Background on modeling for explanation

MODELING WITH DATA IN THE TIDYVERSE

Albert Y. Kim

Assistant Professor of Statistical and Data Sciences

Course overview

1. Introduction to modeling: theory and terminology
2. Regression:
   - Simple linear regression
   - Multiple regression
3. Model assessment

General modeling framework formula

$$y = f(\vec{x}) + \epsilon$$

Where:

- y: outcome variable of interest
- $\vec{x}$: explanatory/predictor variables
- $f()$: function of the relationship between y and $\vec{x}$ AKA *the signal*
- ϵ: unsystematic error component AKA *the noise*

Two modeling scenarios

Modeling for either:

- Explanation: $\vec{x}$ are *explanatory* variables
- Prediction: $\vec{x}$ are *predictor* variables

Modeling for explanation example

A University of Texas in Austin study on teaching evaluation scores (available at **openintro.org**).

Question: Can we explain differences in teaching evaluation score based on various teacher attributes?

Variables:

- y: Average teaching `score` based on students evaluations
- $\vec{x}$: Attributes like `rank`, `gender`, `age`, and `bty_avg`

Modeling for explanation example

From the `moderndive` package for ModernDive.com:

```
library(dplyr)
library(moderndive)
glimpse(evals)
```

```
Observations: 463
Variables: 13
$ ID          <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10...
$ score       <dbl> 4.7, 4.1, 3.9, 4.8, 4.6, 4.3...
$ age         <int> 36, 36, 36, 36, 59, 59, 59, 51...
$ bty_avg     <dbl> 5.000, 5.000, 5.000, 5.000...
$ gender      <fct> female, female, female, female...
...
```

Exploratory data analysis

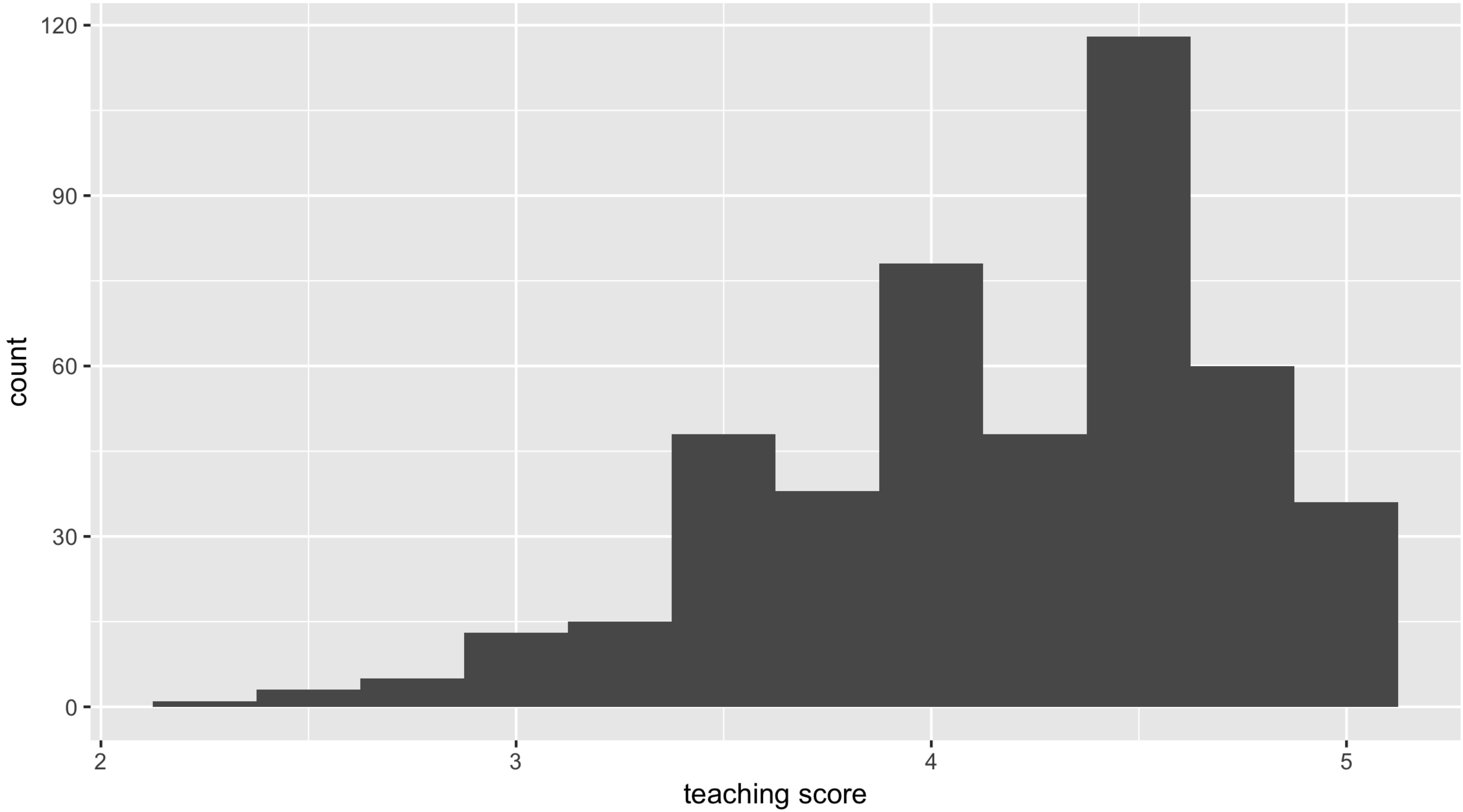
Three basic steps to exploratory data analysis (EDA):

1. Looking at your data
2. Creating visualizations
3. Computing summary statistics

Exploratory data analysis

```
library(ggplot2)
ggplot(evals, aes(x = score)) +
  geom_histogram(binwidth = 0.25) +
  labs(x = "teaching score", y = "count")
```

Exploratory data analysis



Exploratory data analysis

```
# Compute mean, median, and standard deviation
evals %>%
  summarize(mean_score = mean(score),
            median_score = median(score),
            sd_score = sd(score))
```

```
# A tibble: 1 x 3
  mean_score median_score sd_score
       <dbl>        <dbl>    <dbl>
1       4.17          4.3    0.544
```

Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

Background on modeling for prediction

MODELING WITH DATA IN THE TIDYVERSE

Albert Y. Kim
Assistant Professor of Statistical and Data Sciences

Modeling for prediction example

A dataset of house prices in King County, Washington State, near Seattle (available at **Kaggle.com**).

Question: Can we predict the sale price of houses based on their features?

Variables:

- y: House sale `price` is US dollars
- $\vec{x}$: Features like `sqft_living`, `condition`, `bedrooms`, `yr_built`, `waterfront`

Modeling for prediction example

From the `moderndive` package for **ModernDive**:

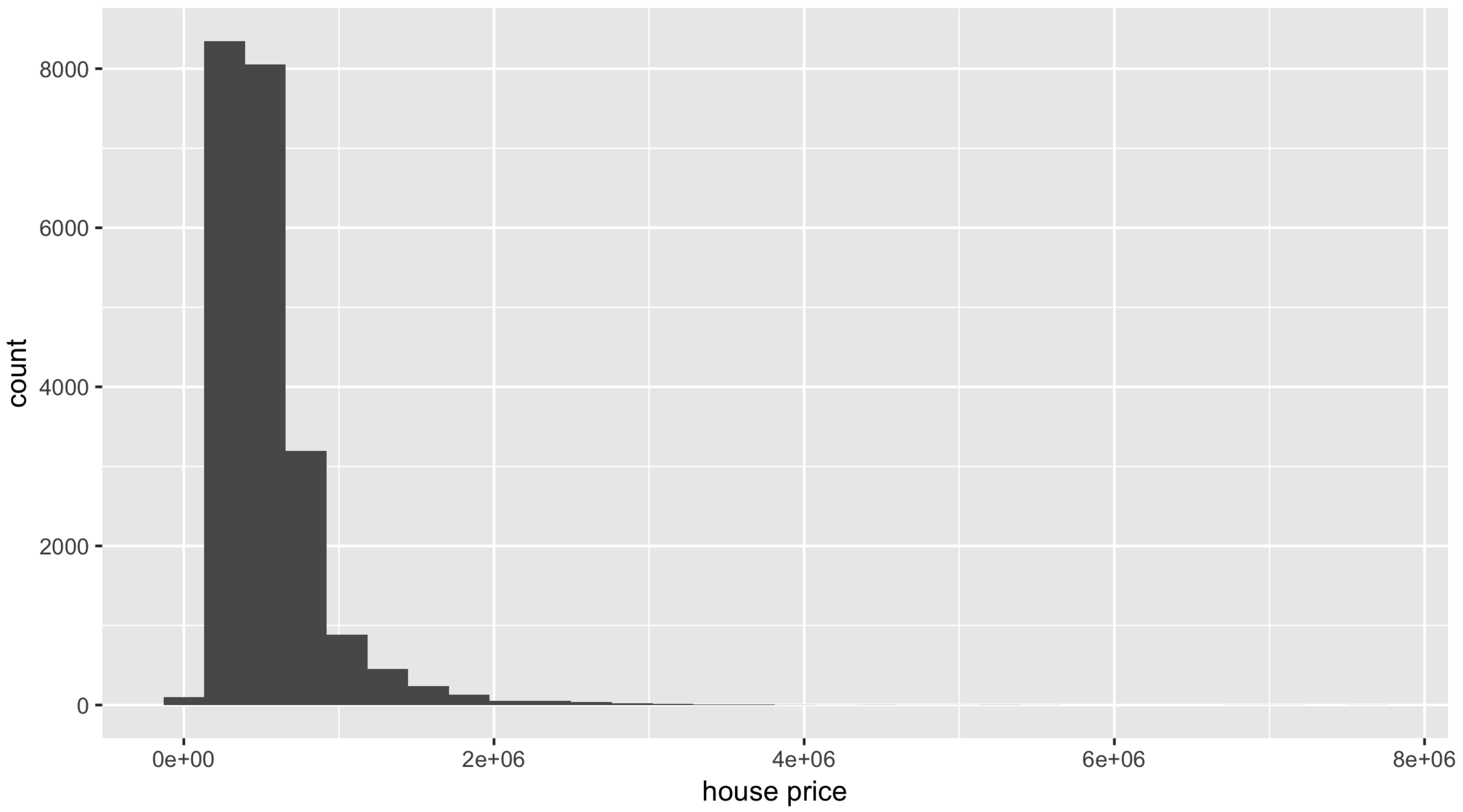
```
library(dplyr)
library(moderndive)
glimpse(house_prices)
```

```
Observations: 21,613
Variables: 21
$ id            <chr> "7129300520", "6414100192"...
$ date          <dttm> 2014-10-13, 2014-12-09, 2015...
$ price         <dbl> 221900, 538000, 180000, 604000...
...
```

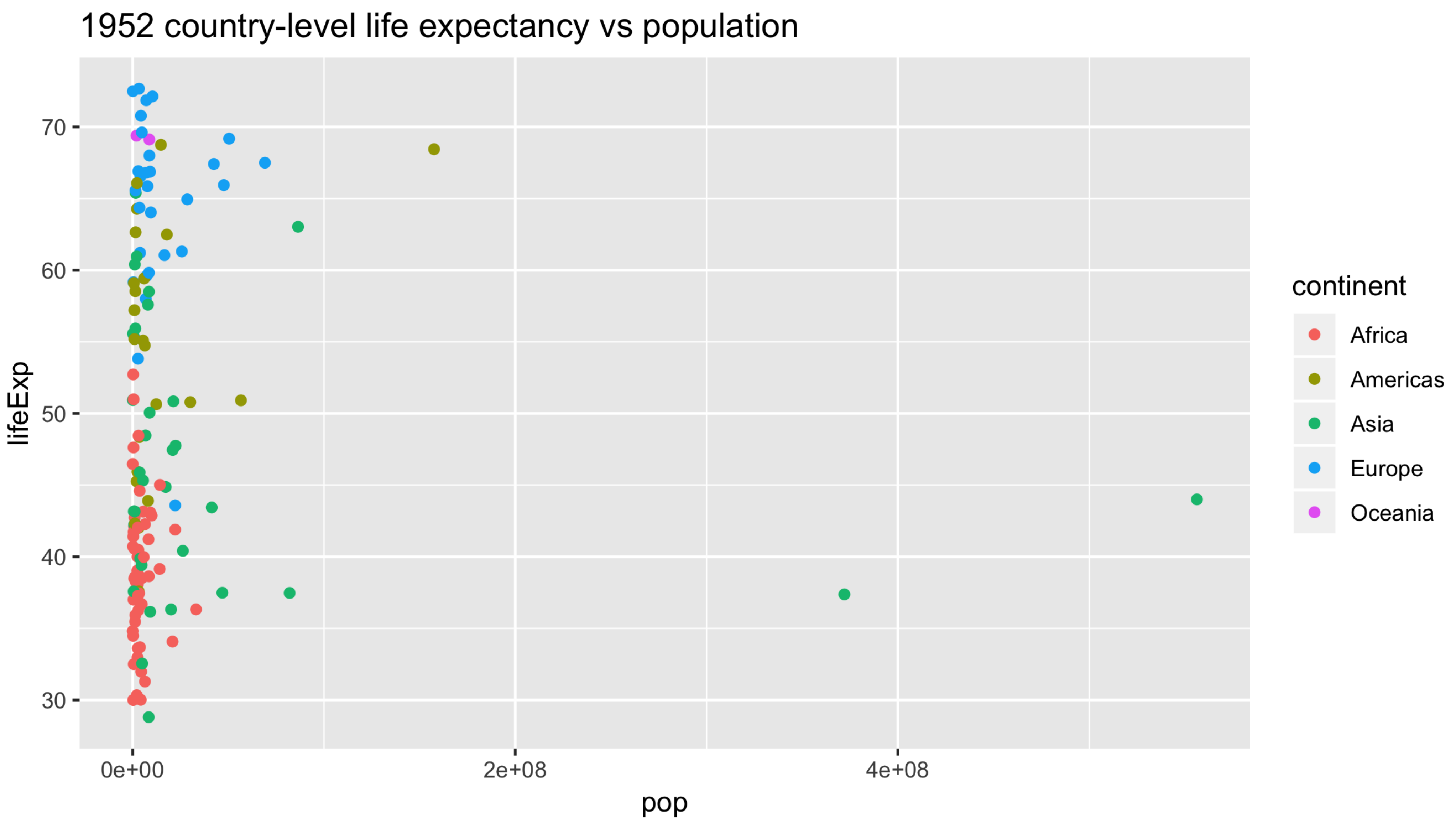
Exploratory data analysis

```
library(ggplot2)
ggplot(house_prices, aes(x = price)) +
  geom_histogram() +
  labs(x = "house price", y = "count")
```

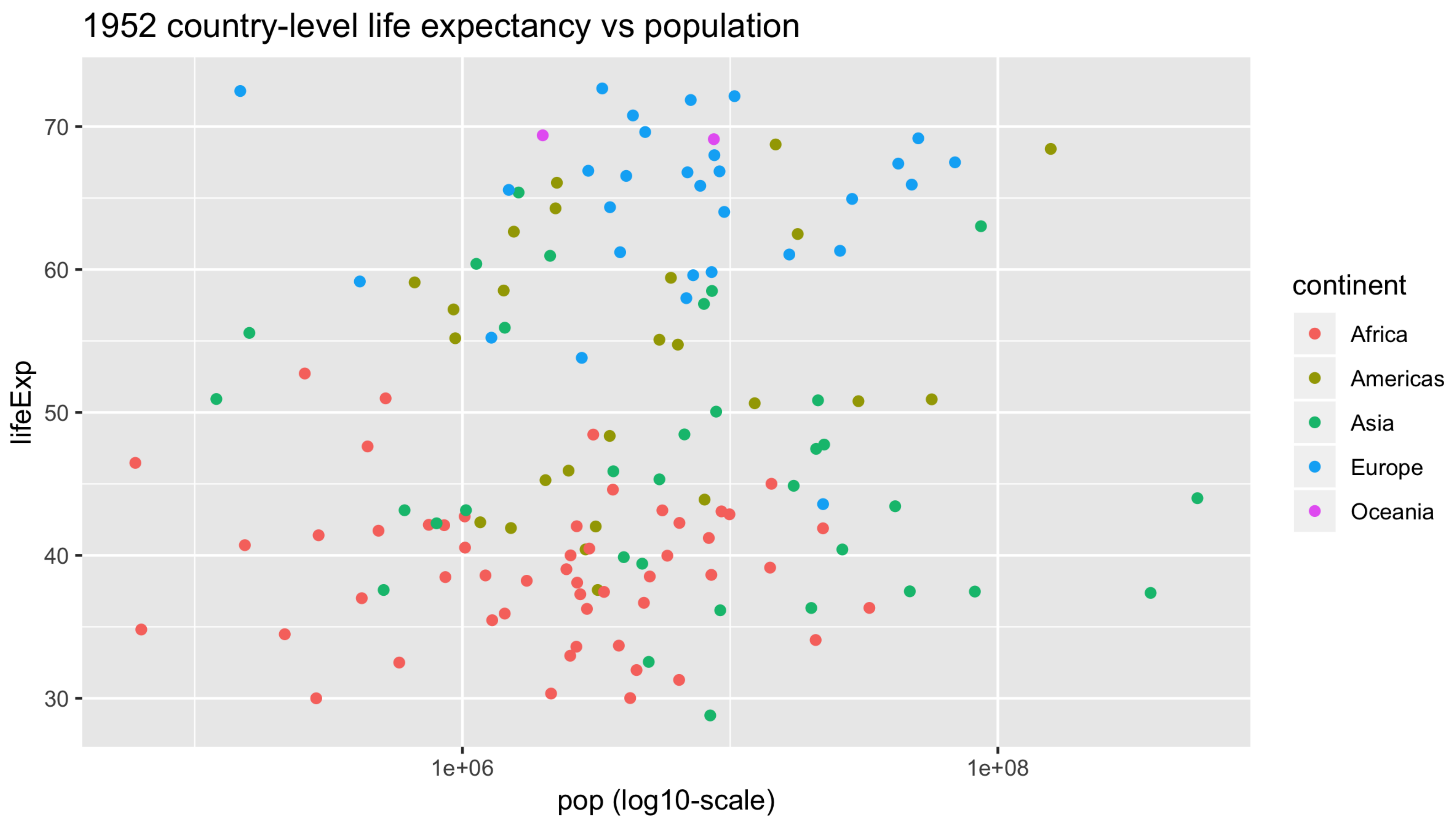
Histogram of outcome variable



Gapminder data



Log10 rescaling of x-axis



Log10 transformation

```
# log10() transform price and size
house_prices <- house_prices %>%
  mutate(log10_price = log10(price)) %>%
  select(price, log10_price)
```

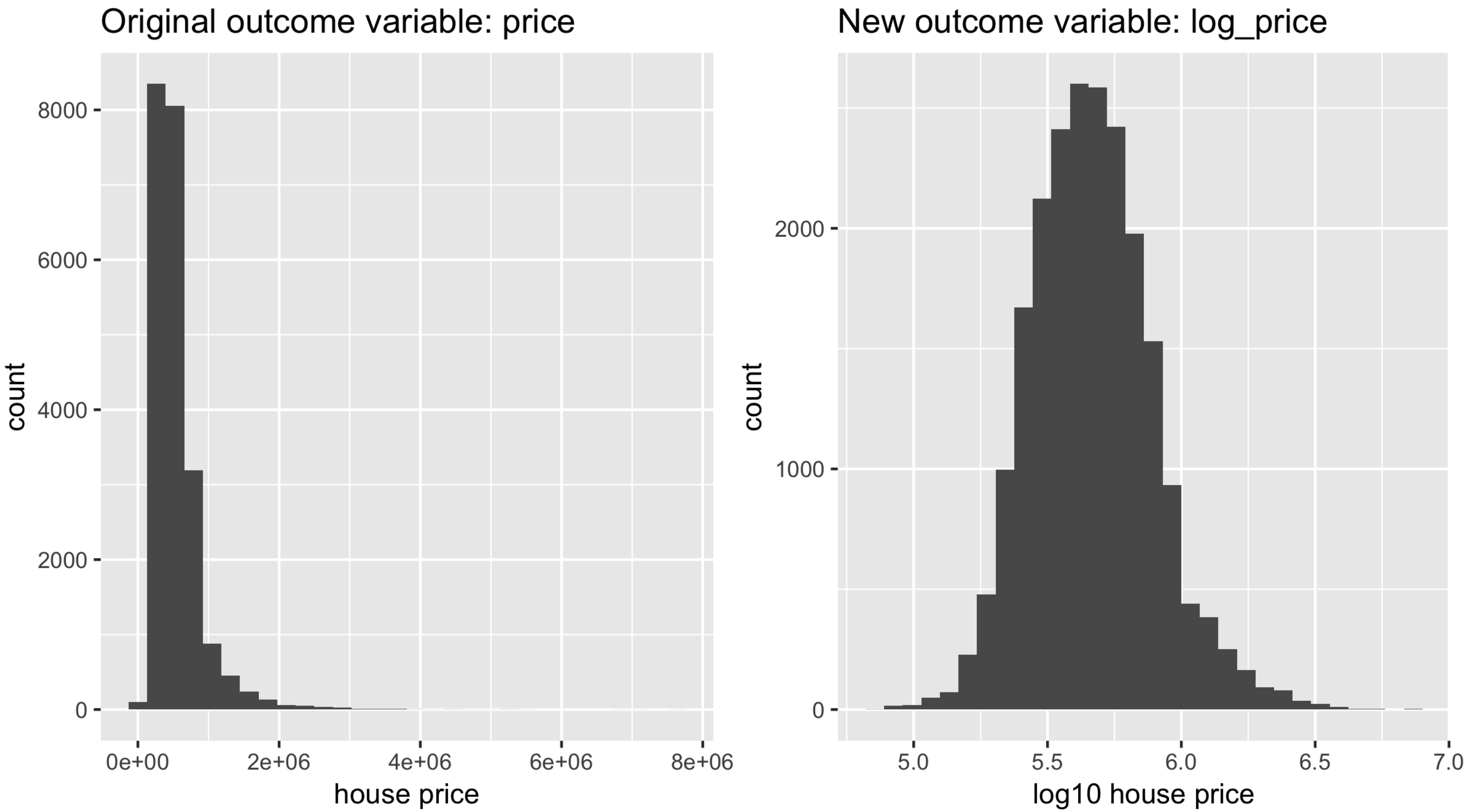
```
# A tibble: 21,613 x 2
     price log10_price
     <dbl>       <dbl>
 1  221900        5.35
 2  538000        5.73
 3  180000        5.26
 4  604000        5.78
 5  510000        5.71
 6 1225000        6.09
```

Histogram of new outcome variable

```
# Histogram of original outcome variable
ggplot(house_prices, aes(x = price)) +
  geom_histogram() +
  labs(x = "house price", y = "count")
```

```
# Histogram of new, log10-transformed outcome variable
ggplot(house_prices, aes(x = log10_price)) +
  geom_histogram() +
  labs(x = "log10 house price", y = "count")
```

Comparing before and after log10-transformation



Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

The modeling problem for explanation

MODELING WITH DATA IN THE TIDYVERSE

Albert Y. Kim

Assistant Professor of Statistical and Data Sciences

Recall: General modeling framework formula

$$y = f(\vec{x}) + \epsilon$$

Where:

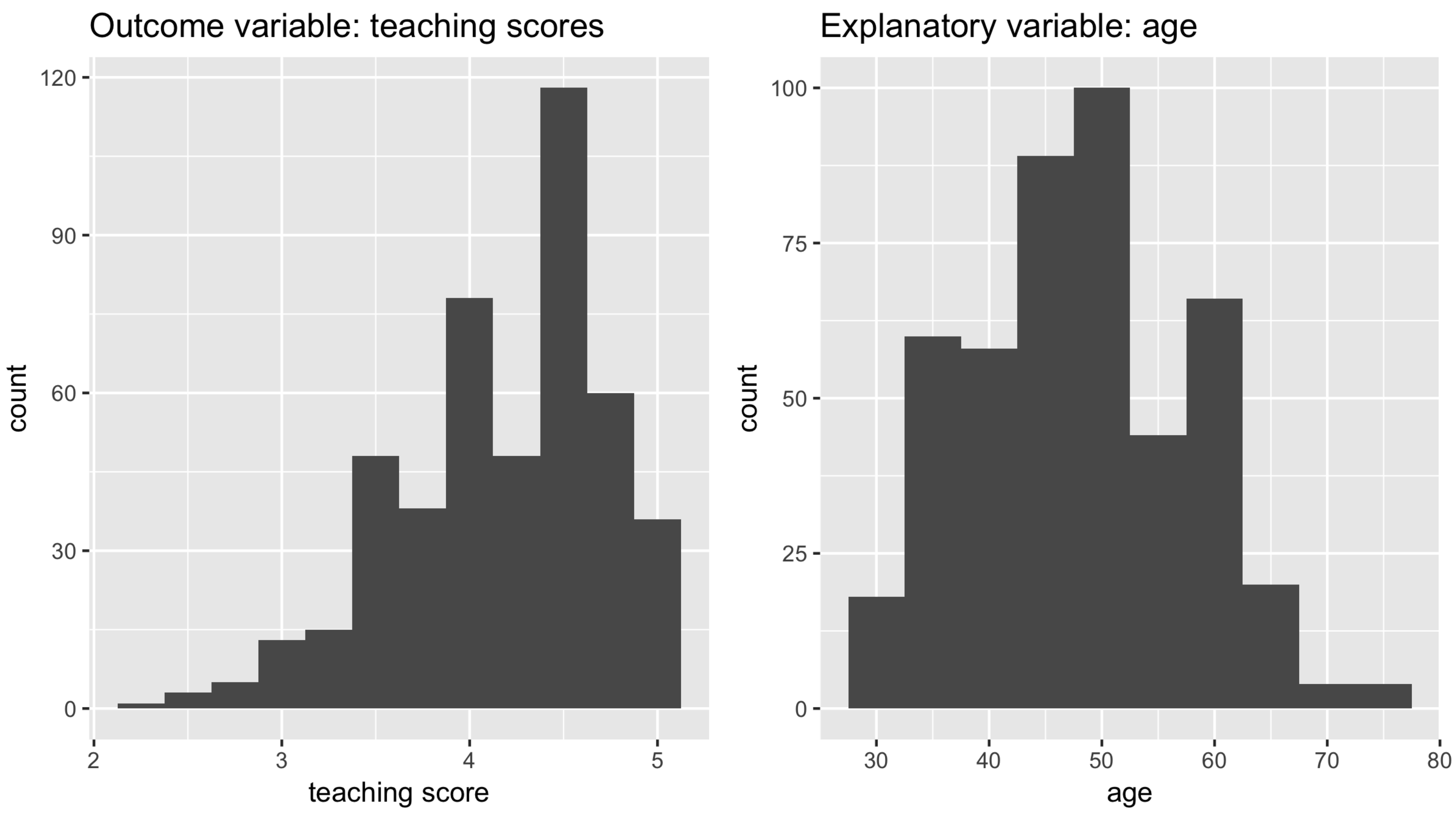
- y: outcome variable of interest
- $\vec{x}$: explanatory/predictor variables
- $f()$: function of the relationship between y and $\vec{x}$ AKA *the signal*
- ϵ: unsystematic error component AKA *the noise*

The modeling problem

Consider $y = f(\vec{x}) + \epsilon$.

1. $f()$ and ϵ are unknown
2. n observations of y and $\vec{x}$ are known/given in the data
3. **Goal:** Fit a model $\hat{f}()$ that *approximates* $f()$ while ignoring ϵ
4. **Goal restated:** *Separate the signal from the noise*
5. Can then generate *fitted/predicted* values $\hat{y} = \hat{f}(\vec{x})$

Modeling for explanation example

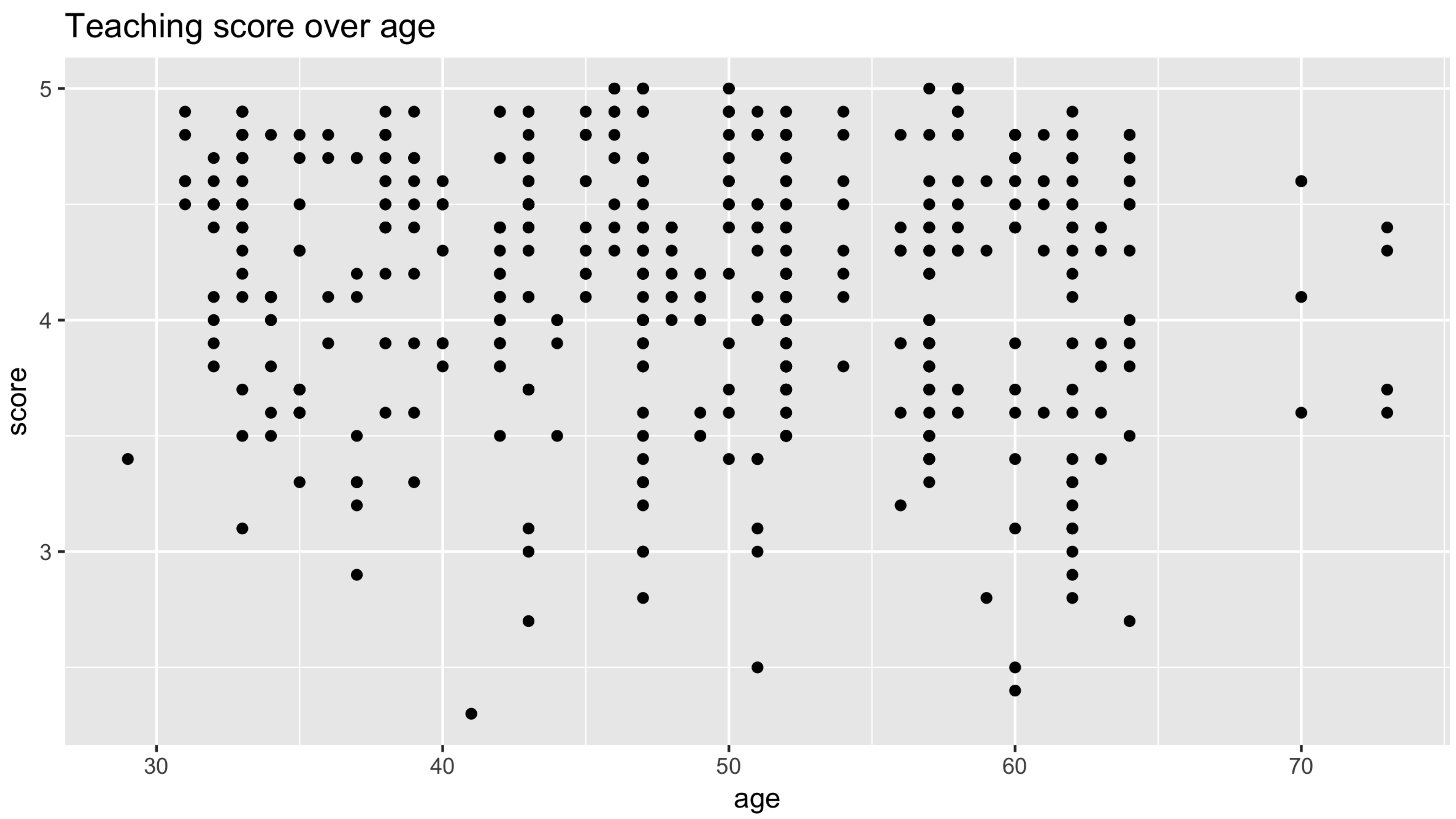


EDA of relationship

```
library(ggplot2)
library(dplyr)
library(moderndive)

ggplot(evals, aes(x = age, y = score)) +
  geom_point() +
  labs(x = "age", y = "score",
       title = "Teaching score over age")
```

EDA of relationship

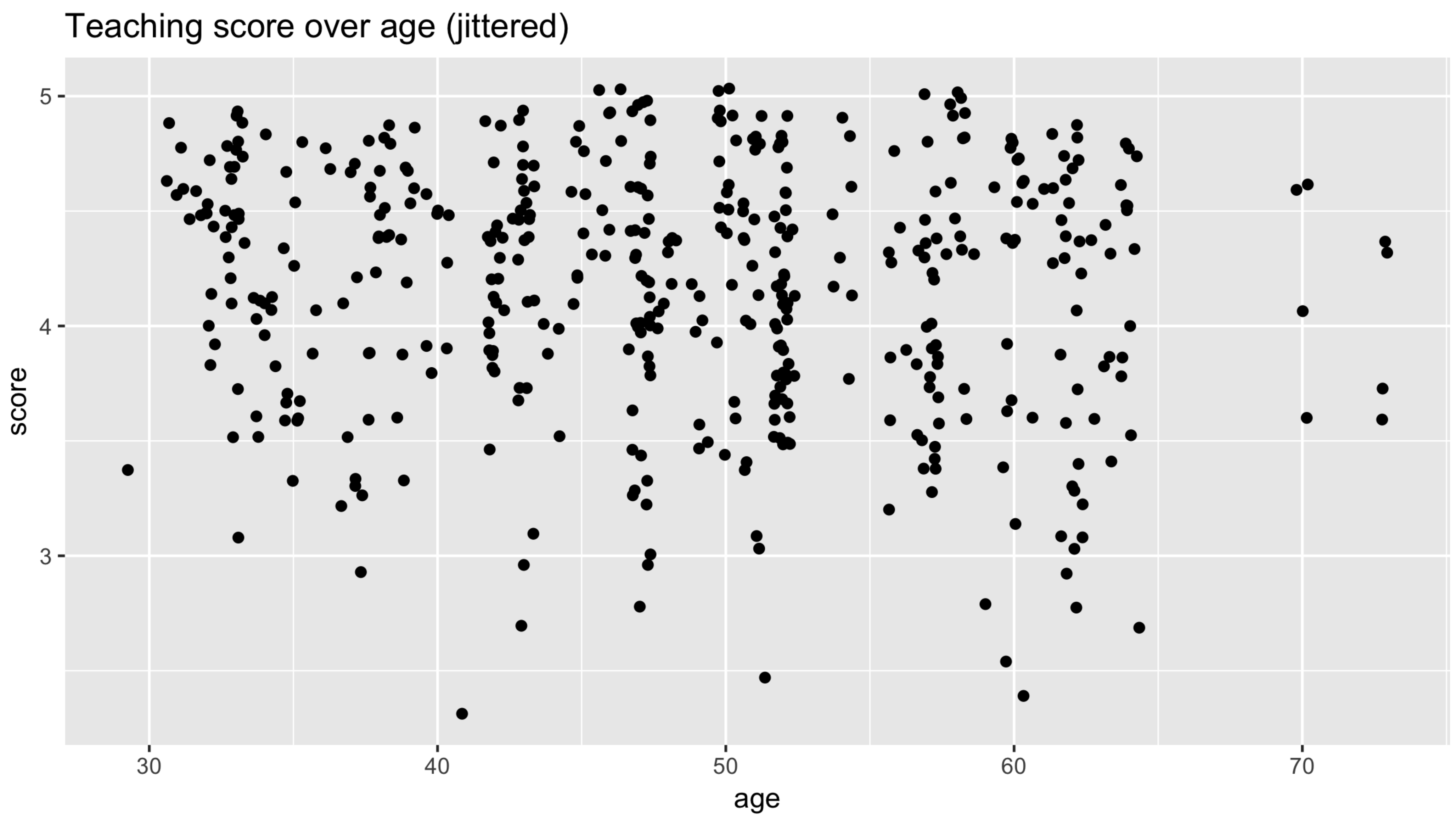


Jittered scatterplot

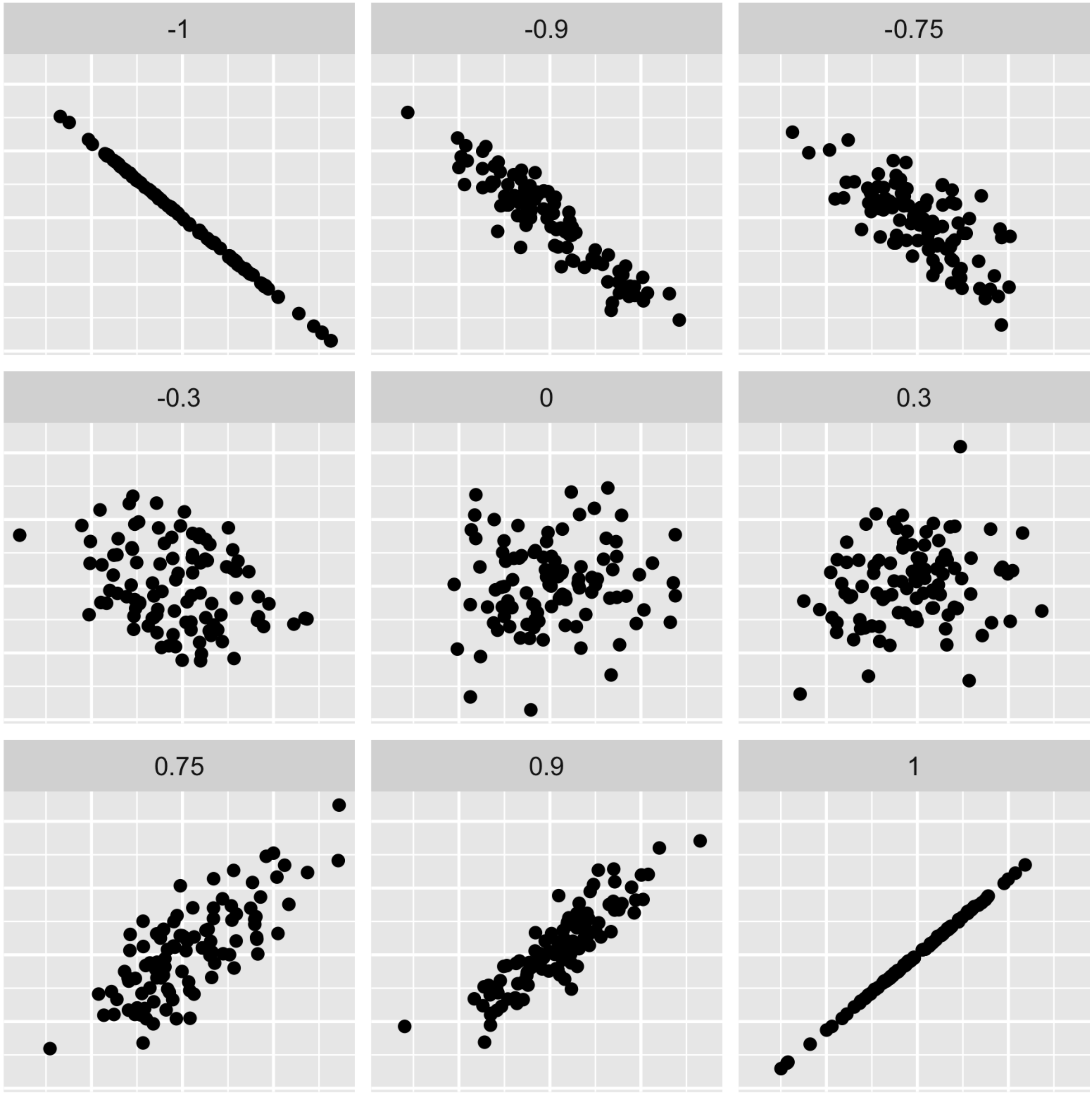
```
library(ggplot2)
library(dplyr)
library(moderndive)

# Use geom_jitter() instead of geom_point()
ggplot(evals, aes(x = age, y = score)) +
  geom_jitter() +
  labs(x = "age", y = "score",
       title = "Teaching score over age (jittered)")
```

Jittered scatterplot



Correlation coefficient



Computing the correlation coefficient

```
evals %>%
  summarize(correlation = cor(score, age))
```

```
# A tibble: 1 x 1
  correlation
        <dbl>
1      -0.107
```

Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

The modeling problem for prediction

MODELING WITH DATA IN THE TIDYVERSE

Albert Y. Kim
Assistant Professor of Statistical and Data Sciences

Modeling problem

Consider $y = f(\vec{x}) + \epsilon$.

1. $f()$ and ϵ are unknown
2. n observations of y and $\vec{x}$ are known/given in the data
3. **Goal:** Fit a model $\hat{f}()$ that *approximates* $f()$ while ignoring ϵ
4. **Goal restated:** Separate the *signal* from the *noise*
5. Can then generate *fitted/predicted* values $\hat{y} = \hat{f}(\vec{x})$

Difference between explanation and prediction

Key difference in modeling goals:

1. **Explanation:** We care about the form of $\hat{f}()$, in particular any values quantifying relationships between y and $\vec{x}$
2. **Prediction:** We don't care so much about the form of $\hat{f}()$, only that it yields "good" predictions $\hat{y}$ of y based on $\vec{x}$

Condition of house

```
house_prices %>%
  select(log10_price, condition) %>%
  glimpse()
```

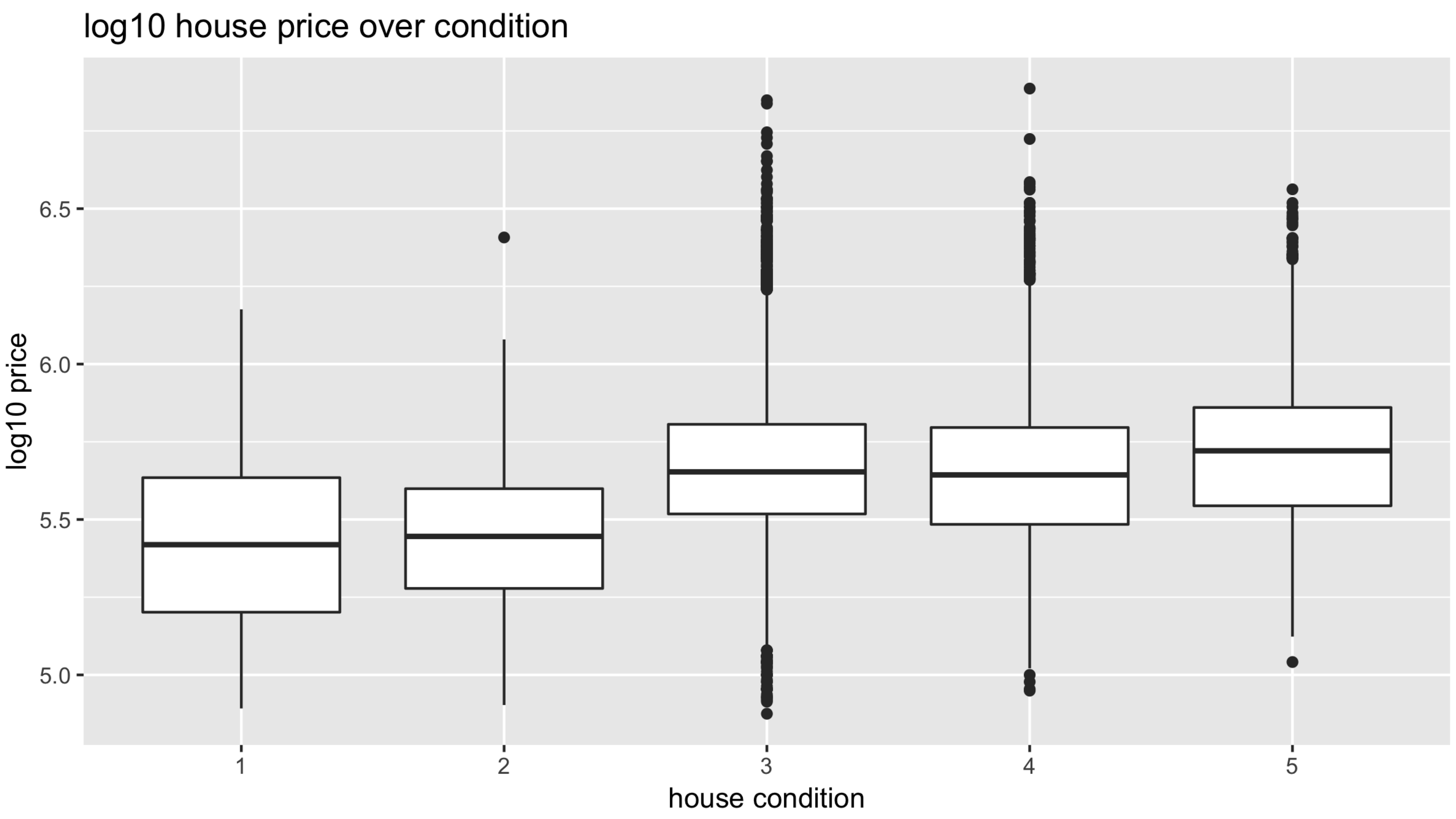
```
Observations: 21,613
Variables: 2
$ log10_price <dbl> 5.346157, 5.730782, 5.255273...
$ condition   <fct> 3, 3, 3, 5, 3, 3, 3, 3, 3, 3, 3...
```

Exploratory data visualization: boxplot

```
library(ggplot2)
library(dplyr)
library(moderndive)

# Apply log10-transformation to outcome variable
house_prices <- house_prices %>%
  mutate(log10_price = log10(price))
# Boxplot
ggplot(house_prices, aes(x = condition, y = log10_price)) +
  geom_boxplot() +
  labs(x = "house condition", y = "log10 price",
       title = "log10 house price over condition")
```

Exploratory data visualization: boxplot



Exploratory data summaries

```
house_prices %>%
  group_by(condition) %>%
  summarize(mean = mean(log10_price),
            sd = sd(log10_price), n = n())
```

```
# A tibble: 5 x 4
  condition  mean    sd     n
  <fct>     <dbl> <dbl> <int>
1 1          5.42 0.293    30
2 2          5.45 0.233   172
3 3          5.67 0.224 14031
4 4          5.65 0.228  5679
5 5          5.71 0.244  1701
```

Exploratory data summaries

```
# Prediction for new house with condition 4 in dollars
10^(5.65)
```

```
446683.6
```

Let's practice!

MODELING WITH DATA IN THE TIDYVERSE