Statistics with R

Linear Regression

Zhuanghua Shi (Strongway)

11 June 2018

Linear regression

- A continuous dependent variable Y
- One of more independent variables X

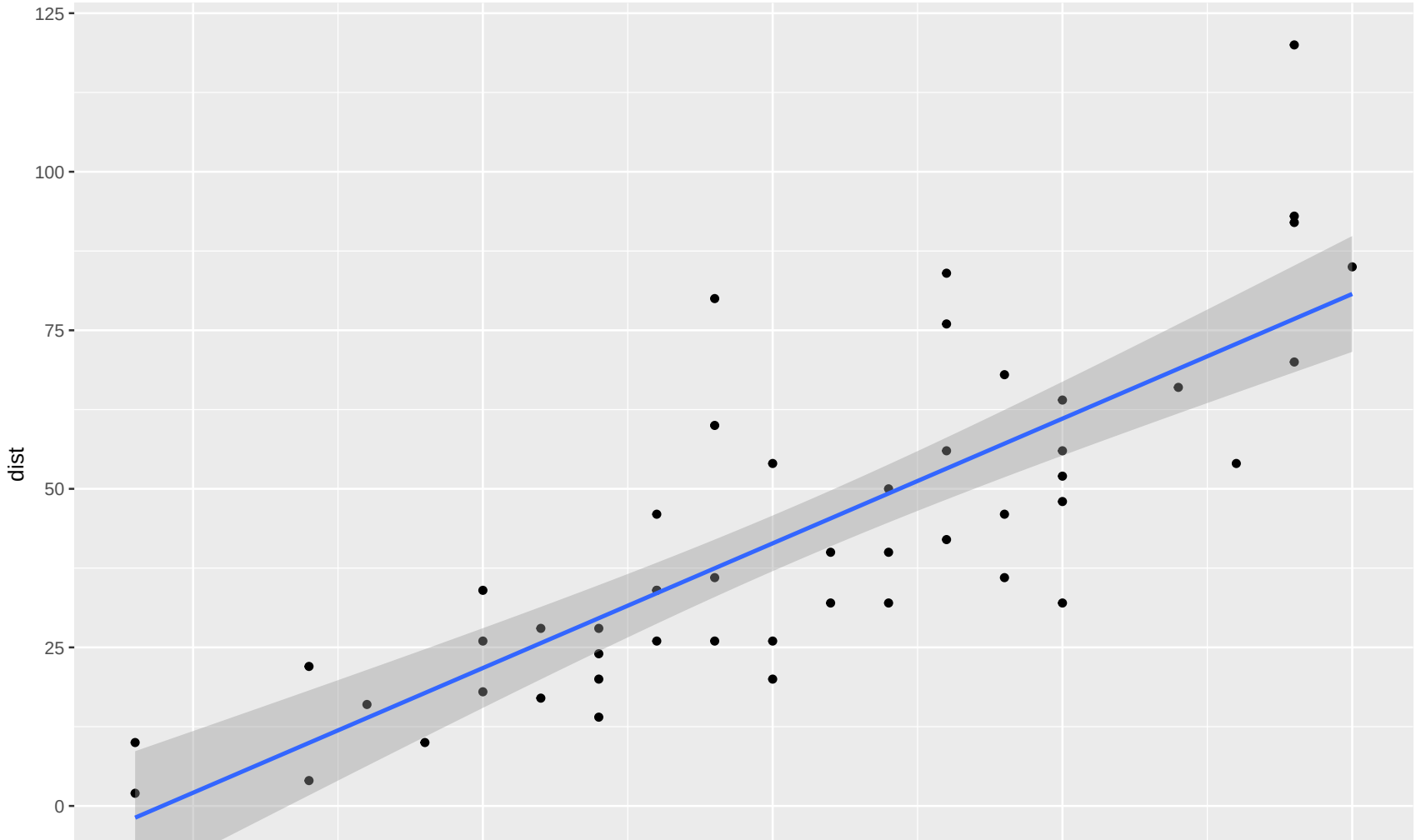
We want to estimate their linear relationship:

$$Y = b_0 + b_1 X + \epsilon$$

An example

- build-in `cars` dataset

```
ggplot(cars, aes(speed, dist)) + geom_point() +
  geom_smooth(method = 'lm')
```



Build linear model

- Linear regression uses function `lm()`
- `lm()` accept
  - formula
  - data

```
mod1 = lm(dist ~ speed, data = cars)
mod1
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Coefficients:
## (Intercept)        speed
##     -17.579        3.932
```

Get model summary

- summary(model) provides residuals, coefficients, statistics, significances, R-squared, F-tests

```
summary(mod1)
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601   0.0123 *
## speed         3.9324     0.4155   9.464 1.49e-12 ***
## ---
```

Goodness of fit

- How do we know if the model is good fit?
  - R-Square: higher the better (>0.7)
  - t-statistics: p-value should be less than 0.05

Prediction of the model

- `predict(model, testData)`
  - note: testData shold contain the IV variables

```
p_dist = predict(mod1, data.frame(speed = c(20,21)))
print(p_dist)
```

```
##        1        2
## 61.06908 65.00149
```

R formula

Recall the last session:

$$Y \sim X1 + X2 + 1$$

$$Y \sim X1 * X2$$

$$Y \sim X1 - 1$$

Example of model comparison

- Data of Thibault et al. (2007) motion sensitity as a function of age, movement type (First/Second order motion), Sex

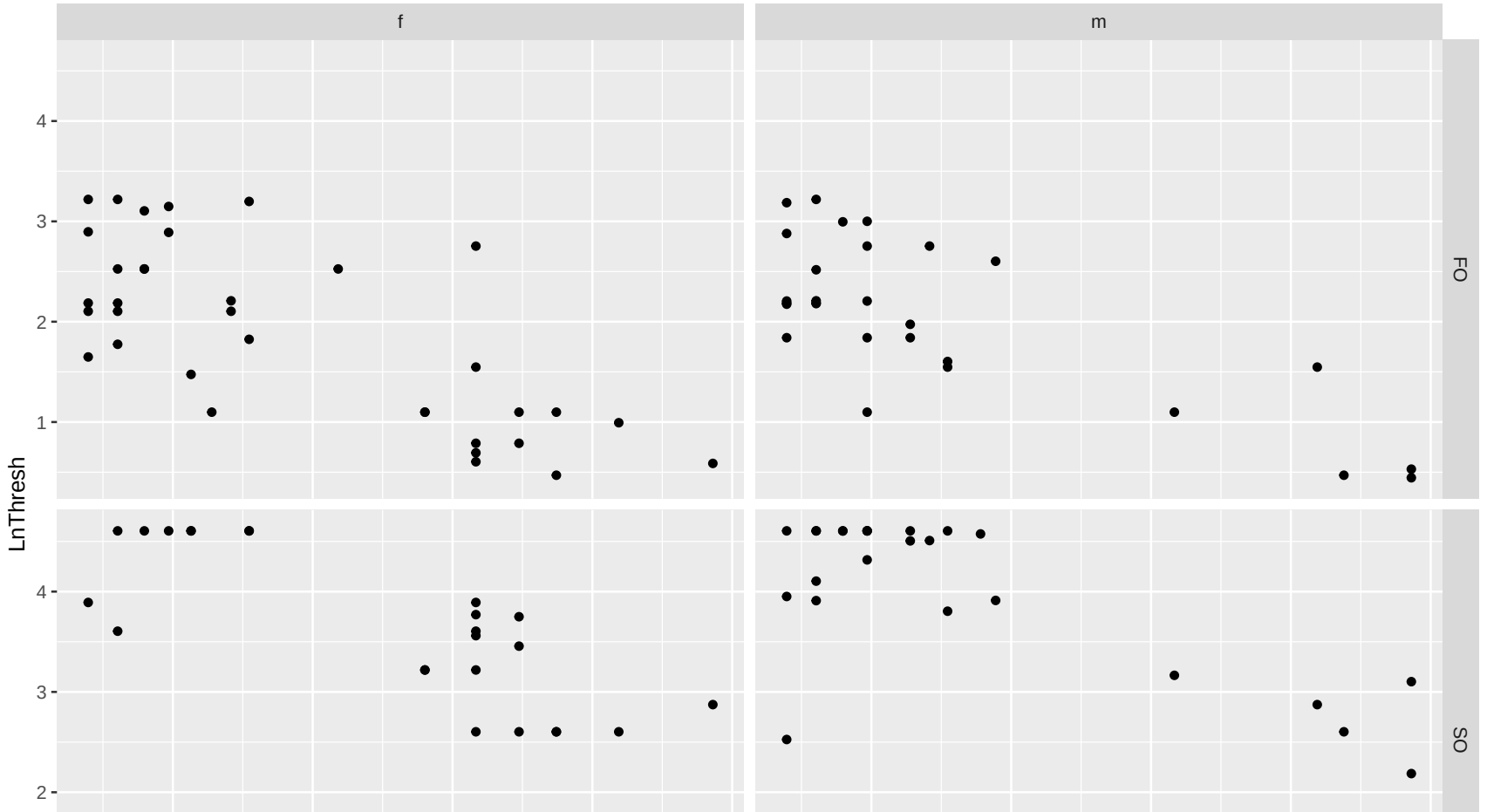
```
motion = read.csv('motion.csv')
head(motion,3)
```

```
##   Subject    LnAge Mtype Sex LnThresh
## 1     S01 2.197225    FO   f 3.218876
## 2     S02 2.197225    FO   f 2.186051
## 3     S03 2.197225    FO   m 2.186051
```

Explore the data

- Explore the relation between the age and motion threshold
  - Separate for motion type and gender

```
motion %>% ggplot(aes(LnAge, LnThresh)) + geom_point() +
  facet_grid(Mtype~Sex)
```



Build linear models

- Only main factors

```
motion_mod1 = lm(LnThresh ~ Mtype + Sex + LnAge, data = mot
coef(summary(motion_mod1))
```

```
##                Estimate Std. Error     t value     Pr(>
## (Intercept)  4.46983546 0.23730796  18.8355901 6.537622e
## MtypeSO      2.04037959 0.10424491  19.5729427 2.622696e
## Sexm        -0.01365394 0.10469116  -0.1304211 8.964759e
## LnAge       -0.88221879 0.07492058 -11.7753867 4.343605e
```

Build linear models

- Main factors and two-way interaction

```
motion_mod2 = lm(LnThresh ~ Mtype + Sex + LnAge + Mtype:Sex
                      Mtype:LnAge + Sex:LnAge, data = motion)
coef(summary(motion_mod2))
```

```
##                    Estimate Std. Error     t value      Pr
## (Intercept)    4.697858368  0.3731837 12.58859509 1.0516
## MtypeSO        1.913781159  0.5041989  3.79568674 2.4665
## Sexm          -0.340101768  0.4535632 -0.74984434 4.5502
## LnAge         -0.960239056  0.1236195 -7.76770005 5.6284
## MtypeSO:Sexm  -0.009681203  0.2181419 -0.04438031 9.6468
## MtypeSO:LnAge  0.046960621  0.1540333  0.30487326 7.6106
## Sexm:LnAge     0.114031974  0.1526832  0.74685341 4.5682
```

Compare two models

- Using ANOVA method to compare the two models, with and without the second-order interaction

```
anova(motion_mod1, motion_mod2)
```

```
## Analysis of Variance Table
##
## Model 1: LnThresh ~ Mtype + Sex + LnAge
## Model 2: LnThresh ~ Mtype + Sex + LnAge + Mtype:Sex + Mt
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    108 31.458
## 2    105 31.251  3   0.20735 0.2322 0.8738
```

Remove insig. factor

- Sex is not a critical factor, shown by the t-test

```
motion_mod3 = lm(LnThresh ~ Mtype +  LnAge, data = motion)
coef(summary(motion_mod3))
```

```
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  4.4579372 0.21808119  20.44164 4.353848e-39
## MtypeSO      2.0390853 0.10330246  19.73898 8.823760e-38
## LnAge       -0.8801681 0.07292107 -12.07015 8.182345e-22
```

Compare models

- Compared with and without the factor `Sex`
  - Removal of `Sex` did not make any difference

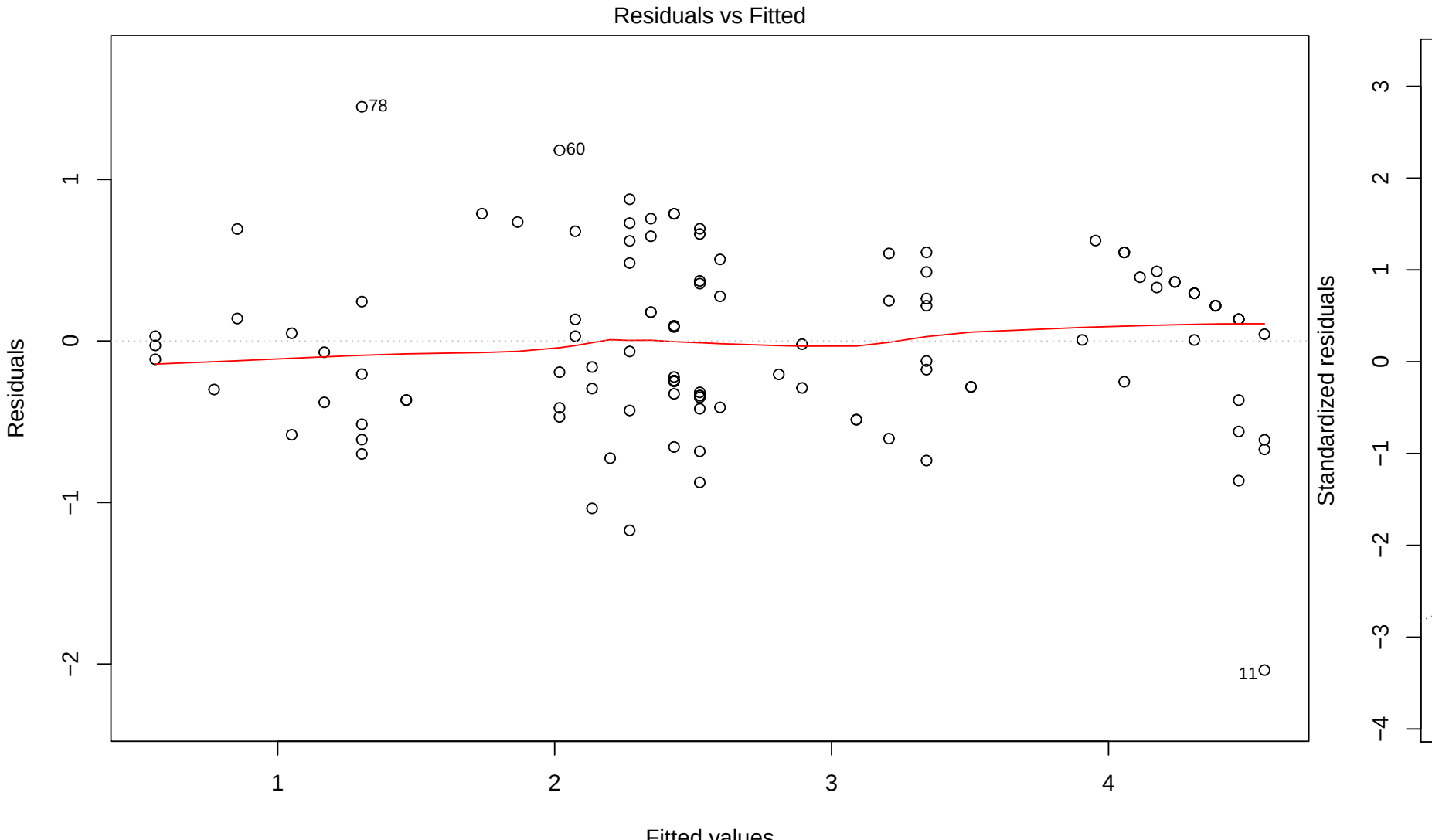
```
anova(motion_mod1, motion_mod3)
```

```
## Analysis of Variance Table
##
## Model 1: LnThresh ~ Mtype + Sex + LnAge
## Model 2: LnThresh ~ Mtype + LnAge
##   Res.Df    RSS Df  Sum of Sq     F Pr(>F)
## 1    108 31.458
## 2    109 31.463 -1 -0.0049546 0.017 0.8965
```

Visualize the best model

- `lm` method provide direct `plot` for diagnose

```
plot(motion_mod3)
```



Linear model and tidyverse

- Example - Modelfest data from Watson & Ahumada (2005)
  - Foveal detection of spatial contrast
  - http://jov.arvojournals.org/5/9/6
- Using RStudio 'File - Import Dataset' to import

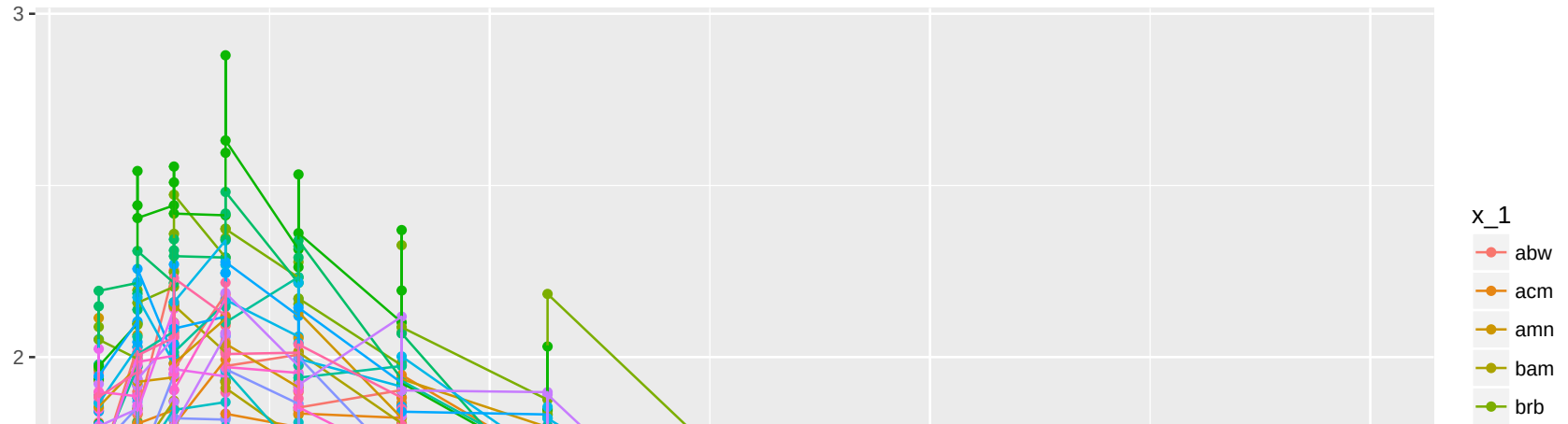
```
library(readxl)
url <- "http://jov.arvojournals.org/data/Journals/JOV/93283
destfile <- "modelfestbaselinedata.xls"
curl::curl_download(url, destfile)
ModelFest <- read_excel(destfile, col_names = FALSE) %>%
  clean_names() # from janitor package, make names accessi
```

Tidy the data

```
library(tidyr)
dat = gather(ModelFest,, threshold, -x_1)
dat$stim = rep(1:43, each = 4*16) # 43 stimuli, 4 repetitio
# first 10 stimuli spatial frequency
SpatFreq <- c(1.12, 2^seq(1, 4.5, 0.5), 30)

# only analysis for the first 10 stimuli
dat %>% filter(stim <=10) %>% mutate(freq = SpatFreq[stim])

ggplot(dat1, aes(freq, threshold, color = x_1, group = x_1)
  geom_point() + geom_line()
```



some useful tricks get model parameters

- code from moderndive.com

```
library(broom)
library(janitor)
model %>% tidy(conf.int = TRUE) %>%
  mutate_if(is.numeric, round, digits = 3) %>%
  clean_names()
```

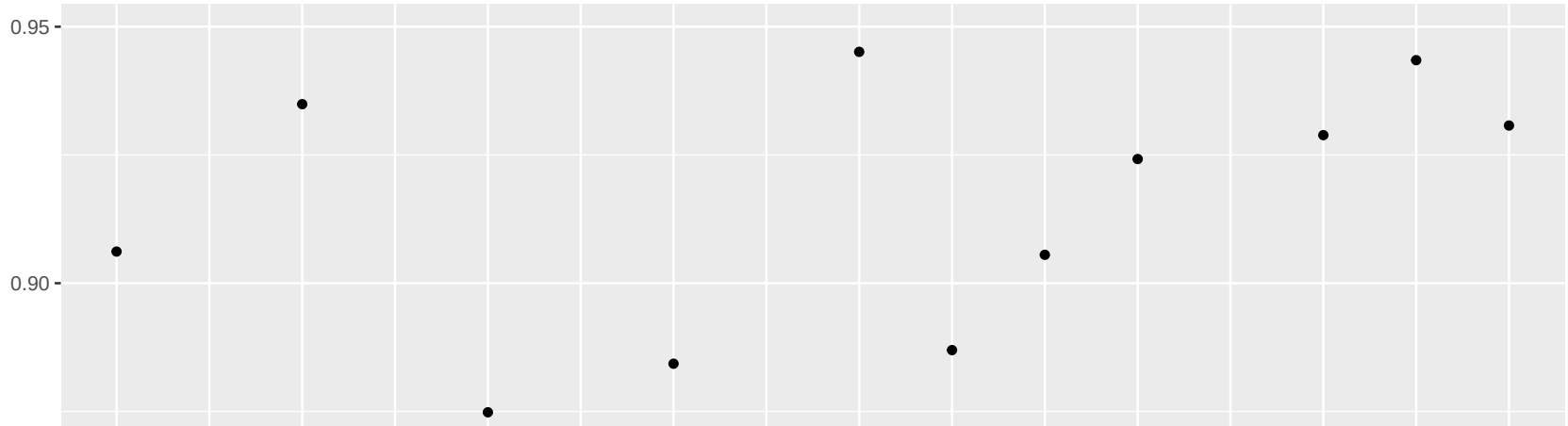
- glance()

Model with multiple subsets

```
lmod <- function(df){
  lmfit = lm(threshold ~ freq, data = df)
  return(lmfit)
}

dat1 %>% group_by(x_1) %>% nest() %>%
  mutate(mod = map(data, lmod)) %>%
  mutate(glance = map(mod, broom::glance)) %>%
  unnest(glance) -> gm

ggplot(gm, aes(x_1, r.squared)) + geom_point()
```



References

Some contents from this session are from:

1. Knoblauch & Maloney, Modeling Psychophysical Data in R, 2012
2. An introduction to statiscal and Data Science via R, Moderndive.com