Multiple Linear Regression

CSU Chico, Math 314

2018-12-05

outline

outline

Recap, simple linear regression

Simple linear regression attempts to predict the response variable Y using a linear model on the explanatory variable X

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma).$$

outline

Multiple Regression, idea

Multiple regression is the extension of simple linear regression to more than one explanatory variable. The notation is a bit more involved, but much is the same as before.

Multiple Regression, model

With multiple regression, there are indices $i = 1, \ldots, n$ for the observations and $j = 1, \ldots, k$ for the jth explanatory variable. The multiple regression model is written

$$Y_i = \beta_0 + \beta_1 X_{i1} + \ldots + \beta_k X_{ik} + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma).$$

Multiple Regression, assumptions

The assumptions of multiple regression are almost the same as for simple linear regression.

- Linearity – each variable is linearly related to the response,
- Independent observations – no two points are dependent on each other.
- Constant Variability – variation of points around least squares fit remains roughly constant, and
- Normality – the residuals should be nearly normal,
- Collinearity – try to avoid collinear (think correlated) explanatory variables.

Multiple Regression, lite example

We'll consider the data set `datasets::state.x77`. Let's try to predict life expectancy, the variable named `Life Exp`.

```
suppressMessages(library(tidyverse))
# ?state.x77
stateData <- as.data.frame(state.x77) # make data frame
names(stateData) # look at variable names

## [1] "Population" "Income"
## [3] "Illiteracy" "Life Exp"
## [5] "Murder"     "HS Grad"
## [7] "Frost"      "Area"

# Plot the data!, make scatter plots.
```

Multiple Regression, lite example

Let's do some prep-work to help ourselves out.

```
## new variable names without spaces
st <- mutate(stateData, HSGrad = `HS Grad`,
             LifeExp = `Life Exp`,
             Density = Population*1000/Area)
```

Multiple Regression, lite example

Now fit a model with lots of variables to predict life expectancy.

```
fit <- lm(LifeExp ~ Income + Illiteracy + Murder +
            HSGrad + Frost + Area + Density,
          data=st)
summary(fit) # Rstudio
```

Multiple Regression, interpretation

Interpretation of the model usually invovles one slope estimate at a time.
Holding all else constant for every one day increase in the mean number of days with minimum temperature below freezing in the capital city life expectancy decreases by 0.007044 years.

Multiple Regression, interpretation

Holding all else constant for every one unit increase in the murder rate life expectancy decreases by 0.2864 years.

Multiple Regression, adjusted R^2

The adjusted R^2 value for this model is 0.6753. Thus, this linear model accounts for 67.53% of the variation in life expectancy.

Multiple Regression, simple model selection

It is generally OK to toss out variables, one at a time based on their p-value. Drop the variable with the largest p-value and refit the model for each next dropped variable, until all the p-values are less than your pre-specified level of significance.

```
## try this and watch what happens to un/adjusted R^2
```

Multiple Regression, check assumptions

Mostly, we check the assumptions of the model just the same as before.

Multiple Regression, check assumptions

Turning back to our the life expectancy model. Let's check the model assumptions.

```
bestfit <- lm(LifeExp ~ Murder + HSGrad + Frost, data=st)
```

Multiple Regression, check assumptions

Let's extract the standardized residuals and the fitted values.

```
r <- rstandard(bestfit)
yhat <- fitted(bestfit)
df <- data.frame(r=r, yhat=yhat)
```

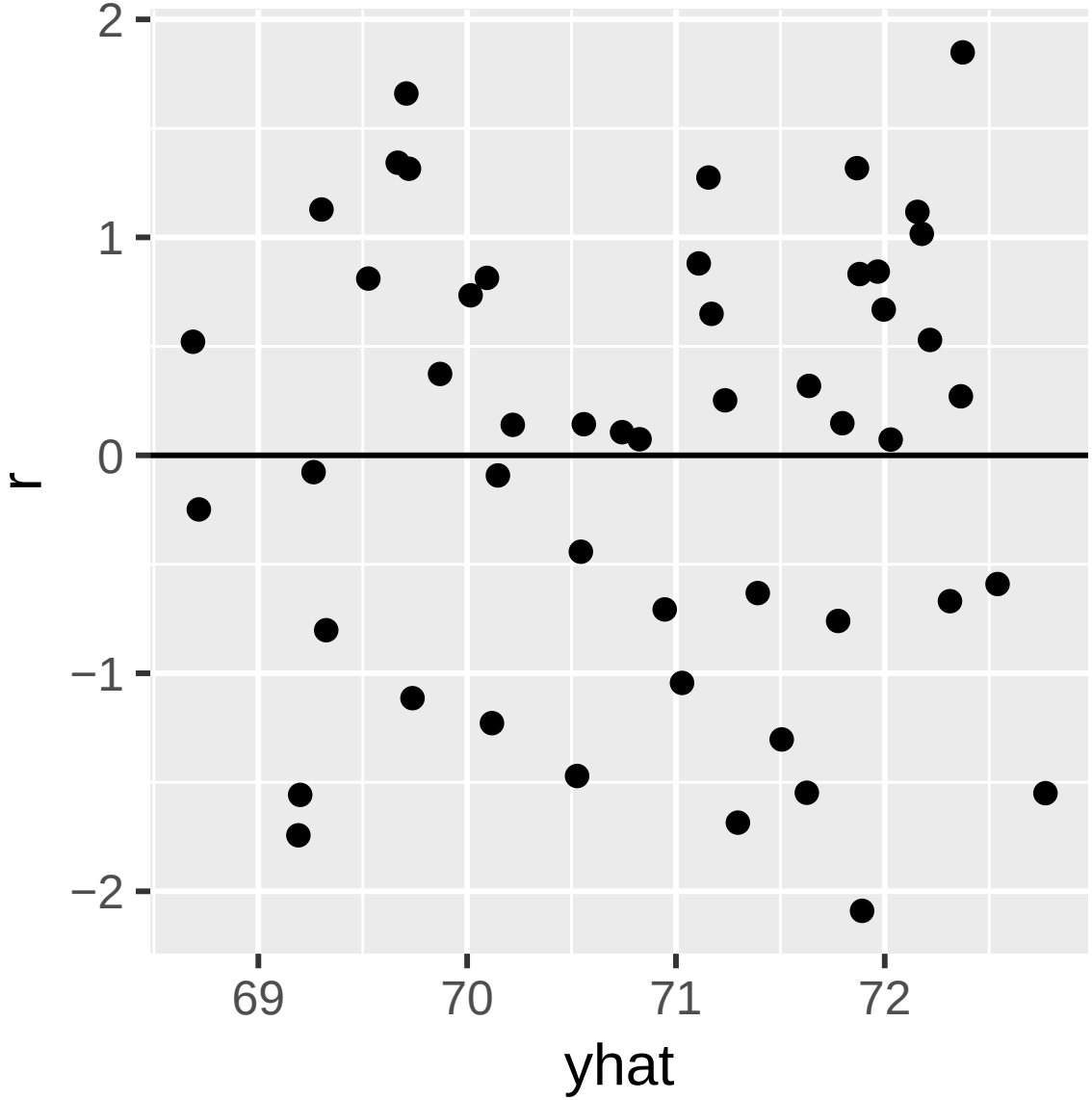
Multiple Regression, check assumptions

Residuals on fitted values – check linearity, constant variation, and outliers.

```
linearity <- ggplot(df, aes(yhat, r)) +
    geom_point() +
    geom_hline(yintercept=0)
```

Multiple Regression, check assumptions

Not bad – no obvious signs of a pattern, and constant variation.



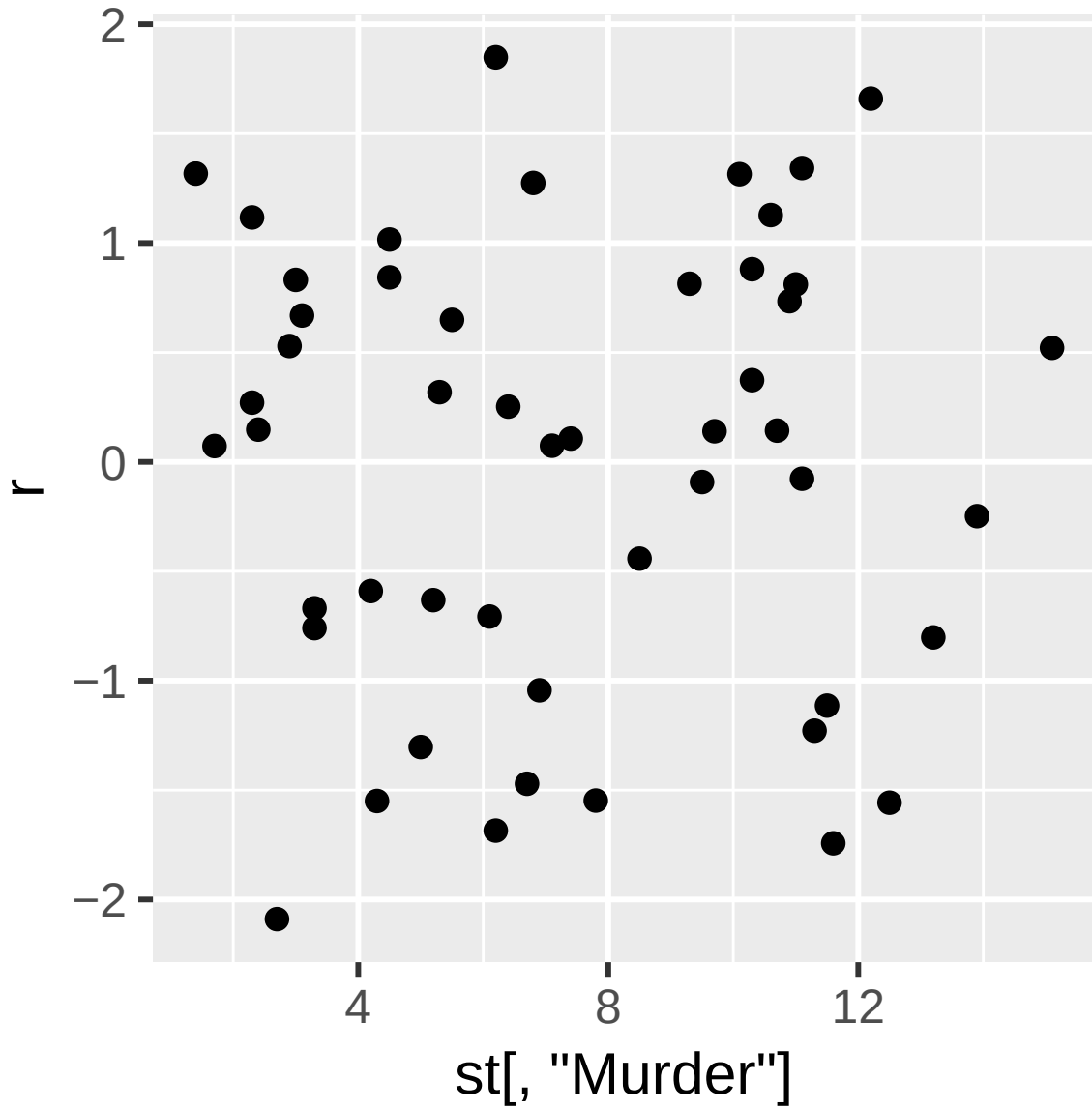
Multiple Regression, check assumptions

It's good, but sometimes cumbersome, to check for linearity relative to each predictor variable of your final model.

```
rm <- qplot(st[,"Murder"], r)
rh <- qplot(st[,"HSGrad"], r)
rf <- qplot(st[,"Frost"], r)
```

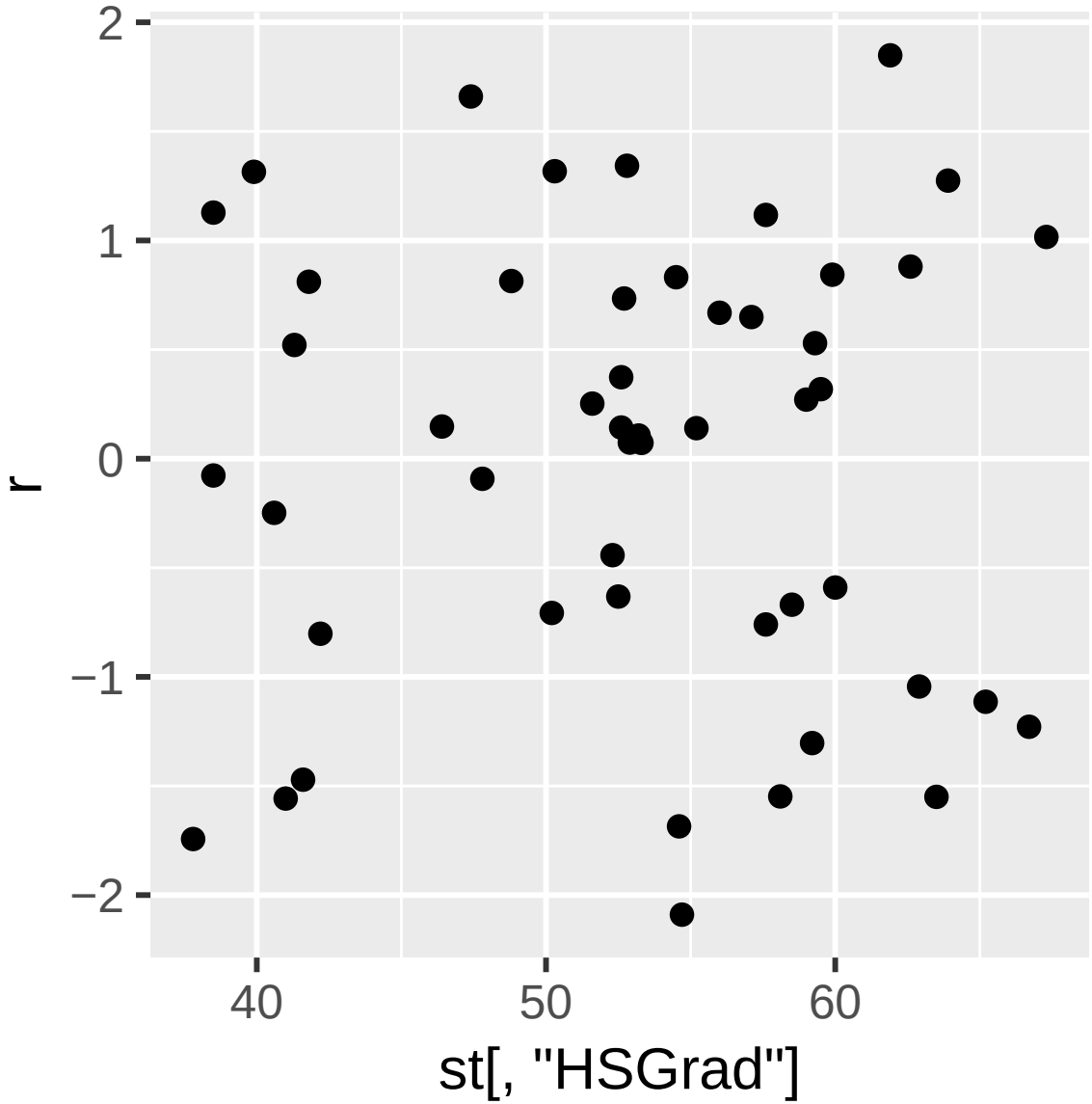
Multiple Regression, check assumptions

Residuals on murder rate.



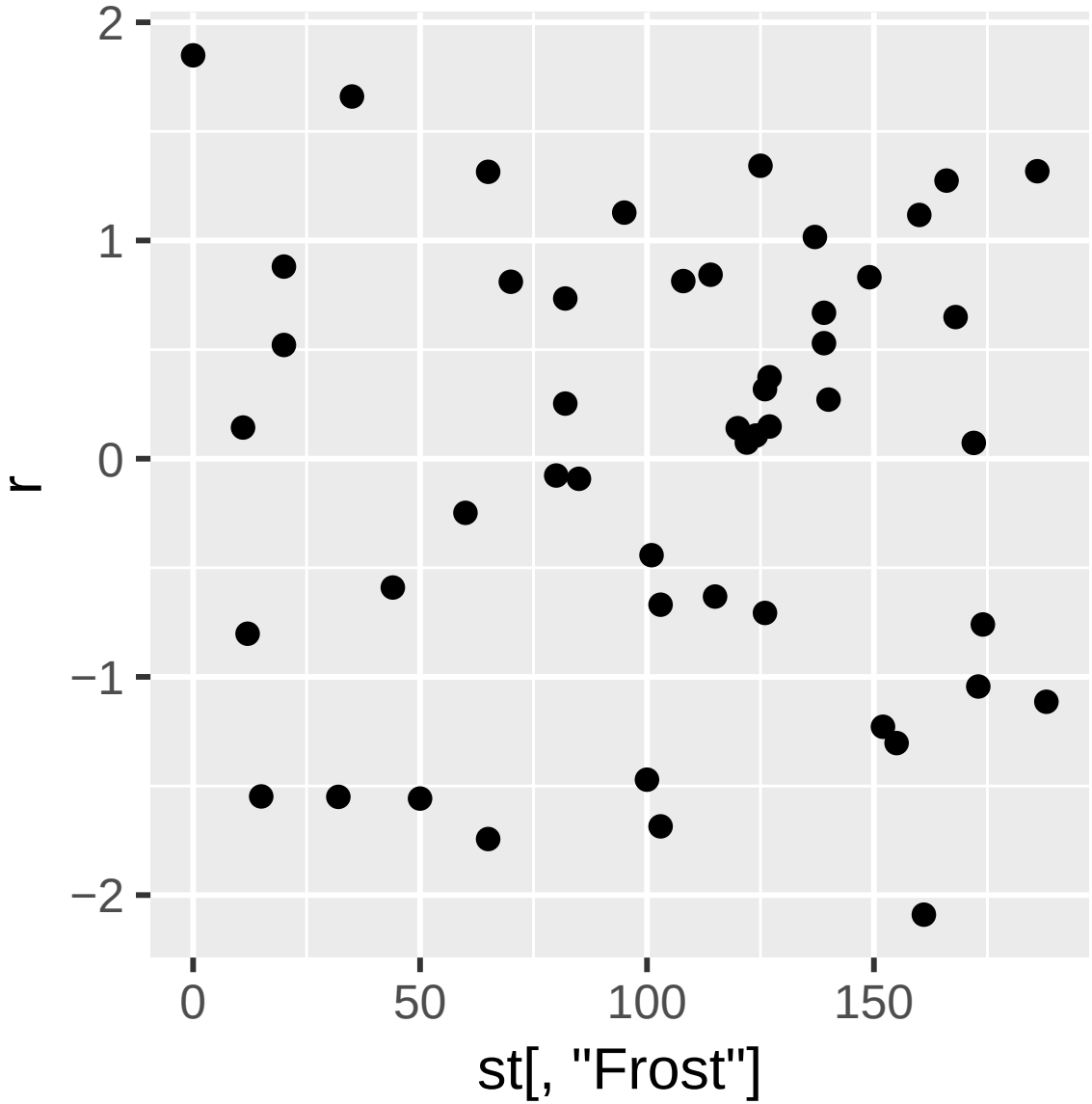
Multiple Regression, check assumptions

Residuals on high school graduation percentage.



Multiple Regression, check assumptions

Residuals on mean number of frost days.



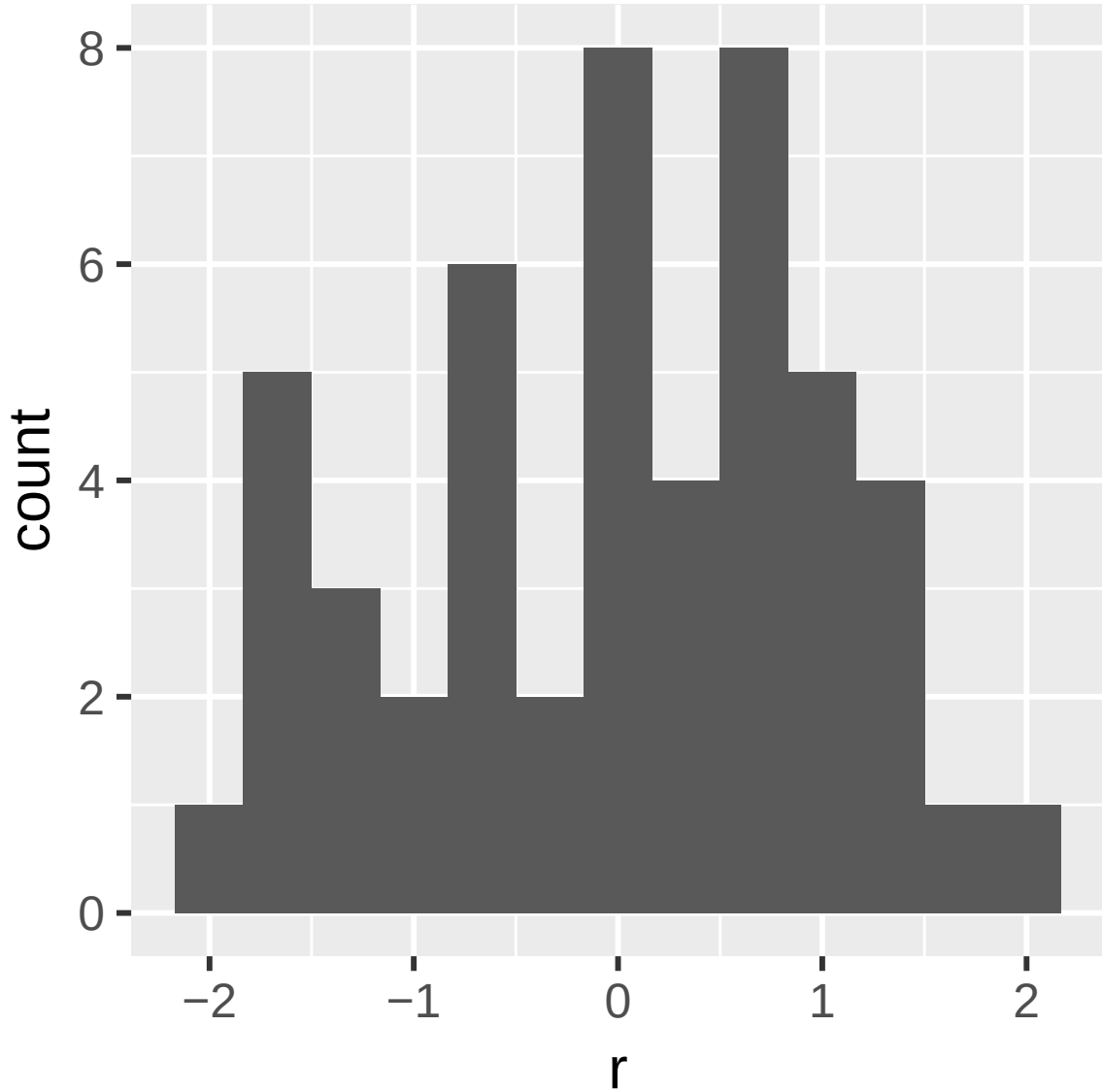
Multiple Regression, check assumptions

Histogram of residuals – check normality.

```
normality <- ggplot(df, aes(r)) +
    geom_histogram(binwidth=1/3)
```

Multiple Regression, check assumptions

Seems not terrible – not heavily skewed.



outline

Brief Recap

Recall that ANOVA esimates means by a categorical variable, and linear regression estimates means by a numerical variable. Multiple linear regression blends these two models, by potentially estimating a slope on the numerical variable(s) for each level of the categorical variable(s)[1].

[1] Often the levels of a categorical variable are called **indicator**s variable in this context.

ANCOVA

We can mix ANOVA and linear regression. Some people call this analysis of covariance, ANCOVA.

Multiple Regression, heavy example

Consider the data set `ape::carnivora`.

```
suppressMessages(library(ape))
data(carnivora)
carnivs <- filter(carnivora,
                  Family %in% c("Canidae",
                                "Felidae",
                                "Mustelidae"))
# Plot the data!, make box/scatter plots.
```

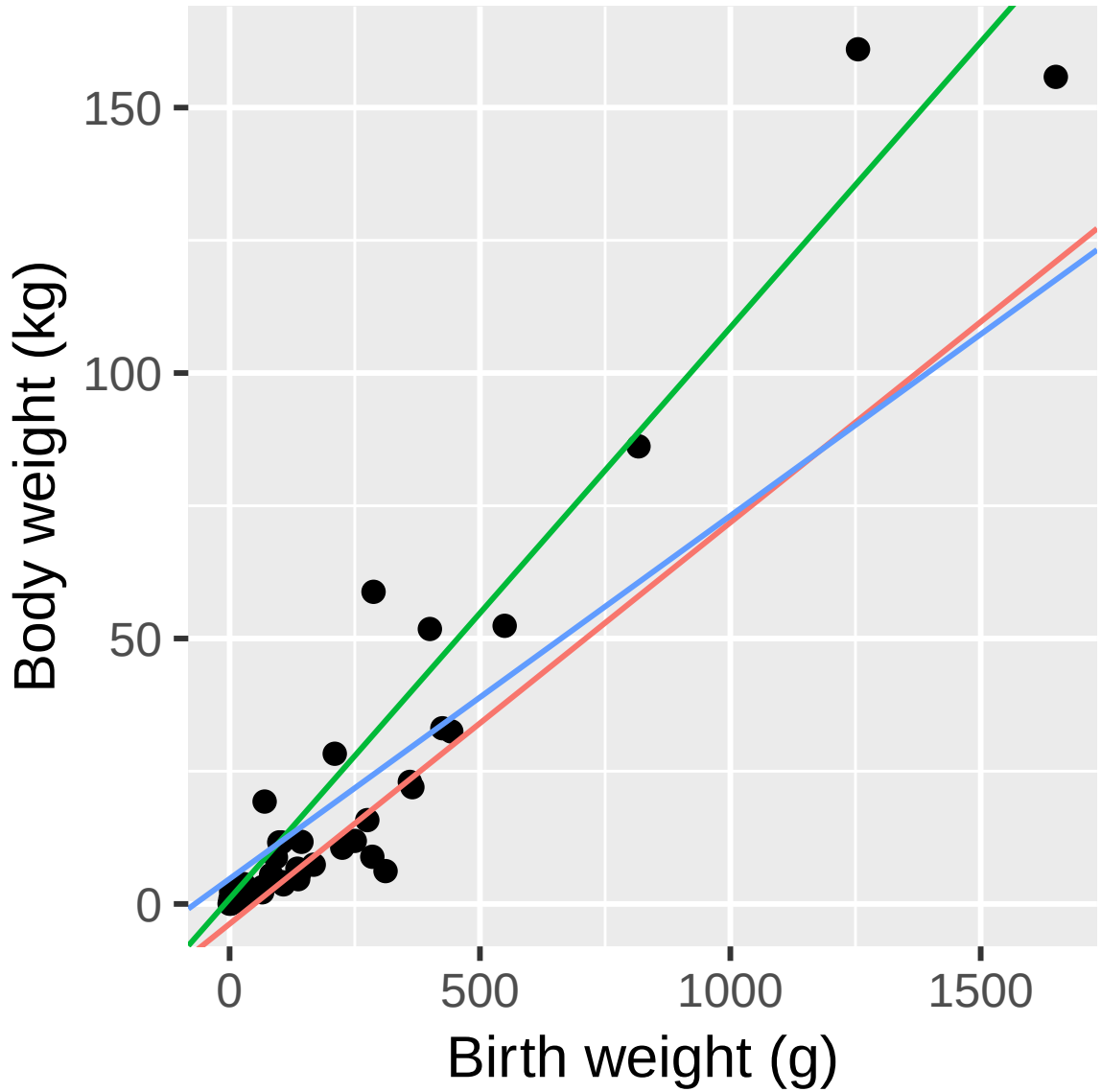
Multiple Regression, heavy example

Slopes and intercepts by a categorical variable.

```
# Recall, R needs to know Family is
# a categorical variable / factor
is.factor(carnivs[,"Family"])
carnfit3 <- lm(SW ~ Family + Family:BW, data=carnivs) ## s
# carnfit3 <- lm(SW ~ Family*BW, data=carnivs)
summary(carnfit3) # RStudio
```

Multiple Regression, heavy example

Inpendent slopes and intercepts by a categorical.



Multiple Regression, heavy example

Slopes and intercepts by a categorical variable. The fitted regression equation is

$$\begin{aligned}\widehat{BodyWeight} = -3.75 &+ 0.99 * Felidae + 4.73 * Mustelidae \\ &+ 0.08 * Canidae * BirthWeight \\ &+ 0.11 * Felidae * BirthWeight \\ &+ 0.07 * Mustelidae * BirthWeight\end{aligned}$$

outline

Take Away

Multiple regression adds new layers of complexity to a relatively simple idea:

- fitting multiple lines across multiple explanatory variables,
- each line is interpretted with the other variables "held constant,"
- mixing ANOVA and linear regression greatly expands our ability to model the real world
  - multiple intercepts by level of categorical variables
  - multiple slopes by level of categorical variables
  - or both slopes and intercepts by categorical variable
- Interpretation of adjusted R^2 remains
- checking model assumptions still necessary

outline

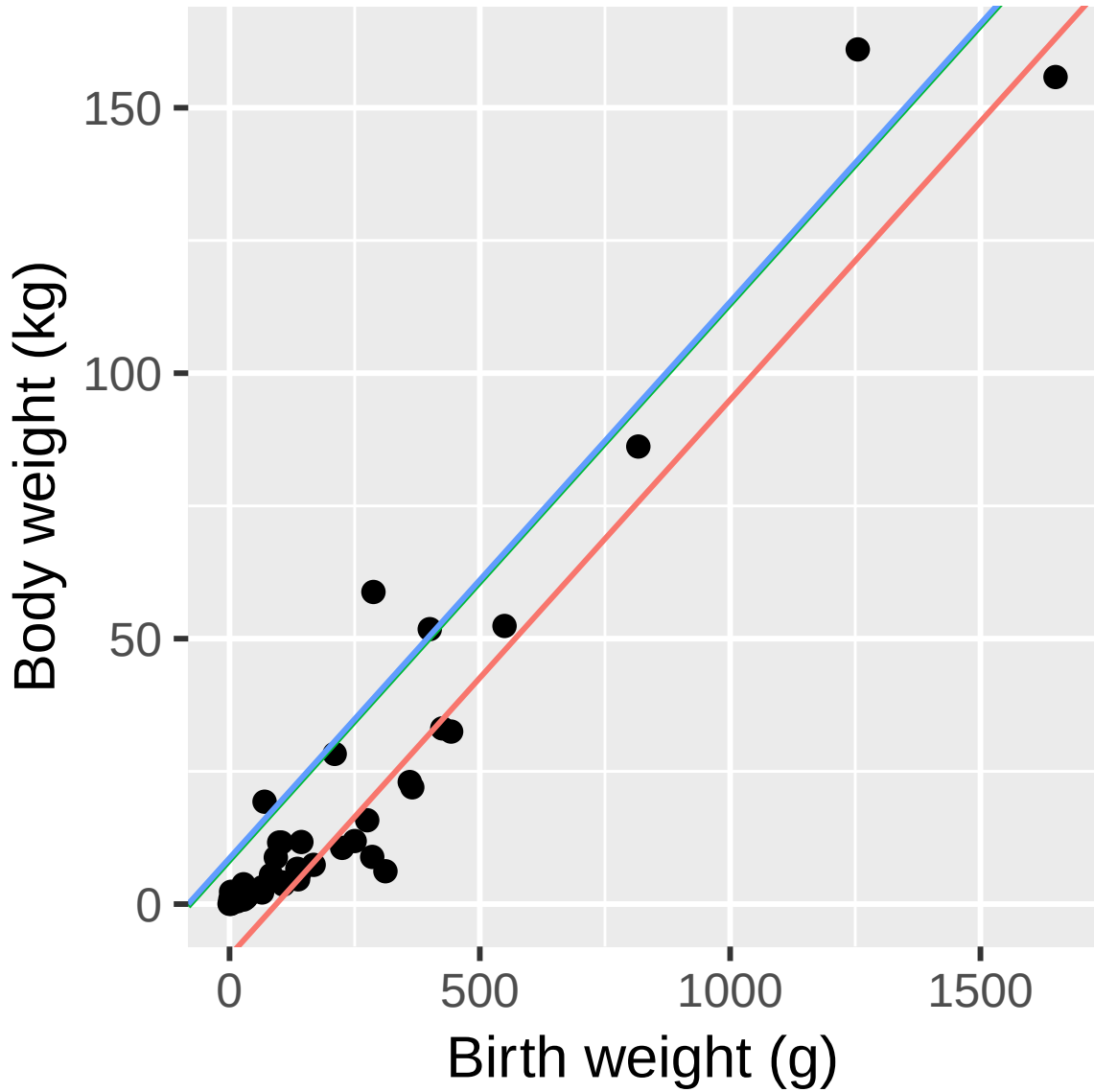
Multiple Regression, heavy example

Let's try to predict body weight from birth weight, creating a new line, each with the same slope, for each family in the data set `carnivs`.

```
# Recall, R needs to know Family is
# a categorical variable / factor
is.factor(carnivs[,"Family"])
carnfit <- lm(SW ~ Family + BW, data=carnivs)
summary(carnfit) # RStudio
```

Multiple Regression, heavy example

Three independent intercepts by a categorical variable – two of them heavily overlap (zoom in).



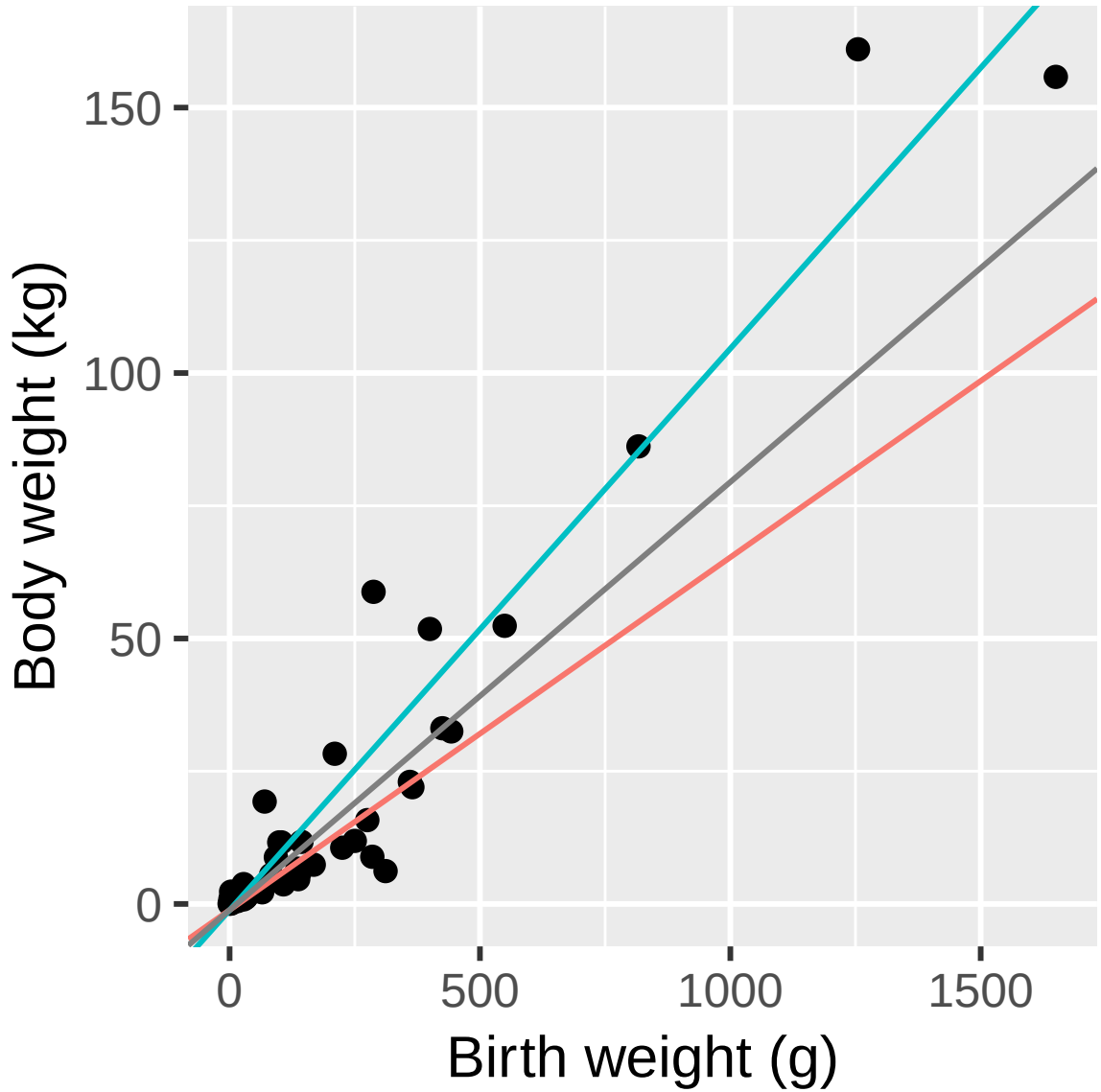
Multiple Regression, heavy example

We could also create new slopes, with one intercept, by the levels of a categorical variable.

```
carnfit2 <- lm(SW ~ Family:BW, data=carnivs)
summary(carnfit2) # RStudio
```

Multiple Regression, heavy example

Independent slopes by a categorical variable.



outline