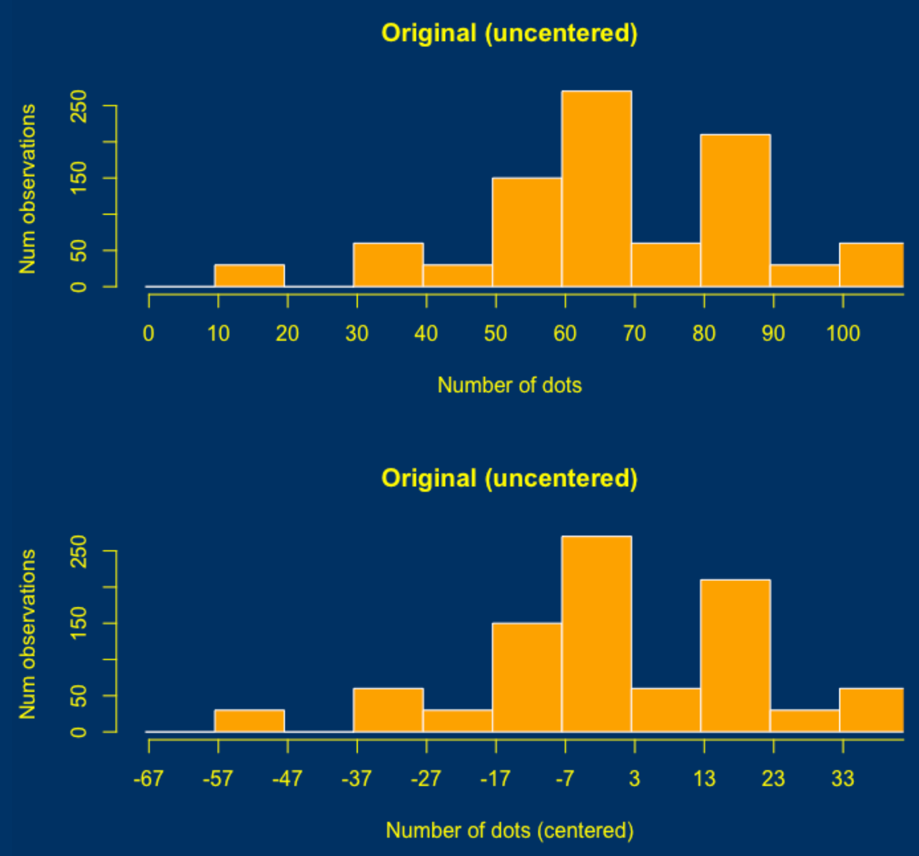
Mean Centering

- Mean number of dots is ~67
- Imagine we subtracted **<u>this mean</u>** size from **<u>each display size</u>**

ORIGINAL		MEAN SUBTRACTED
102	→	35
67	→	0
58	→	-9

- *New* **<u>zero</u>** represents **<u>mean</u>** display size
  - "**<u>Mean centering</u>**"

Mean Centering



- *New* **zero** represents **mean** display size
  - “**Mean centering**”

Centering—How to Do It

- First, create a new variable:
  - `library(misty)`
  - `numerosity %>% mutate(NumDots.cen = center(NumDots)) -> numerosity`

```
 NumDots.cen
Min.   :-51.3333
1st Qu.: -9.3333
Median : -0.8333
Mean   :  0.0000
3rd Qu.: 17.6667
Max.   : 34.6667
```

- Then, use the new variable in your model
  - ```
    dotModel.cen.Maximal <- lmer(RT ~
    1 + NumDots.cen +
    (1 + NumDots.cen|Subject) +
    data = numerosity)
    ```
    ```

Centering—Results

- Old:

```
Fixed effects:
             Estimate Std. Error      df t value Pr(>|t|)
(Intercept) 1690.421    152.547  24.800  11.081 4.28e-11 ***
NumDots        9.401      1.178  27.562   7.984 1.21e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
        (Intr)
NumDots -0.499
```

New:

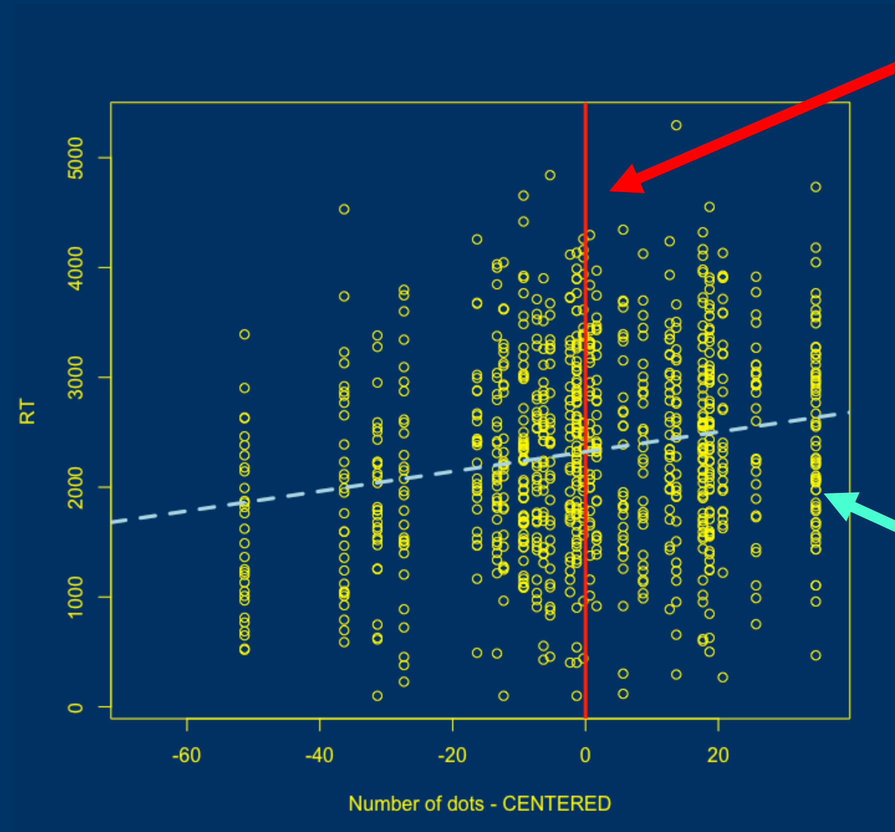
```
Fixed effects:
             Estimate Std. Error      df t value Pr(>|t|)
(Intercept) 2323.448    127.129  28.991  18.276  < 2e-16 ***
NumDots.cen    9.401      1.164  29.003   8.076 6.61e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
NumDots.cen 0.073
```

- What **hasn't** changed?
  - NumDots effect is still ~9.4 per dot
- What **has** changed?
  - Intercept is 2323 ms at *mean* numerosity
  - Correlation of NumDots effect with intercept is now almost 0. Indicates that we centered correctly.
- New model: $y = 2323 + 9.4 * \text{NumDots}$

Centering—Results

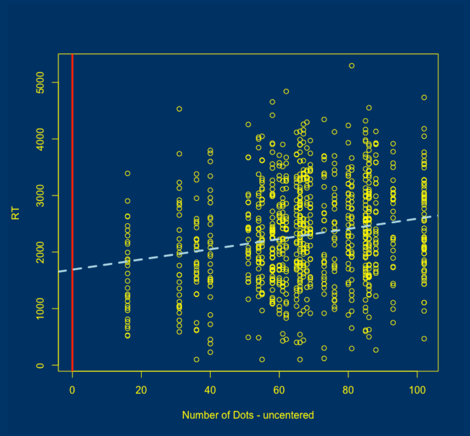
Intercept: 2094 ms



RT = 2094 ms +
8 * NumDots

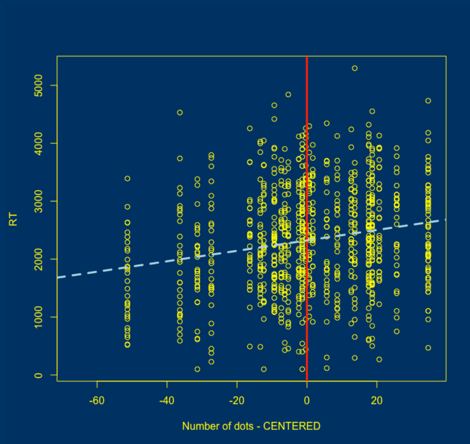
- Intercept: RT is 2094 ms at mean display size
- NumDots effect: +8 ms for each additional dot

Which Do You Like Better?



UNCENTERED

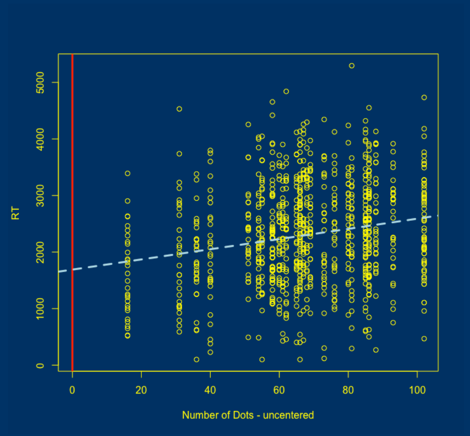
- Good if zero is *meaningful*
  - Years of study abroad, number of previous trials, number of missed classes…

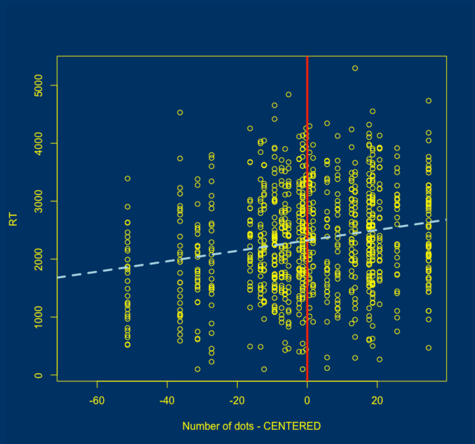


CENTERED

- Good if zero is *not meaningful* or *not observed*
  - Reduced correlation w/ intercept also helps with convergence (esp. in binomial models)

Which Do You Like Better?





- *Both* regression equations apply only to plausible number of dots
  - With raw number of dots, can’t have a numerosity of 0 or less
  - With centered number of dots, can’t have a value of -67 or less (0 minus the mean of 67)

Week 7.1: Centering & Transformations

- Centering
  - Today's Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab

Centering Around Other Values

- We could also make 0 correspond to some other sensible/useful value
  - The smallest logically *possible* value
    - ```
      numerosity %>%
      mutate(NumDots2 = center(NumDots, value=1))
      -> numerosity
      ```
  - The smallest *observed* value in our data
    - ```
      numerosity %>%
      mutate(NumDots3 = center(NumDots,
       value=min(NumDots)) -> numerosity
      ```

```
    NumDots2
Min.   : 15.00
1st Qu.: 57.00
Median : 65.50
Mean   : 66.33
3rd Qu.: 84.00
Max.   :101.00
```

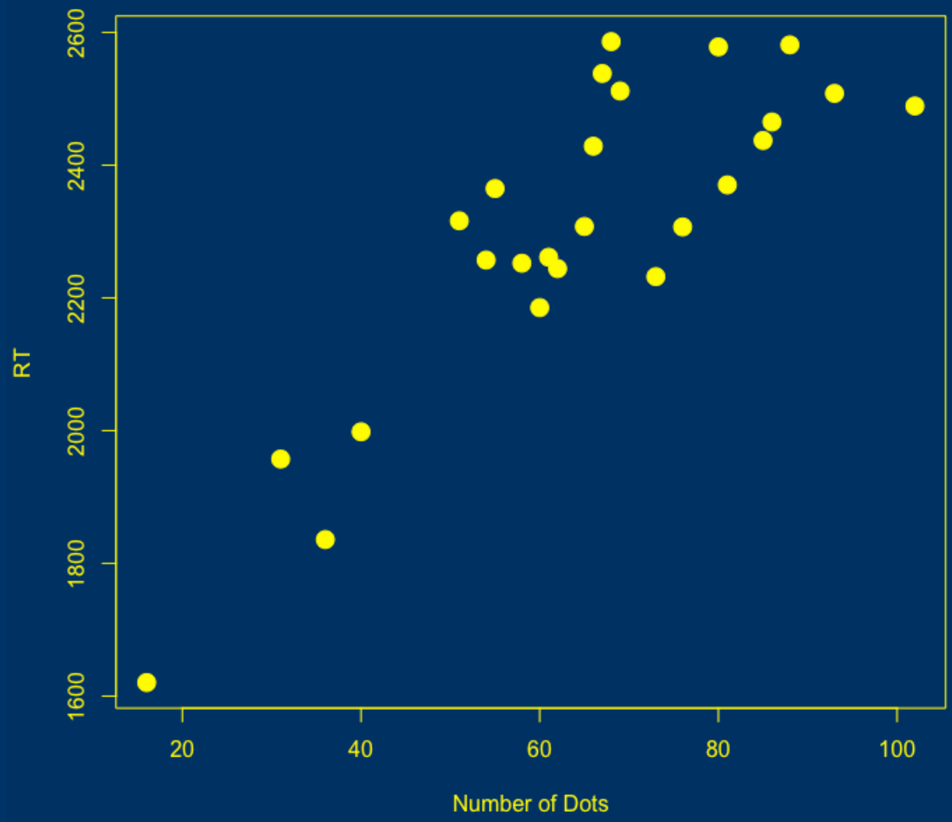
```
    NumDots3
Min.   : 0.00
1st Qu.:42.00
Median :50.50
Mean   :51.33
3rd Qu.:69.00
Max.   :86.00
```

Week 7.1: Centering & Transformations

- Centering
  - Today's Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab

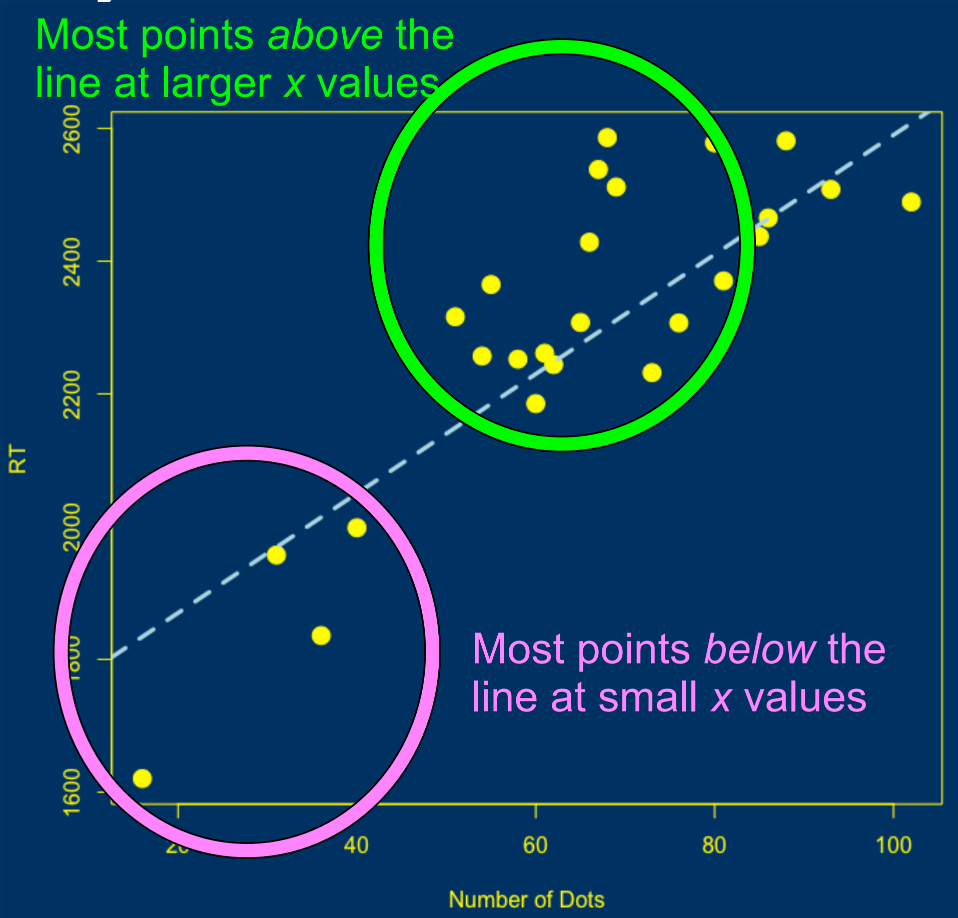
Logarithmic Transformations

- Let's look more closely at the relation between dots and RT…
  - Not totally linear…



Logarithmic Transformations

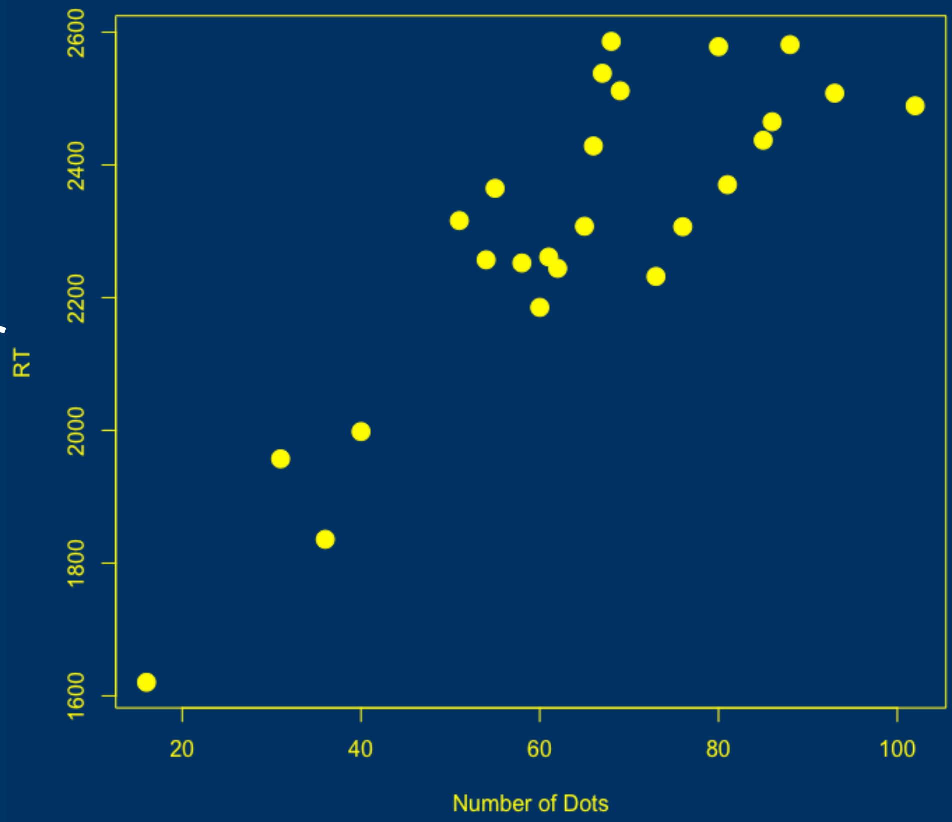
- Let's look more closely at the relation between dots and RT…
  - Not totally linear…



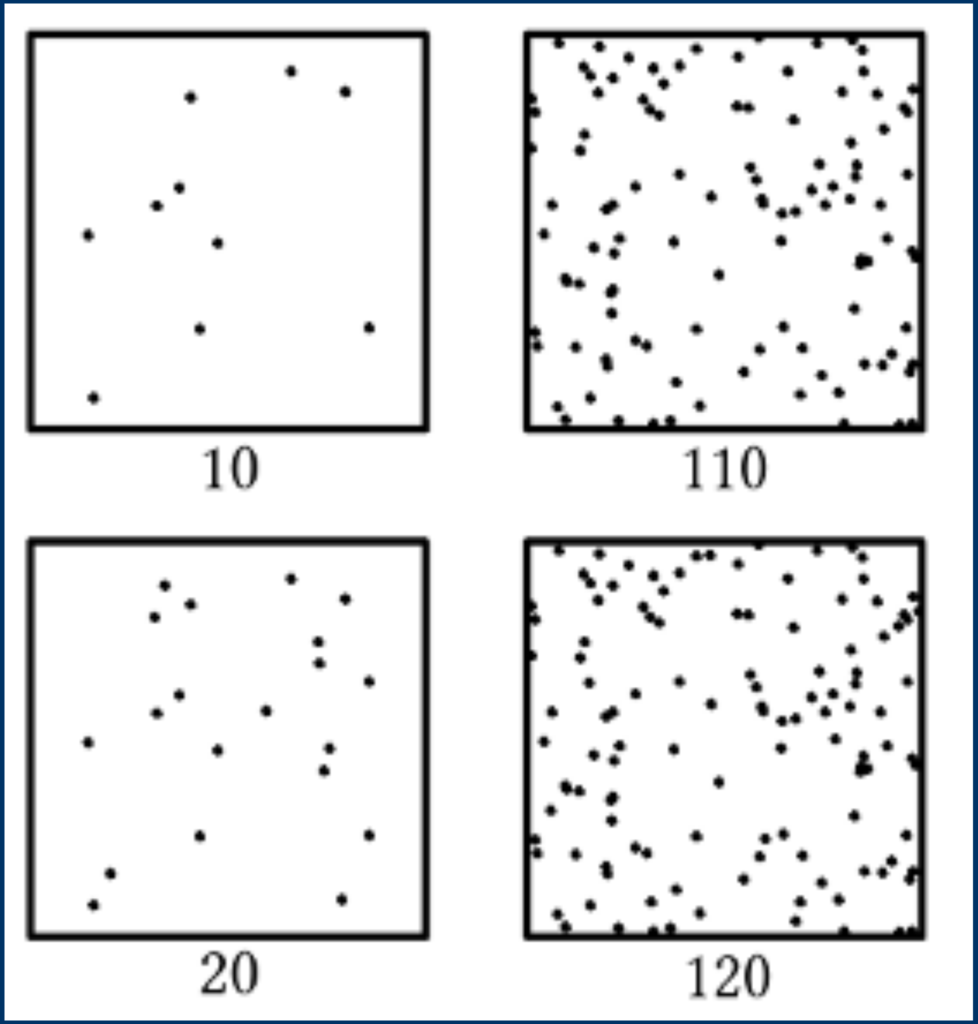
Logarithmic Transformations

- Let's look more closely at the relation between dots and RT…
  - Not totally linear…

- Effect "levels off"
  - Effect of adding 1 more dot is smaller when there are already a lot of dots
  - "Diminishing returns"



Logarithmic Transformations



Fechner's law

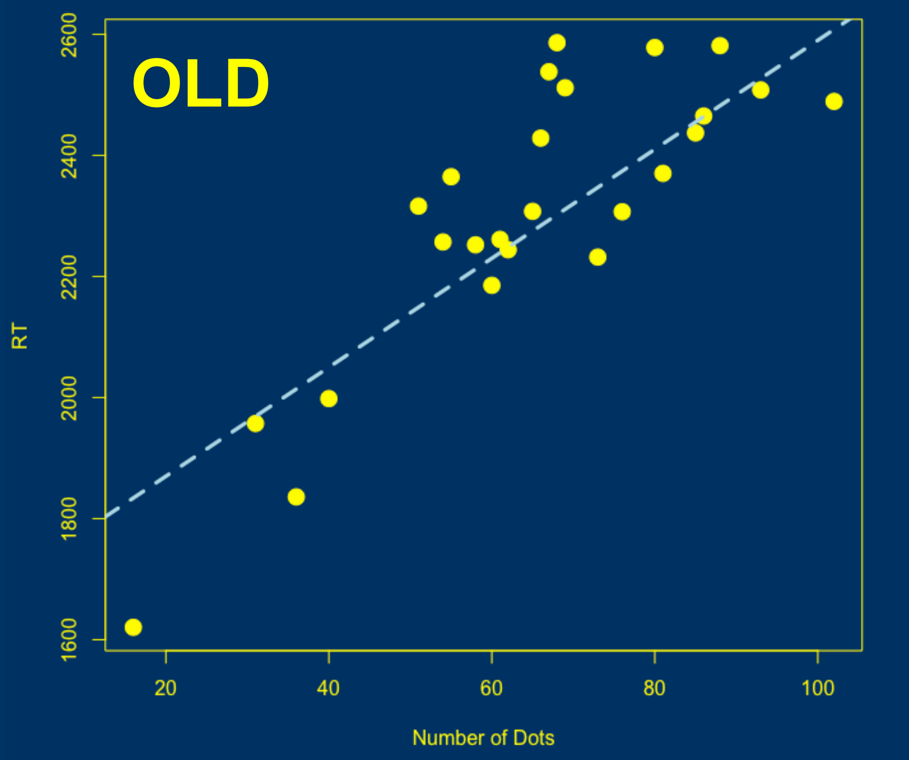
Logarithmic Transformations

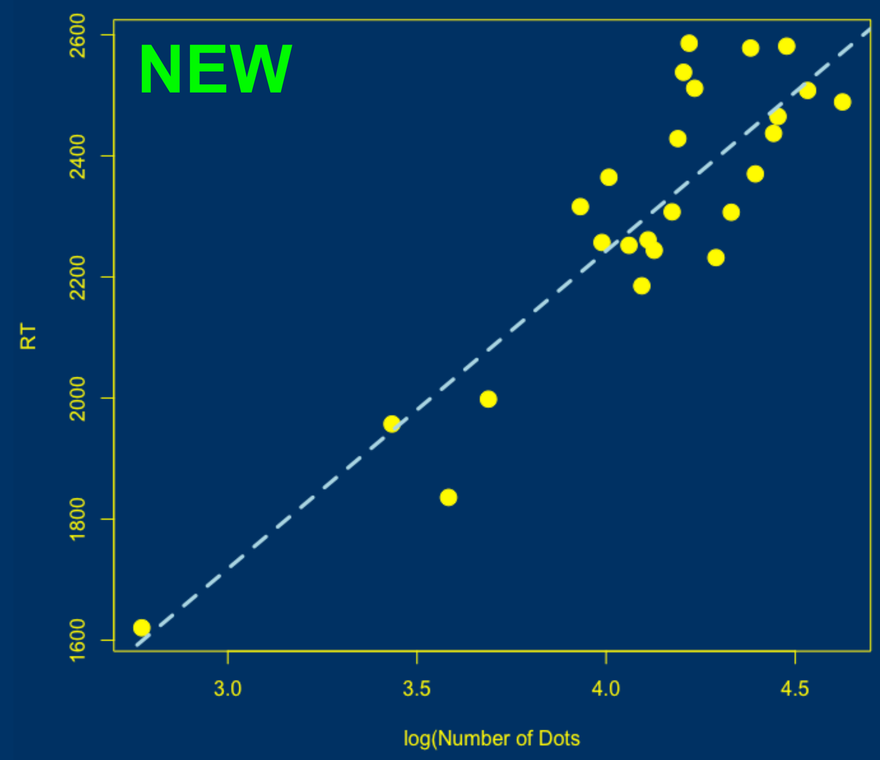
- This type of non-linear relationship is well modeled by a **logarithmic** transformation

  - ```
    numerosity %>%
    mutate(NumDots.log = log(NumDots)) ->
    numerosity
    ```
    ```

Logarithmic Transformations

- This *looks* more linear
- How can we compare the fit of these models?
  - Hint: We subtracted NumDots *and* added NumDots.log, so these are *not* nested models





Logarithmic Transformations

- This *looks* more linear
- How can we compare the fit of these models?
  - Hint: We subtracted NumDots *and* added NumDots.log, so these are *not* nested models

- `anova(dotModel.Maximal, dotModel.log.Maximal)`

	npar	AIC	BIC	logLik	deviance
dotModel.Maximal	6	14066	14095	-7027.2	14054
dotModel.log.Maximal	6	14053	14082	-7020.7	14041

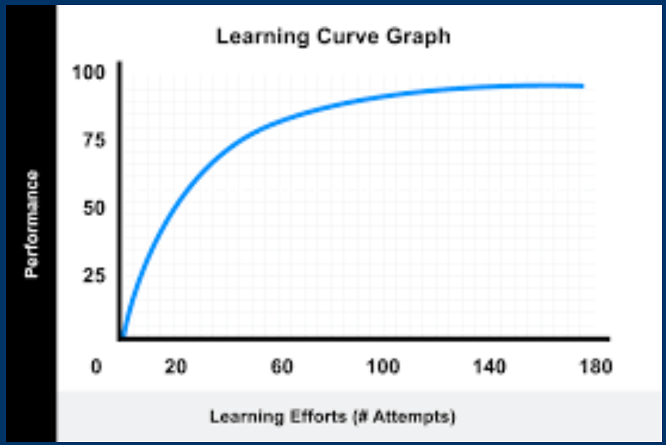
Model with log(NumDos) has lower (better) AIC and BIC

Logarithmic Transformations

- Logarithmic relationships pervasive
  - Many aspects of perception (Fechner and Weber laws)
  - Many aspects of experience
  - Dose-response
  - Income

Source	Intensity	Intensity level	× TOH
Threshold of hearing (TOH)	10^{-12}	**0 dB**	1
Whisper	10^{-10}	**20 dB**	10^{2}
Pianissimo	10^{-8}	**40 dB**	10^{4}
Normal conversation	10^{-6}	**60 dB**	10^{6}
Fortissimo	10^{-2}	**100 dB**	10^{10}
Threshold of pain	10	**130 dB**	10^{13}
Jet take-off	10^{2}	**140 dB**	10^{14}
Instant perforation of eardrum	10^{4}	**160 dB**	10^{16}

Table 1.1 from [Müller, FMP, Springer 2015]



Logarithmic Transformations

- Logarithmic relationships pervasive
  - Many aspects of perception (Fechner and Weber laws)
  - Many aspects of experience
  - Dose-response
  - Income

Source	Intensity	Intensity level	× TOH
Threshold of hearing (TOH)	10^{-12}	**0 dB**	1
Whisper	10^{-10}	**20 dB**	10^{2}
Pianissimo	10^{-8}	**40 dB**	10^{4}
Normal conversation	10^{-6}	**60 dB**	10^{6}
Fortissimo	10^{-2}	**100 dB**	10^{10}
Threshold of pain	10	**130 dB**	10^{13}
Jet take-off	10^{2}	**140 dB**	10^{14}
Instant perforation of eardrum	10^{4}	**160 dB**	10^{16}

Table 1.1 from [Müller, FMP, Springer 2015]

- If you want to center, apply the log transform first, *then* center
  - Otherwise, the final variable will not be centered
  - Idea is that the "true" variable has a log scale, so we want to center around *that* mean

Logarithmic Transformations

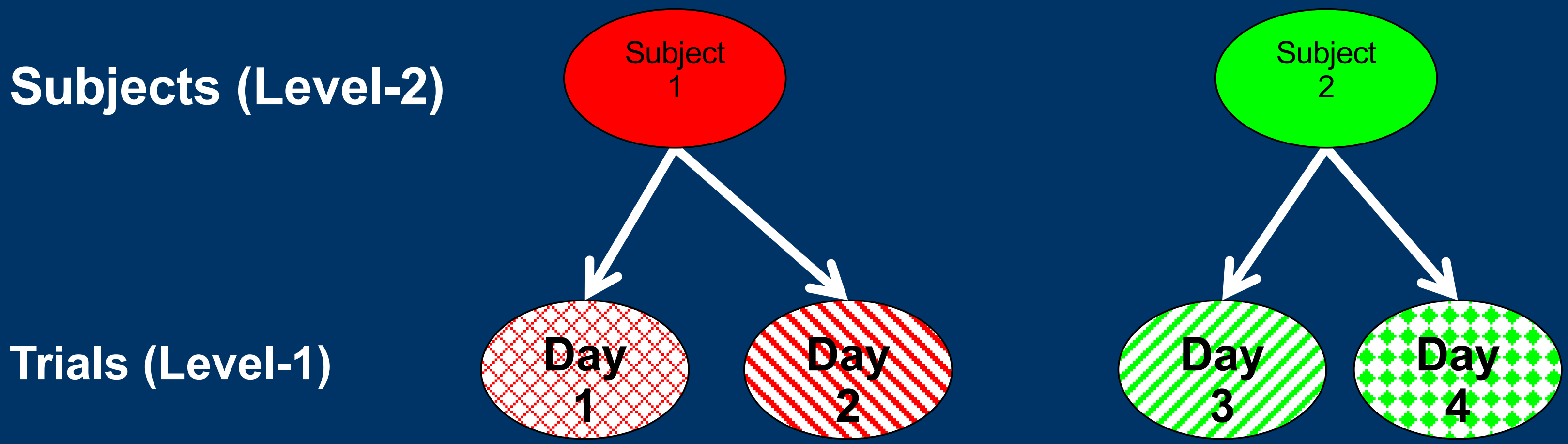
- When to apply a transformation?
  - We have an *a priori* reason to expect a logarithmic (or other) relationship
    - e.g., some well-studied variable, like loudness or word frequency
  - Empirically based on the observed relationship to the DV
    - But beware of overfitting!

Week 7.1: Centering & Transformations

- Centering
  - Today's Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab

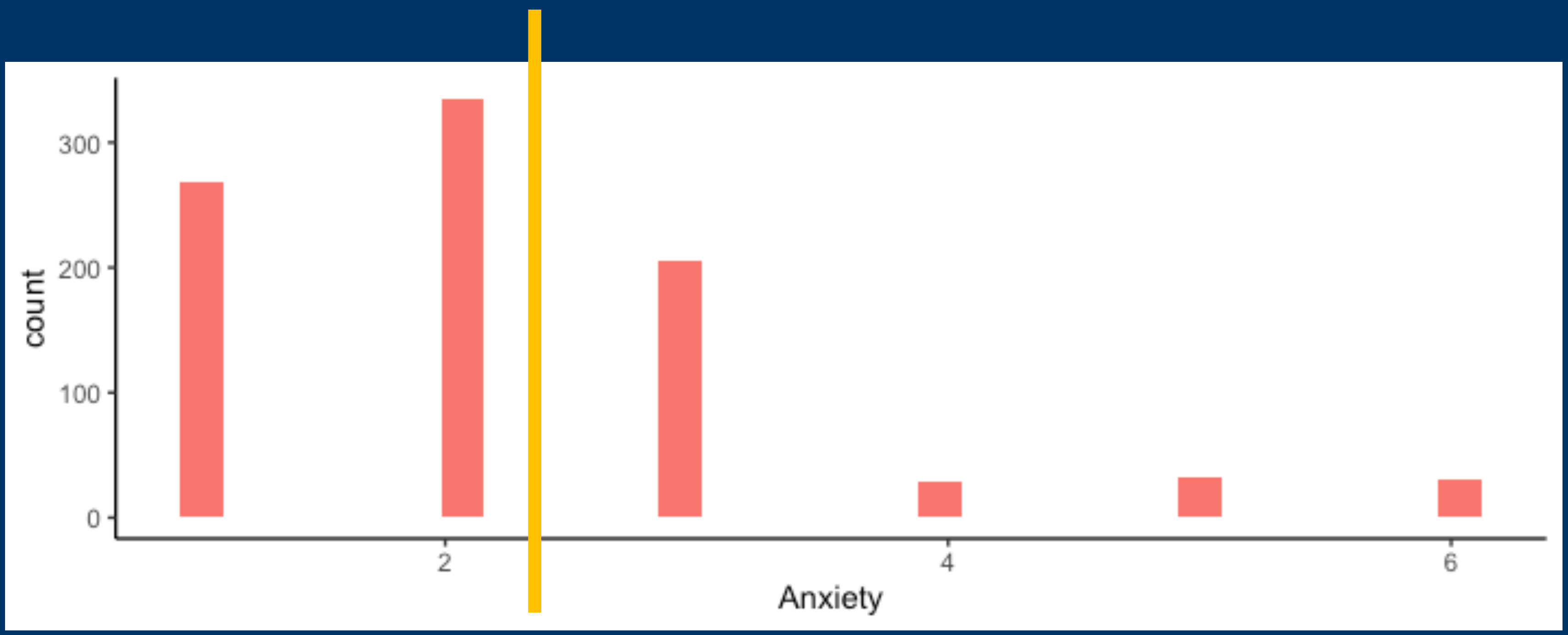
Grand-Mean vs. Cluster-Mean Centering

- Let's turn to the effects of math anxiety
  - Remember that people provided a different `Anxiety` rating on each of the 30 days (trials)
  - Is this a level-1 or level-2 variable?



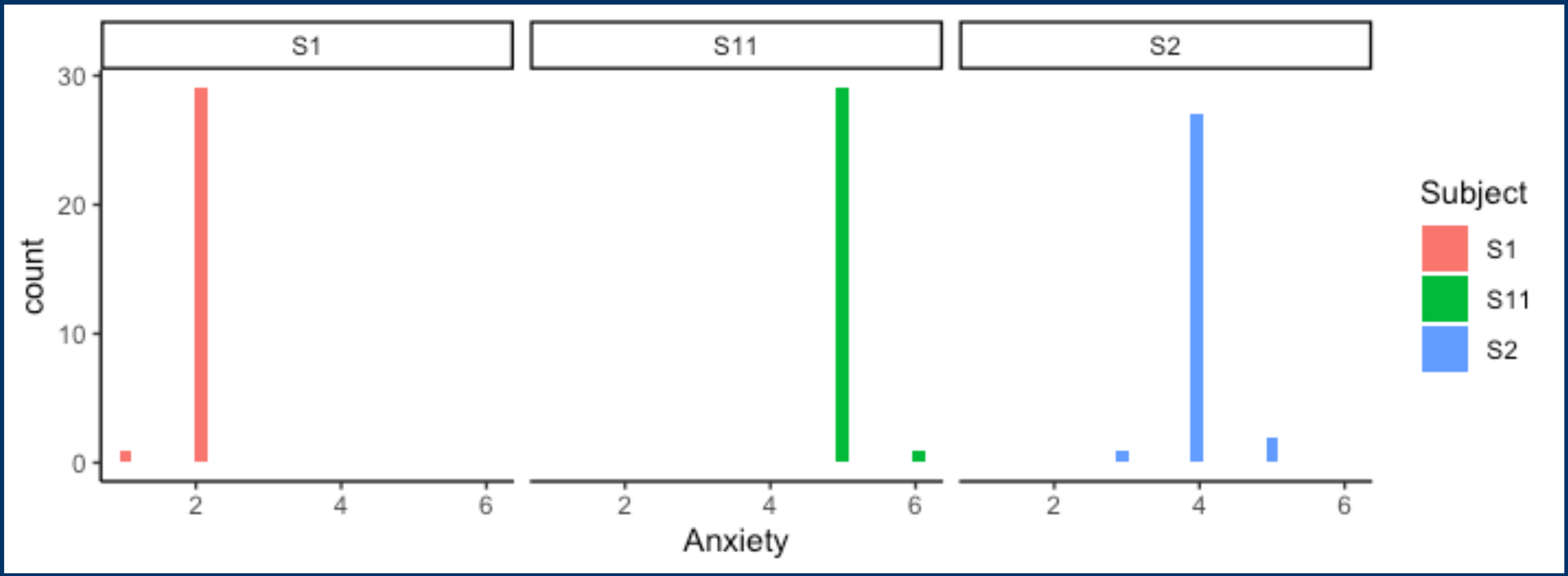
Grand-Mean vs. Cluster-Mean Centering

- What mean to use to center a level-1 variable?
  - Grand mean?: 2.23



Grand-Mean vs. Cluster-Mean Centering

- What mean to use to center a level-1 variable?
  - Grand mean?: 2.23
  - But individual subject means differ a lot…
  - Subject 1: 1.97, Subject 11: 5.03, Subject 2: 4.03



Grand-Mean vs. Cluster-Mean Centering

- *Both* sources of variation may be relevant
  - **Grand-mean centering:** Are you more or less math-anxious compared to the average person (2.23)?
    - **Trait anxiety**
  - **Cluster-mean centering:** Are you more or less math-anxious *on this day* compared to *your* average (for subject 2: 4.03)?
    - **State anxiety**

Grand-Mean vs. Cluster-Mean Centering

- Centering a level-1 variable "smushes" together these two sources of variance (Hoffman, 2015, 2019)
  - How the cluster differs from the grand mean
  - How the observation differs from the cluster mean
- A better approach is to track these two influences *separately*
  - `numerosity %>% mutate(TraitAnxiety = center(Anxiety, type='CGM', group=Subject)) -> numerosity`
    - Center around the grand mean (a level-2 variable)
  - `numerosity %>% mutate(StateAnxiety = center(Anxiety, type='CWC', group=Subject)) -> numerosity`
    - Center around the cluster mean (a level-1 variable)

Grand-Mean vs. Cluster-Mean Centering

- Then, include *both* of these in the model

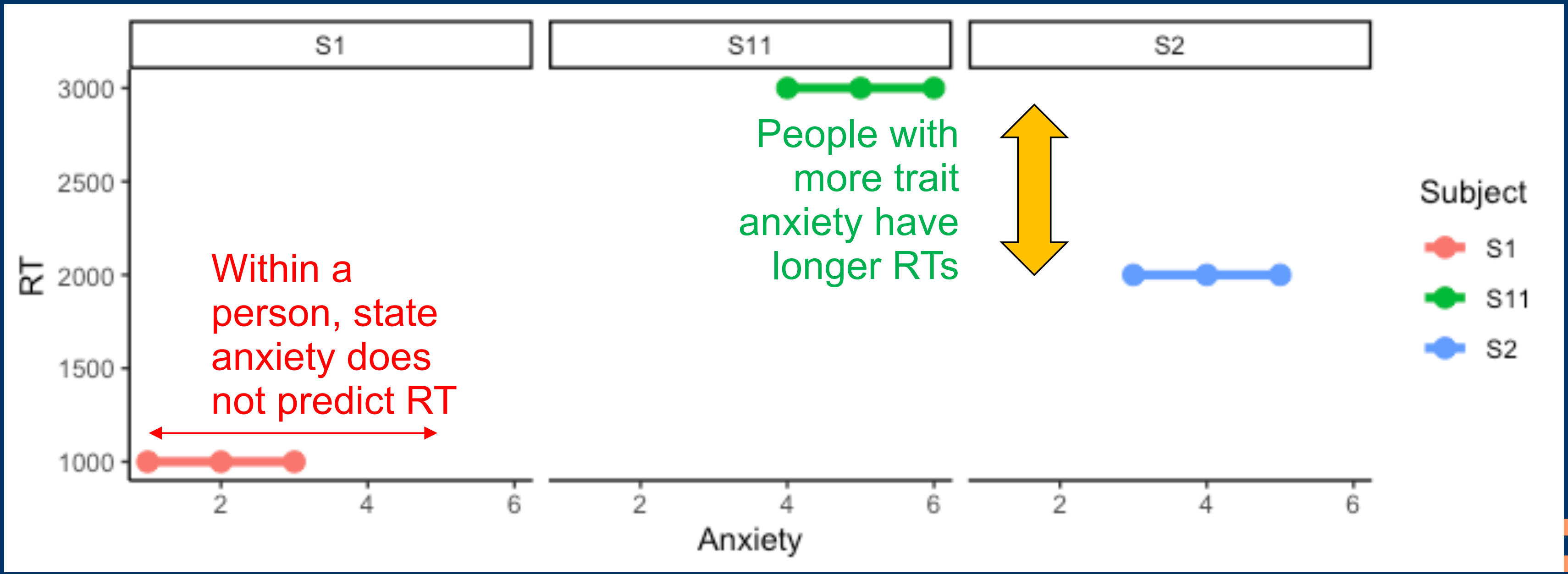
- `model.Anxiety <- lmer(RT ~ 1 + TraitAnxiety + StateAnxiety + (1|Subject), data=numerosity)`
- `model.Anxiety %>% summary()`

```
Fixed effects:
             Estimate Std. Error     df t value Pr(>|t|)
(Intercept)   2323.45     129.16  28.02  17.989   <2e-16 ***
StateAnxiety   235.59     156.96  99.22   1.501    0.137
TraitAnxiety   -33.55     107.72  28.05  -0.311    0.758
```

- Can also include a random slope for the level-1 variable (StateAnxiety)

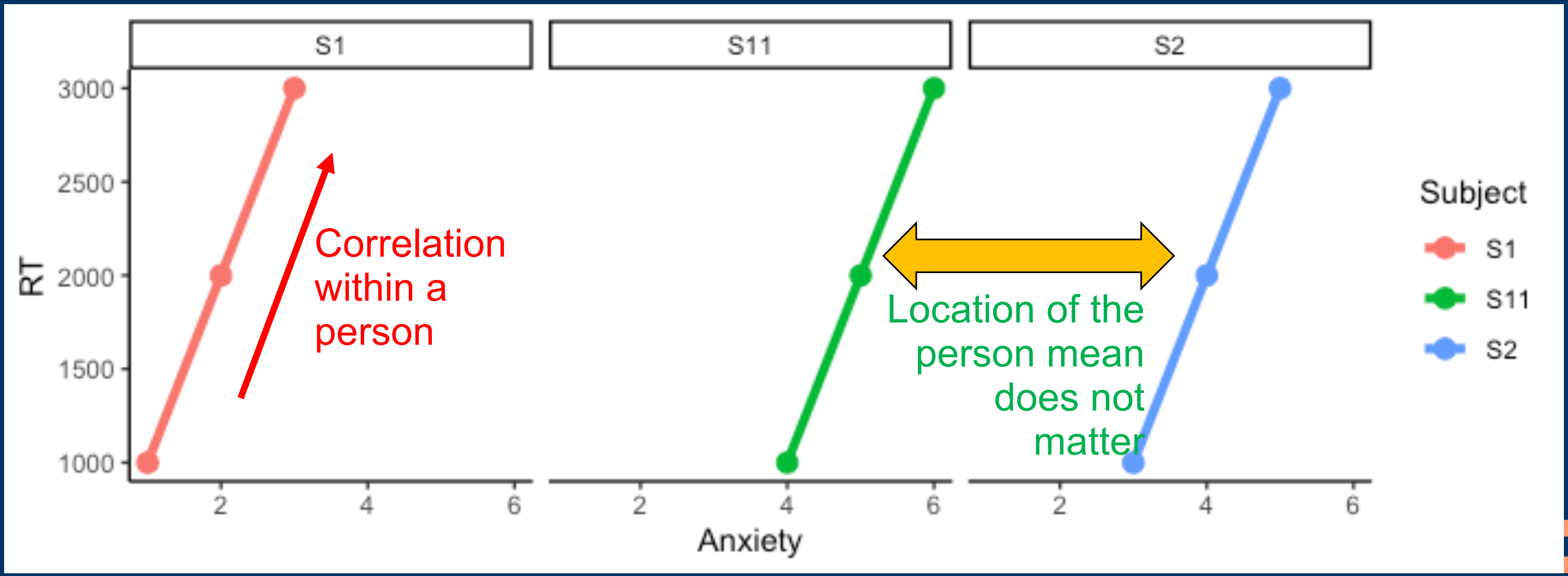
Grand-Mean vs. Cluster-Mean Centering

- Possible to have one effect, but not the other
  - Trait effect but not a state effect
    - Could reflect another individual difference (math skill?)



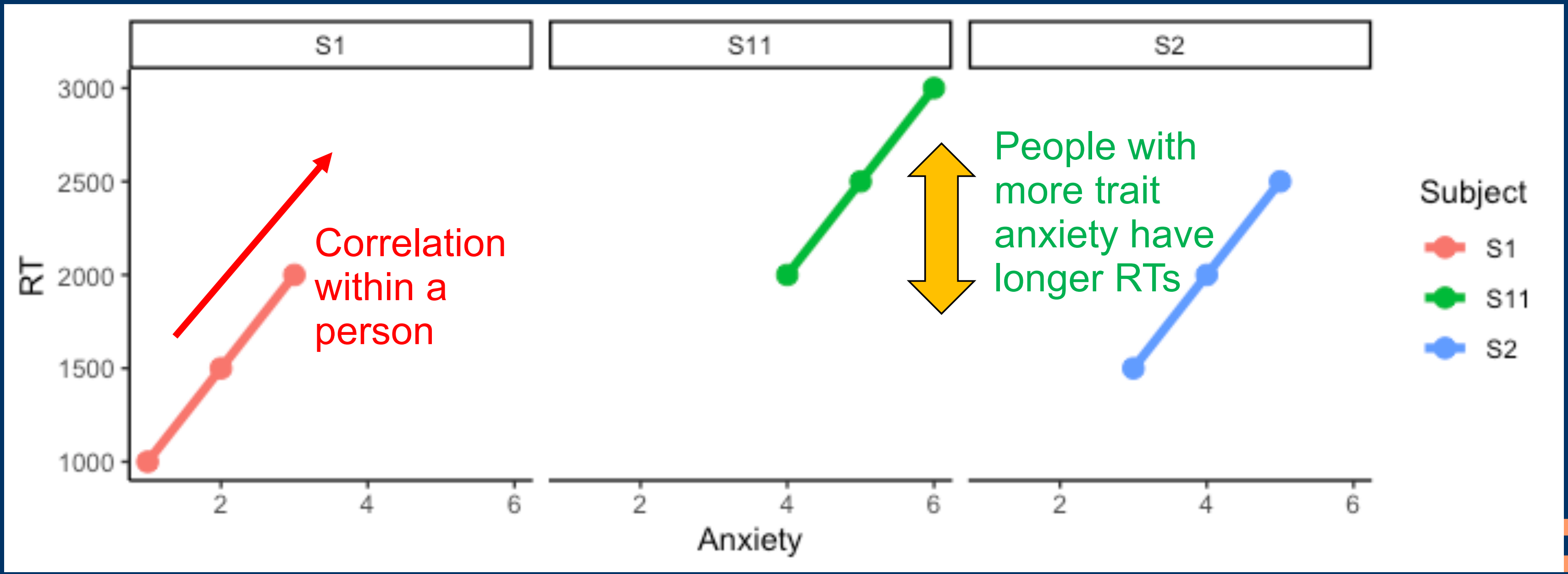
Grand-Mean vs. Cluster-Mean Centering

- Possible to have one effect, but not the other
  - Trait effect but not a state effect
    - Could reflect another individual difference (math skill?)
  - State effect but not a trait effect
    - What matters is where you are relative to your norms



Grand-Mean vs. Cluster-Mean Centering

- Possible to have one effect, but not the other
  - Trait effect but not a state effect
    - Could reflect another individual difference (math skill?)
  - State effect but not a trait effect
    - What matters is where you are relative to your norms
  - Or **both**!



Grand-Mean vs. Cluster-Mean Centering

- Conclusions:
  - For level-1 variables, want to consider both **between-** and **within-** cluster ariation
  - This does not matter:
    - For level-2 variables (e.g., no within-person slope for IQ or math class GPA)
      - *No* within-cluster variance
    - Every cluster has the same mean (e.g., experimental manipulation like `NumDots`)
      - *No* between-cluster variance

Week 7.1: Centering & Transformations

- Centering
  - Today's Dataset
  - Mean Centering
  - Centering Around Other Values
  - Logarithmic Transformation
  - Grand-Mean vs. Cluster-Mean Centering
- Lab