STT 3850 : Week 4

Spring 2024

Appalachian State University

Section 1

Outline for the week

By the end of the week: Basic Regression

- Data Modeling
- Exploratory data analysis
- Linear regression

Section 2

Basic Regression

Basic Regression

- Now that we are equipped with
  - an understanding of how to import data
  - data visualization and
  - data wrangling skill
- Let's now proceed with **data modeling**.
- The fundamental premise of data modeling is to make explicit the relationship between:
  - an **outcome variable** y, also called a **dependent variable** or **response variable**, and
  - an e**xplanatory/predictor** variable x, also called an **independent variable** or **covariate**.

Data Modeling

Data modeling serves one of two purposes:

1. Modeling for explanation:
   - Describe and quantify the relationship between the outcome variable y and a set of explanatory variables x.
   - Determine the significance of any relationships.
   - Have measures summarizing these relationships.
   - Possibly identify any causal relationships between the variables.
2. Modeling for prediction:
   - Predict an outcome variable y based on the information contained in a set of predictor variables x.
   - Here, you don't care so much about understanding how all the variables relate and interact with one another.

Data Modeling

- For example, say you are interested in
  - an outcome variable y of whether patients develop lung cancer and
  - information x on their risk factors, such as smoking habits, age, and socioeconomic status.
- If we are modeling for explanation,
  - we would be interested in both describing and quantifying the effects of the different risk factors.
  - One reason could be that you want to design an intervention to reduce lung cancer incidence in a population, such as targeting smokers of a specific age group with advertising for smoking cessation programs.
- If we are modeling for prediction,
  - we wouldn't care so much about understanding how all the individual risk factors contribute to lung cancer,
  - but rather only whether we can make good predictions of which people will contract lung cancer.

Linear regression

- There are many techniques for modeling, such as
  - tree-based models and
  - neural networks,
- But in this class, we'll focus on one particular technique: **linear regression**.
- Linear regression involves a numerical outcome variable y and explanatory variables x that are either numerical or categorical.
  - the relationship between y and x is assumed to be linear, or in other words, a line.
  - However, we'll see that what constitutes a "line" will vary depending on the nature of your explanatory variables x.
  - Linear regression is one of the most commonly-used and easy-to-understand approaches to modeling.

Needed packages

Let's now load all the packages needed

```
library(ggplot2)    #  for data visualization
library(dplyr)      #  for data wrangling
library(readr)      # for importing spreadsheet data into R
library(moderndive) # package of datasets and regression funct
library(skimr)      # provides simple-to-use functions
                    # for summary statistics
```

One numerical explanatory variable

- Researchers at the University of Texas in Austin, Texas (UT Austin) tried to answer the following research question:
  - what factors explain differences in instructor teaching evaluation scores?
- To this end, they collected instructor and course information on 463 courses.
- A full description of the study can be found at https://openintro.org.
- The data on the 463 courses at UT Austin can be found in the `evals` data frame included in the `moderndive` package.

One numerical explanatory variable

Let's fully describe the 4 variables we will focus on:

1. `ID`: An identification variable used to distinguish between the 1 through 463 courses in the dataset.
2. `score`: A numerical variable of the course instructor's average teaching score, where the average is computed from the evaluation scores from all students in that course. Teaching scores of 1 are lowest and 5 are highest. This is the outcome variable y of interest.
3. `bty_avg`: A numerical variable of the course instructor's average "beauty" score, where the average is computed from a separate panel of six students. "Beauty" scores of 1 are lowest and 10 are highest. This is the explanatory variable x of interest.
4. `age`: A numerical variable of the course instructor's age. This will be another explanatory variable x that we'll use later.

One numerical explanatory variable

We'll answer these questions by modeling the relationship between teaching scores and "beauty" scores using simple linear regression where we have:

1. A numerical outcome variable y (the instructor's teaching score) and
2. A single numerical explanatory variable x (the instructor's "beauty" score).

Exploratory data analysis

- A crucial step before doing any kind of analysis or modeling is performing an exploratory data analysis, or EDA for short.
  - Get distributions of the individual variables in your data,
  - Find out any potential relationships exist between variables,
  - Find out any outliers and/or missing values, and
  - (most importantly) helps you to decide how to build your model.
- Here are three common steps in EDA:
  1. Examine the raw data values.
  2. Compute summary statistics, such as means, medians, and interquartile ranges.
  3. Create data visualizations.

Step 1: Examine the raw data values

```
evals_ch5 <- evals %>%
  select(ID, score, bty_avg, age)   # take subset
glimpse(evals_ch5)
```

```
Rows: 463
Columns: 4
$ ID      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
$ score   <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3, 2.8, 4.1, 3.4, 4.5, 3
$ bty_avg <dbl> 5.000, 5.000, 5.000, 5.000, 3.000, 3.000, 3.000, 3.
$ age     <int> 36, 36, 36, 36, 59, 59, 59, 51, 51, 40, 40, 40, 40,
```

Step 1: Examine the raw data values

An alternative way to look at the raw data values is by choosing a random sample of the rows.

```
evals_ch5 %>%
  sample_n(size = 5)
```

```
# A tibble: 5 x 4
     ID score bty_avg   age
  <int> <dbl>   <dbl> <int>
1   218   4.4    4       42
2   435   3.1    2       62
3    68   4.1    4.83    42
4   227   3.3    8.17    39
5   128   4.3    3       62
```

Step 2: summary statistics

```r
evals_ch5 %>%
  summarize(mean_bty_avg = mean(bty_avg),
            mean_score = mean(score),
            median_bty_avg = median(bty_avg),
            median_score = median(score))
```

```
# A tibble: 1 x 4
  mean_bty_avg mean_score median_bty_avg median_score
         <dbl>      <dbl>          <dbl>        <dbl>
1         4.42       4.17           4.33          4.3
```

Step 2: summary statistics

The `skim()` function from the `skimr` package, "skims" the data, and returns commonly used summary statistics.

```
library(skimr)
evals_ch5 %>%
  select(score, bty_avg) %>%
  skim_without_charts()
```

Table 1: Data summary

Name	Piped data
Number of rows	463
Number of columns	2
Column type frequency:	
numeric	2
Group variables	None