MODELING WITH DATA IN THE TIDYVERSE

Explaining house price with year & size

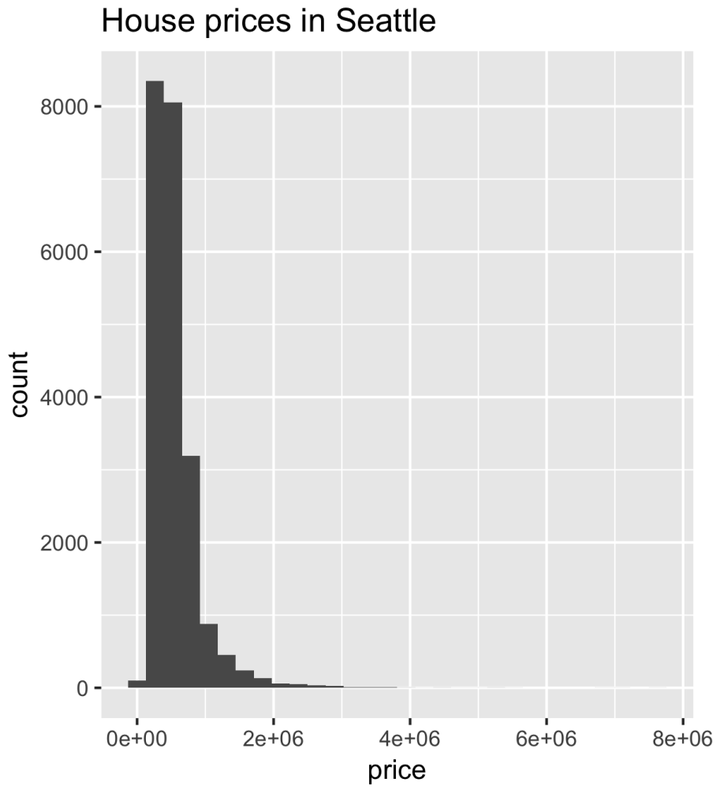
Albert Y. Kim
Assistant Professor of Statistical and Data Sciences, Smith College

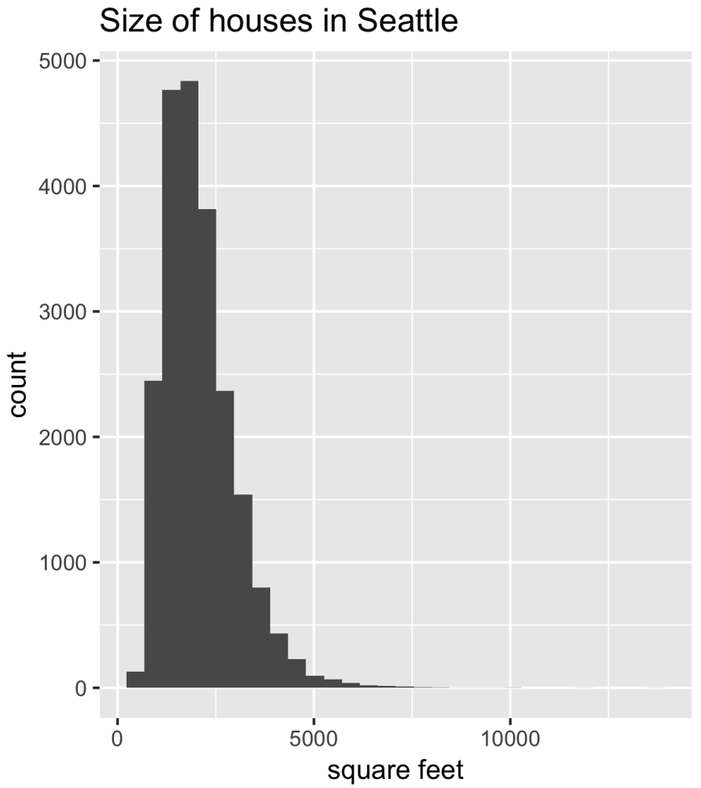
Refresher: Seattle house prices

From the `moderndive` package for ModernDive:

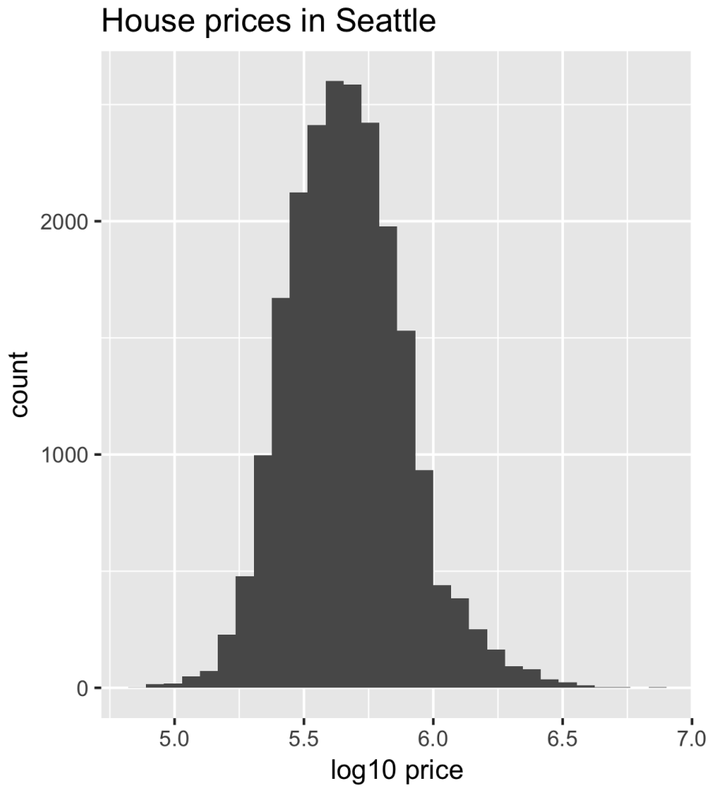
```
library(dplyr)
library(moderndive)

# Preview only certain variables:
house_prices %>%
  select(price, sqft_living, condition, waterfront) %>%
  glimpse()

Observations: 21,613
Variables: 4
$ price      <dbl> 221900, 538000, 180000, 604000, 510000, 1225000, 257500, 291
$ sqft_living <int> 1180, 2570, 770, 1960, 1680, 5420, 1715, 1060, 1780, 1890, 3
$ condition   <fct> 3, 3, 3, 5, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3, 3, 3, 4, 4, 4,
$ waterfront  <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS
```

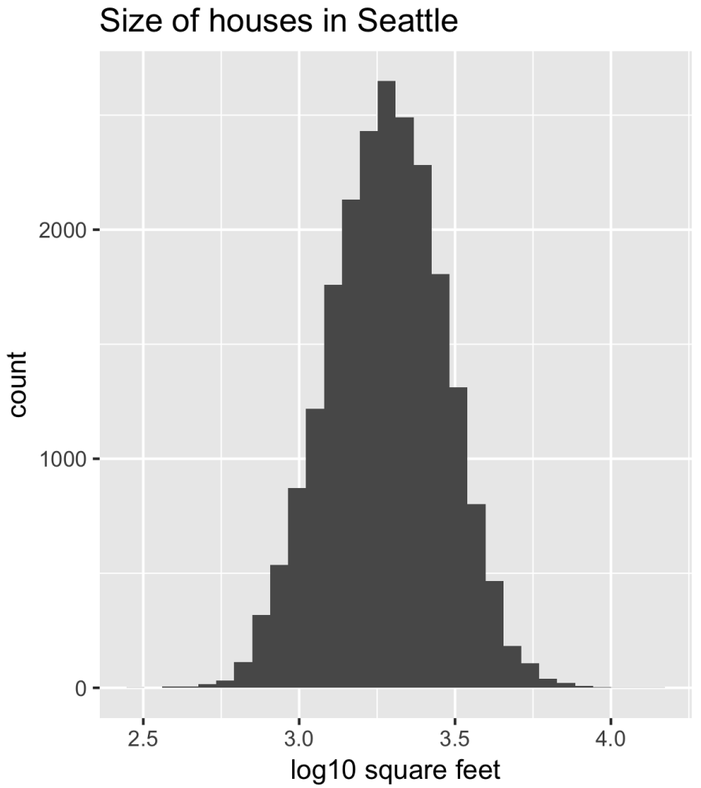
Refresher: Price and size variables





Refresher: log10 transformation





Refresher: Data transformation

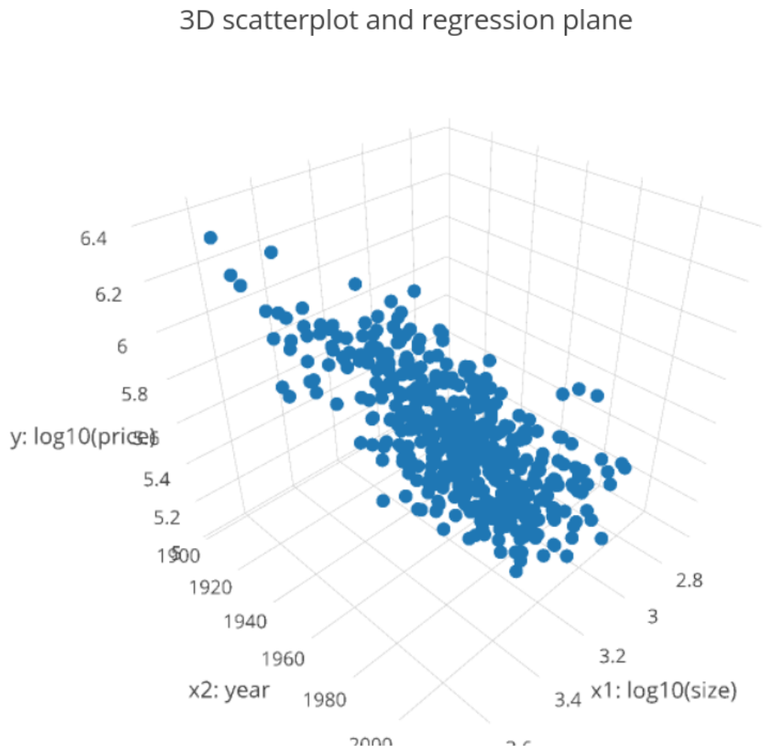
```
# log10() transform price and size
house_prices <- house_prices %>%
  mutate(
    log10_price = log10(price),
    log10_size = log10(sqft_living)
  )
```

Model for house price

- Outcome variable y - house price (USD): `price`
- Two numerical explanatory/predictor variables:
  - x_1 - house size: `log10_size`
  - x_2 - year built: `yr_built`

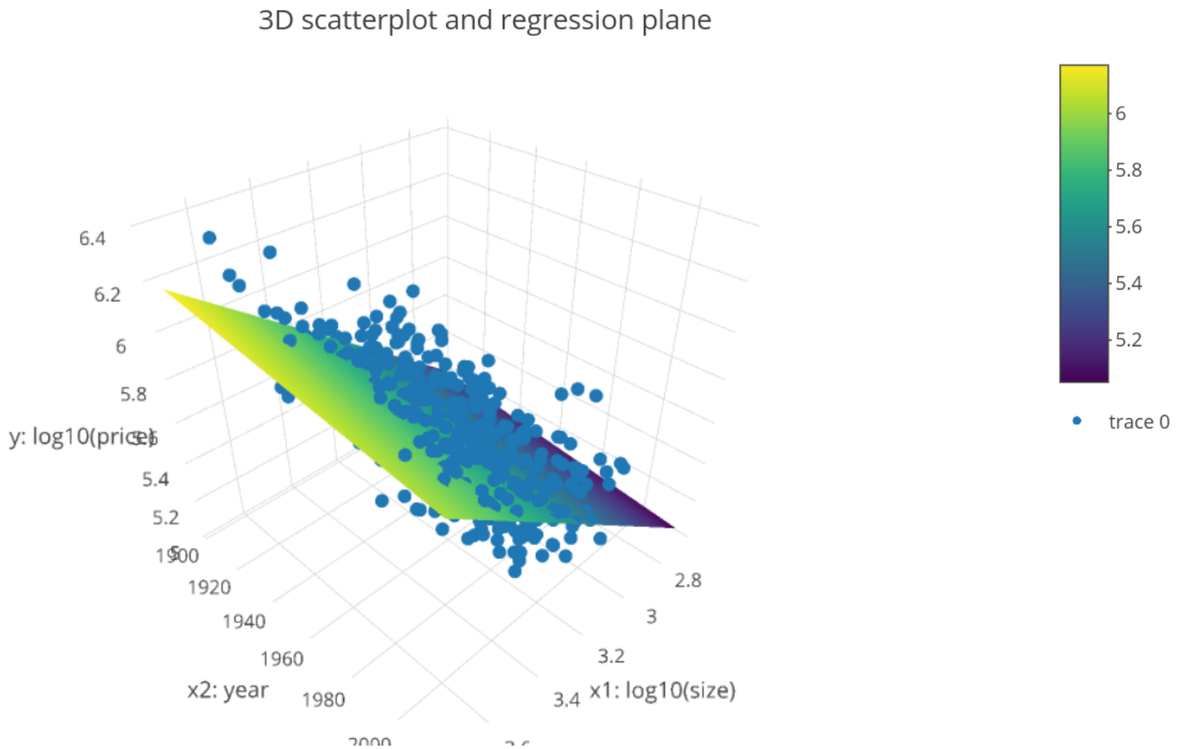
Exploratory visualizing of house price, size & year

3D scatterplot of `log10_price`, `log10_size`, and `yr_built`



Regression plane

3D scatterplot with regression plane (link to interactive version).



Regression table

```
# Fit regression model using formula of form: y ~ x1 + x2
model_price_1 <- lm(log10_price ~ log10_size + yr_built,
                    data = house_prices)

# Output regression table
get_regression_table(model_price_1)

# A tibble: 3 x 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept   5.38      0.0754       71.4       0  5.24      5.53
2 log10_size  0.913     0.00647     141.        0  0.901     0.926
3 yr_built   -0.00138   0.00004     -33.8       0 -0.00146  -0.0013
```

MODELING WITH DATA IN THE TIDYVERSE

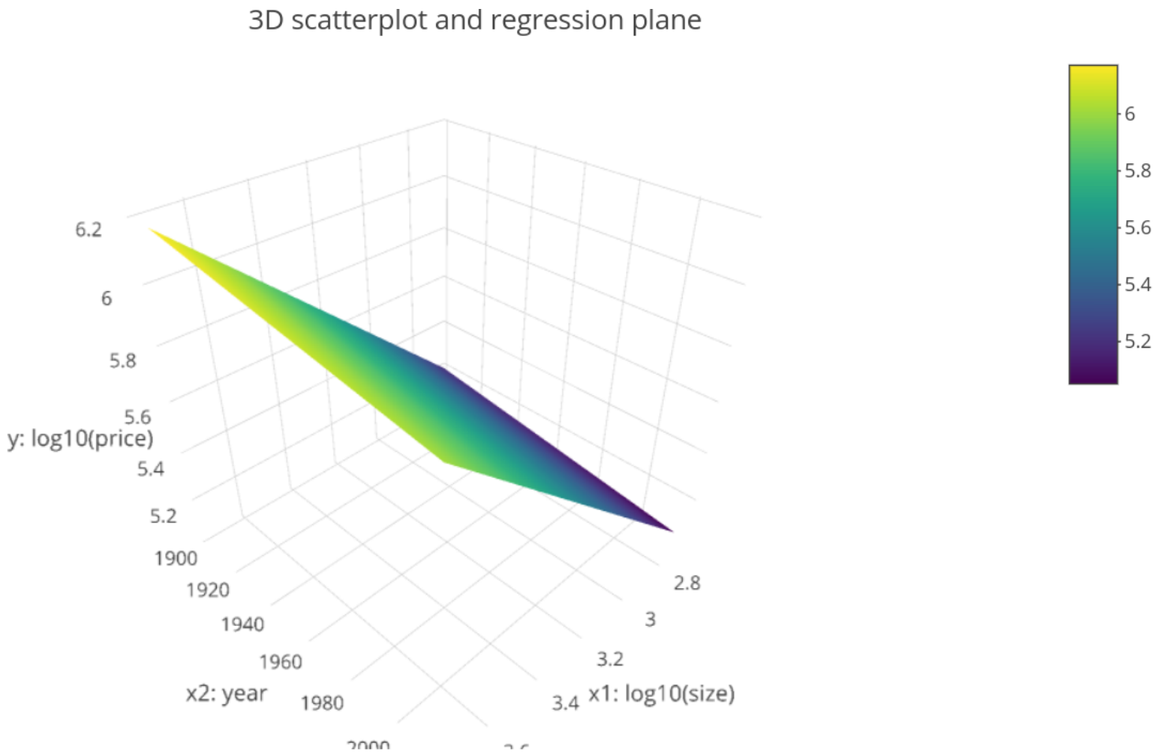
Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

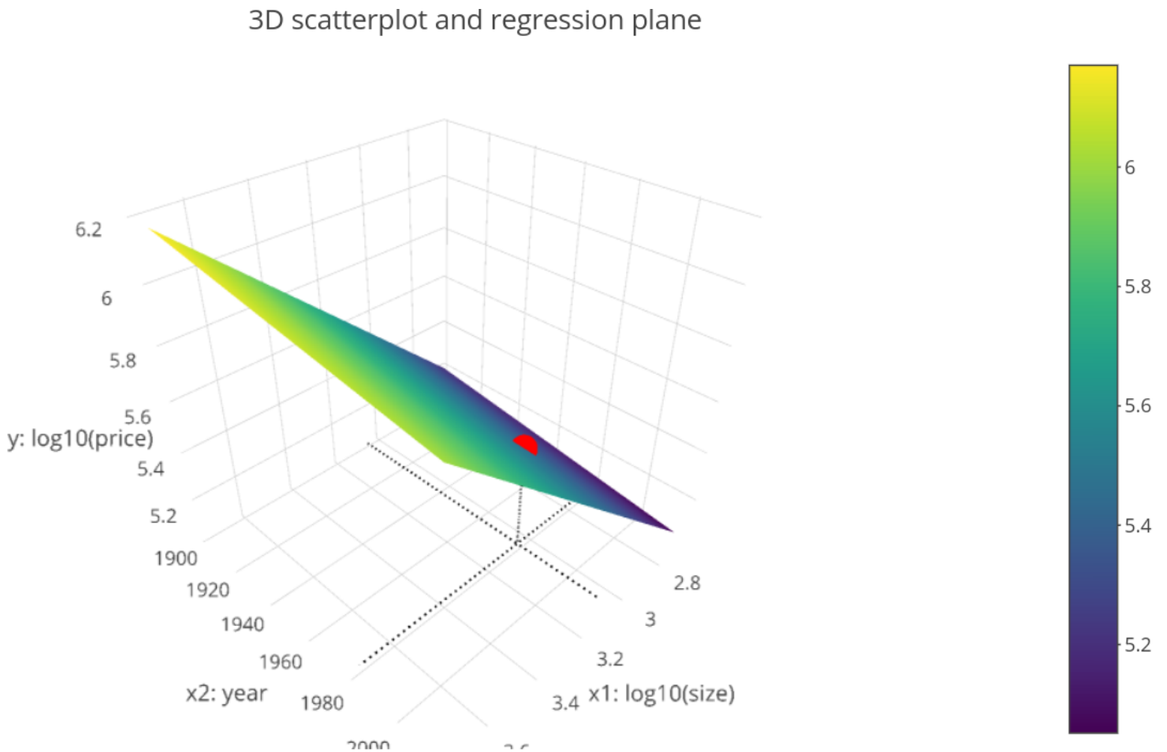
Predicting house price using year & size

Albert Y. Kim
Assistant Professor of Statistical and Data Sciences, Smith College

Refresher: regression plane



Regression plane for prediction



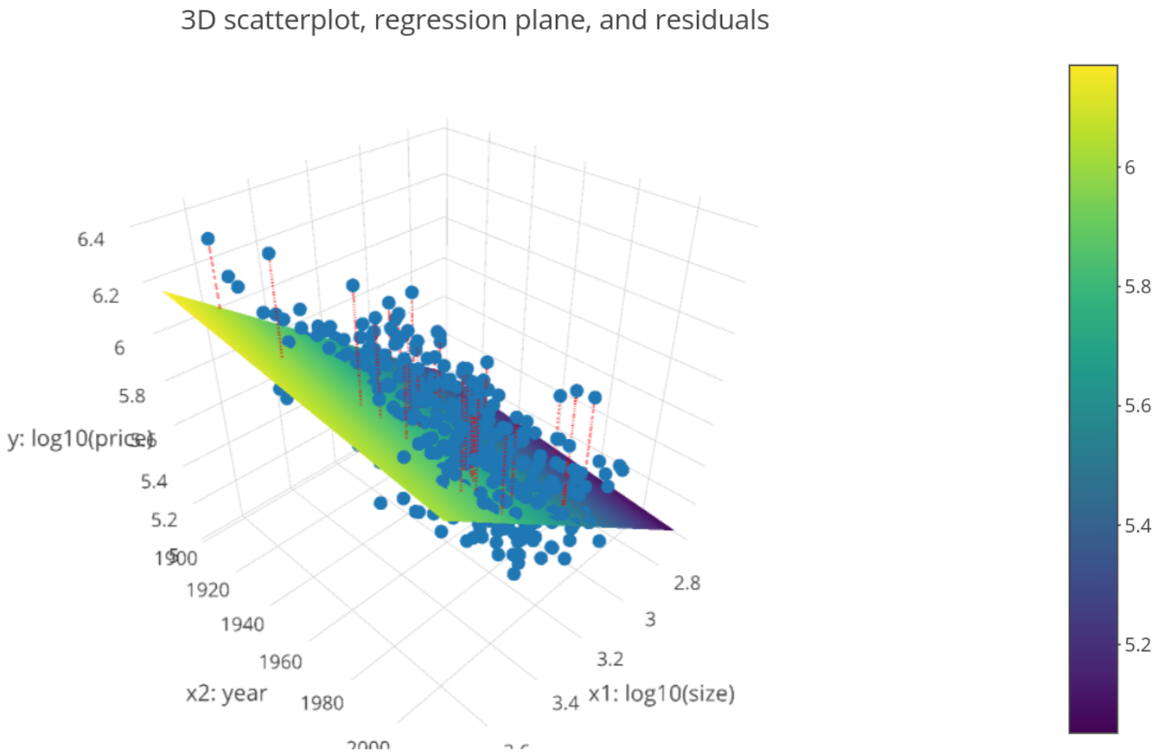
Predicted value

```
# Fit regression model using formula of form: y ~ x1 + x2
model_price_1 <- lm(log10_price ~ log10_size + yr_built,
                    data = house_prices)

# Output regression table
get_regression_table(model_price_1)

# A tibble: 3 x 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept   5.38      0.0754       71.4       0  5.24       5.53
2 log10_size  0.913     0.00647     141.        0  0.901      0.926
3 yr_built   -0.00138   0.00004     -33.8       0 -0.00146  -0.0013

# Make prediction
5.38 + 0.913 * 3.07 - 0.00138 * 1980

5.45051

# Convert back to original untransformed units: US dollars
10^(5.45051)

282169.5
```

Computing all predicted values and residuals

```
# Output point-by-point information
get_regression_points(model_price_1)

# A tibble: 21,613 x 6
      ID log10_price log10_size yr_built log10_price_hat residual
   <int>       <dbl>      <dbl>    <dbl>           <dbl>    <dbl>
 1     1        5.35       3.07     1955            5.50   -0.153
 2     2        5.73       3.41     1951            5.81   -0.083
 3     3        5.26       2.89     1933            5.36   -0.105
 4     4        5.78       3.29     1965            5.69    0.094
 5     5        5.71       3.22     1987            5.60    0.112
 6     6        6.09       3.73     2001            6.04    0.047
 7     7        5.41       3.23     1995            5.59   -0.182
 8     8        5.46       3.02     1963            5.45    0.019
 9     9        5.36       3.25     1960            5.66   -0.295
10    10        5.51       3.28     2003            5.62   -0.111
# ... with 21,603 more rows
```

Best fit and residuals



Sum of squared residuals

```
# Output point-by-point information
get_regression_points(model_price_1)

# A tibble: 21,613 x 6
      ID log10_price log10_size yr_built log10_price_hat residual
   <int>       <dbl>      <dbl>    <dbl>           <dbl>    <dbl>
 1     1        5.35       3.07     1955            5.50   -0.153
 2     2        5.73       3.41     1951            5.81   -0.083
 3     3        5.26       2.89     1933            5.36   -0.105
 4     4        5.78       3.29     1965            5.69    0.094
 5     5        5.71       3.22     1987            5.60    0.112

# Square all residuals and sum them
get_regression_points(model_price_1) %>%
  mutate(sq_residuals = residual^2) %>%
  summarize(sum_sq_residuals = sum(sq_residuals))

# A tibble: 1 x 1
  sum_sq_residuals
             <dbl>
1             585.
```

MODELING WITH DATA IN THE TIDYVERSE

Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

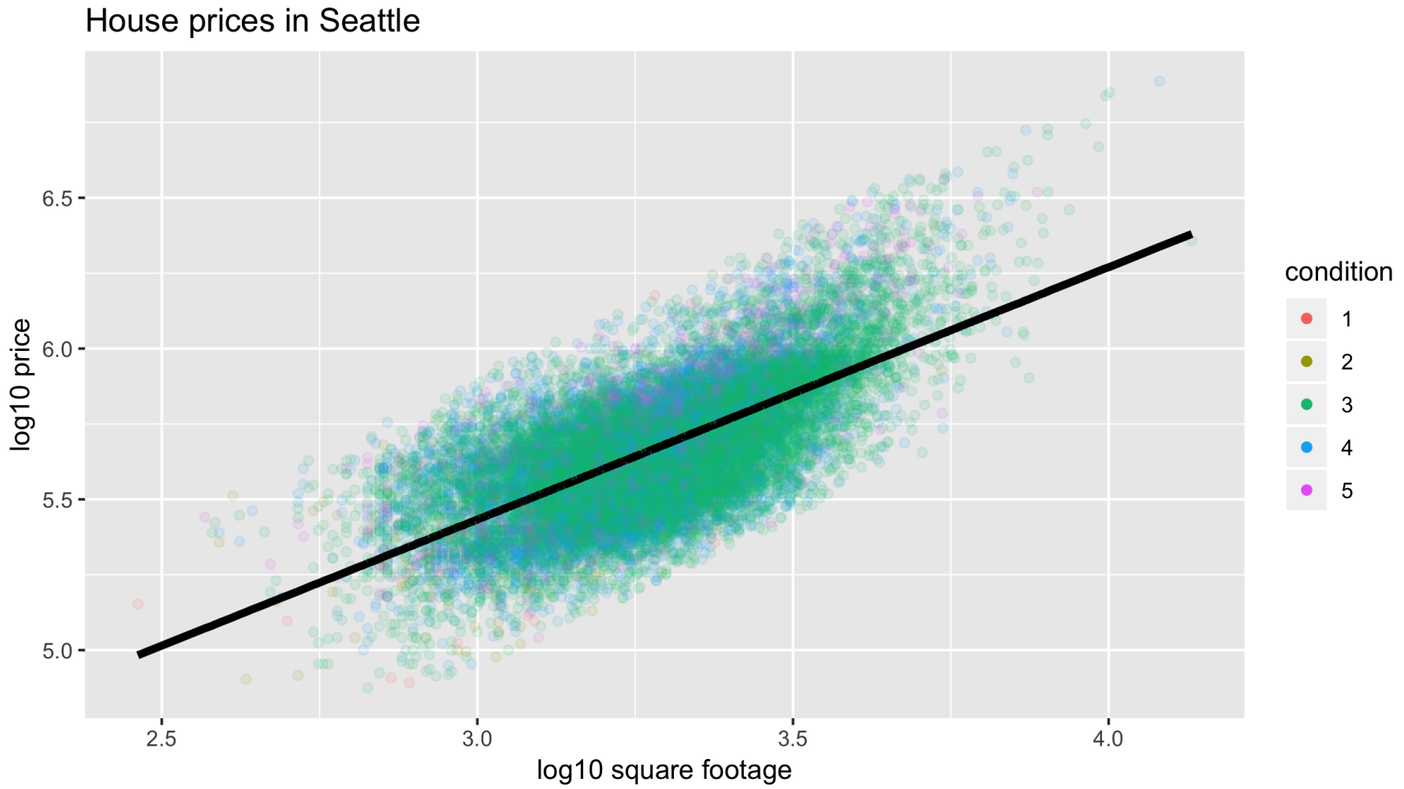
Explaining house price with size & condition

Albert Y. Kim
Assistant Professor of Statistical and Data Sciences, Smith College

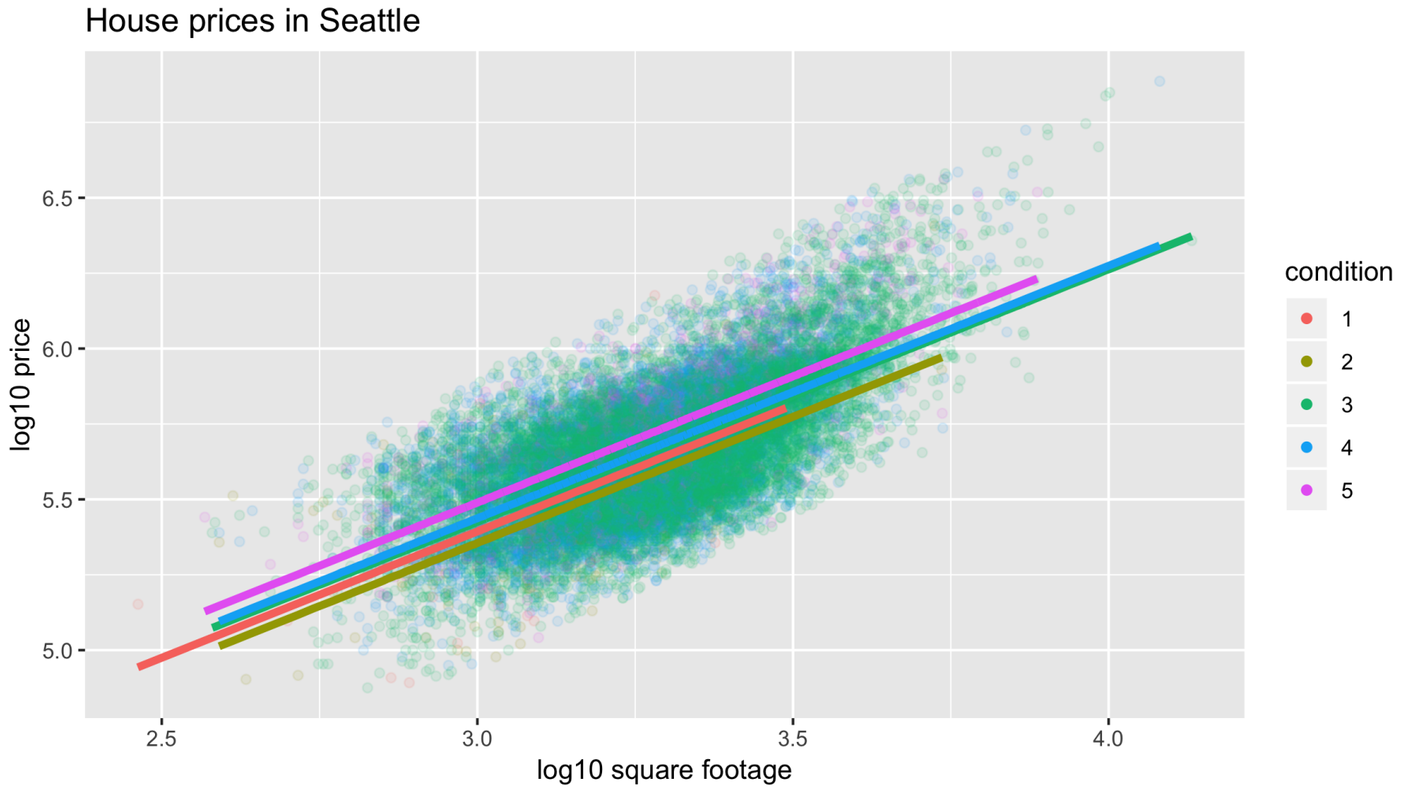
Refresher: Exploratory data analysis

```
library(dplyr)
library(moderndive)

# log transform variables
house_prices <- house_prices %>%
  mutate(
    log10_price = log10(price),
    log10_size = log10(sqft_living)
  )

# Group mean & sd of log10_price and counts
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
```

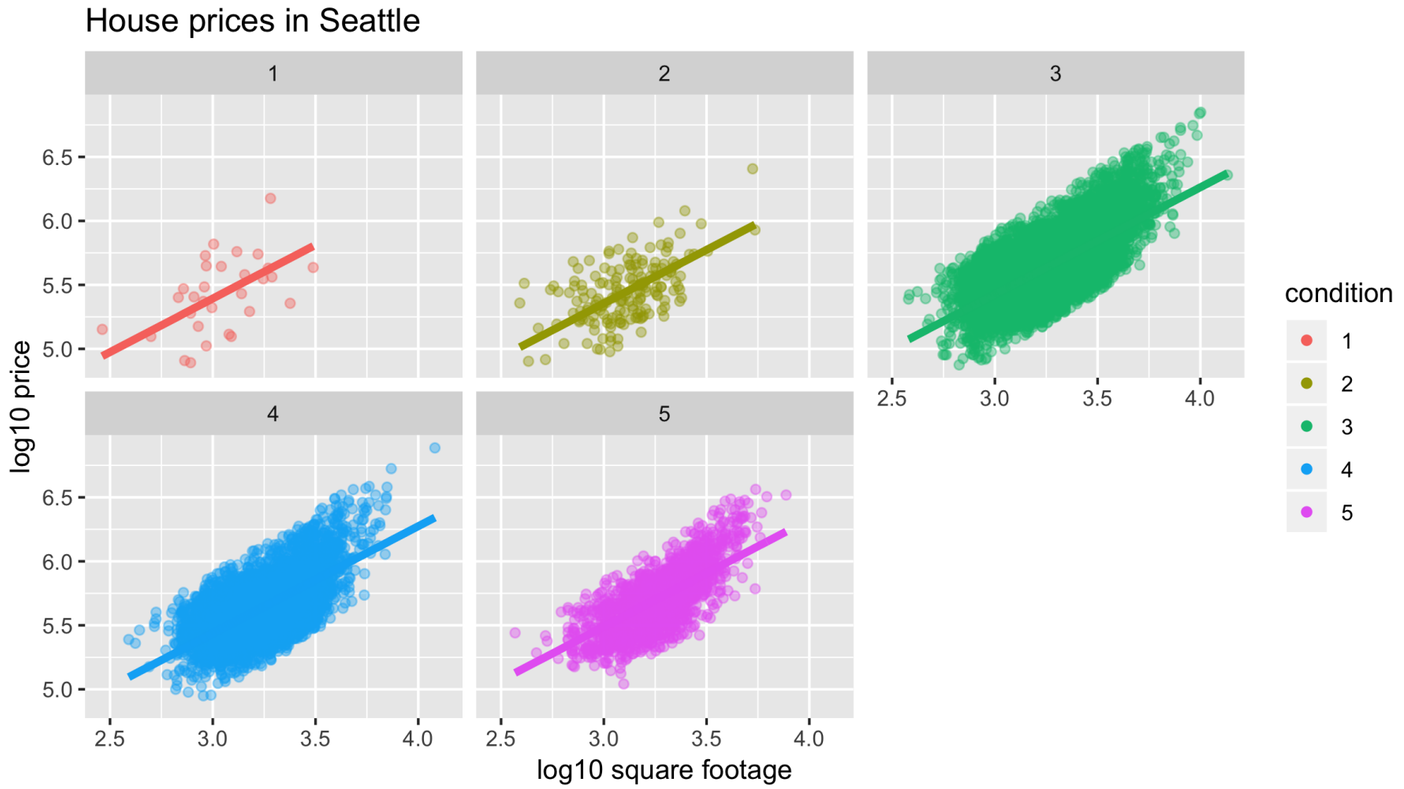
Exploratory visualization house price, size & condition



Parallel slopes model



Parallel slopes model



Quantifying relationship between house price, size & condition

```
# Fit regression model using formula of form: y ~ x1 + x2
model_price_3 <- lm(log10_price ~ log10_size + condition,
                    data = house_prices)

# Output regression table
get_regression_table(model_price_3)

# A tibble: 6 x 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept     2.88      0.036     80.0    0         2.81     2.95
2 log10_size    0.837     0.006    134.     0         0.825    0.85
3 condition2   -0.039     0.033     -1.16   0.246    -0.104    0.027
4 condition3    0.032     0.031      1.04   0.3      -0.028    0.092
5 condition4    0.044     0.031      1.42   0.155    -0.017    0.104
6 condition5    0.096     0.031      3.09   0.002     0.035    0.156
```

MODELING WITH DATA IN THE TIDYVERSE

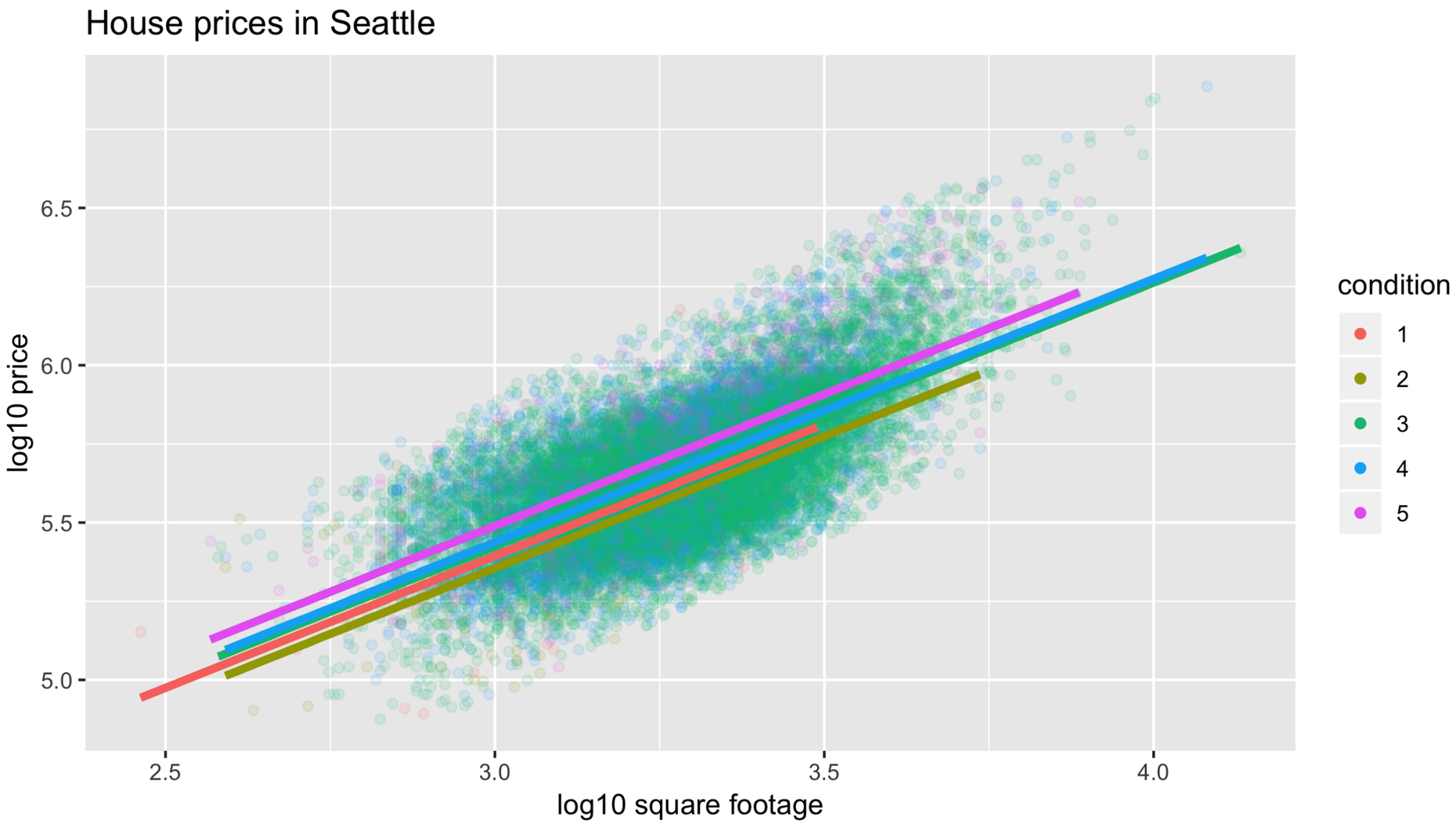
Let's practice!

MODELING WITH DATA IN THE TIDYVERSE

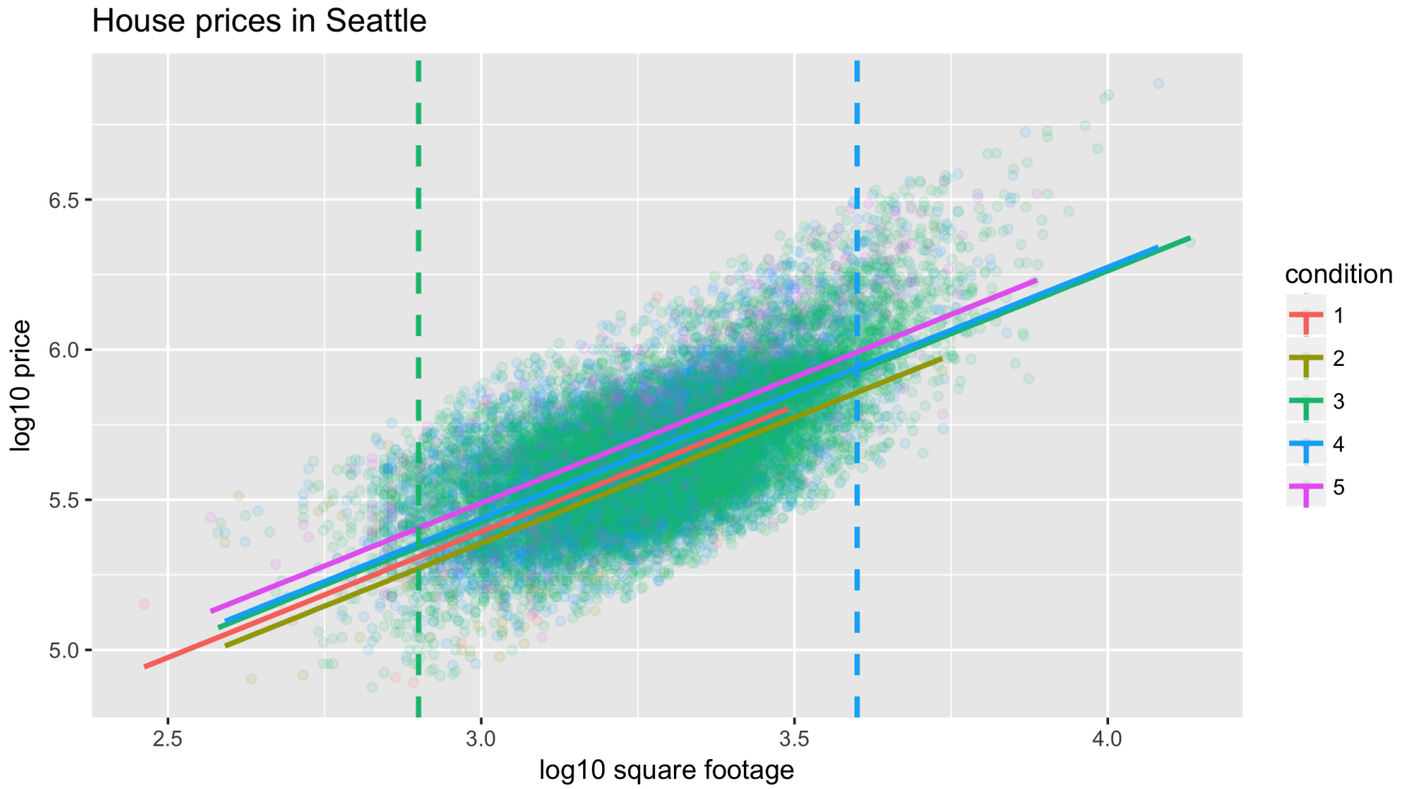
Predicting house price using size & condition

Albert Y. Kim
Assistant Professor of Statistical and Data Sciences, Smith College

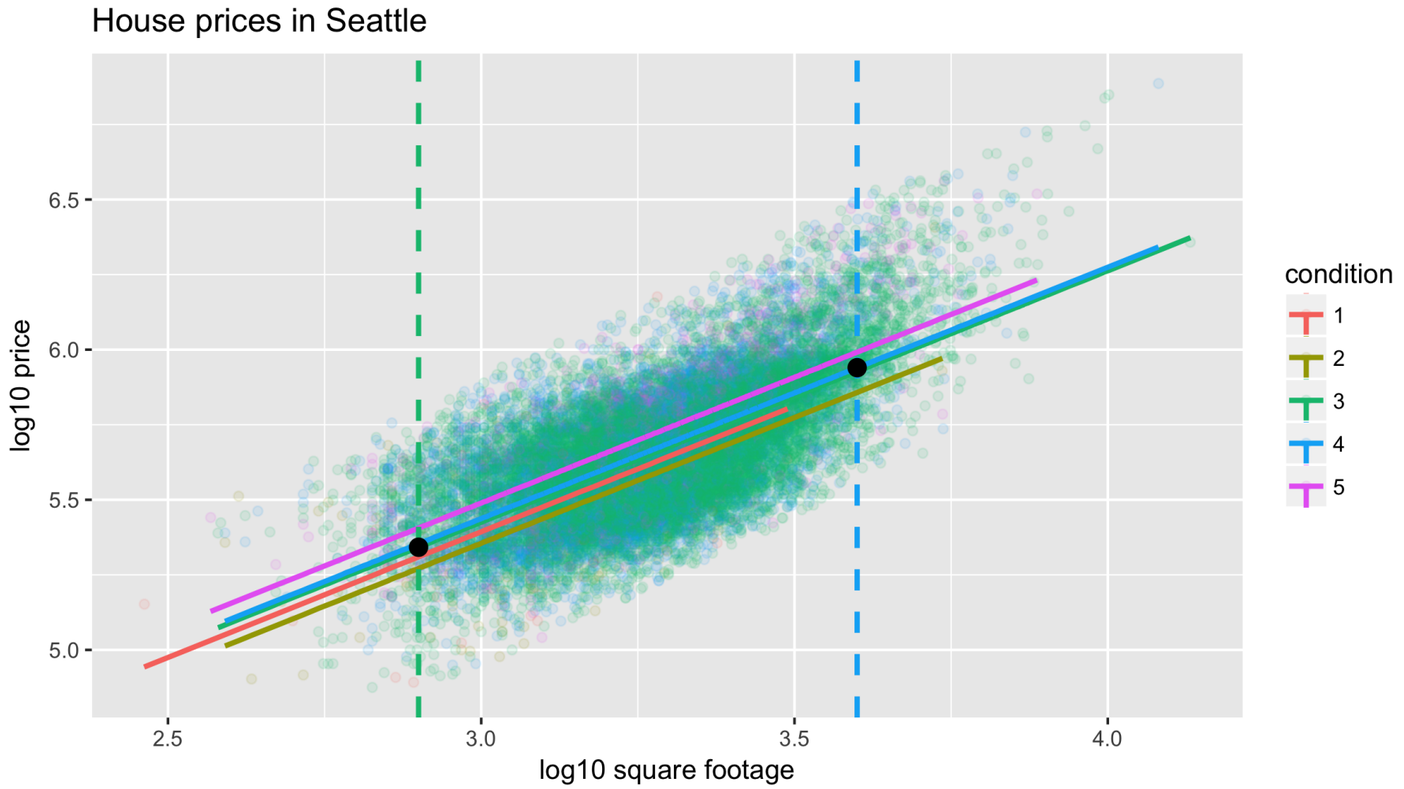
Refresher: Parallel Slopes



Making a prediction



Visualizing predictions



Numerical predictions

Using values in `estimate` in regression table below:

- First house: $\hat{y} = 2.88 + 0.032 + 0.837 \cdot 2.90 = 5.34$
- Second house: $\hat{y} = 2.88 + 0.044 + 0.837 \cdot 3.60 = 5.94$

```
# Fit regression model and get regression table
model_price_3 <- lm(log10_price ~ log10_size + condition,
                    data = house_prices)
get_regression_table(model_price_3)

# A tibble: 6 x 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept     2.88      0.036     80.0    0         2.81     2.95
2 log10_size    0.837     0.006    134.     0         0.825    0.85
3 condition2   -0.039     0.033     -1.16   0.246    -0.104    0.027
4 condition3    0.032     0.031      1.04   0.3      -0.028    0.092
5 condition4    0.044     0.031      1.42   0.155    -0.017    0.104
6 condition5    0.096     0.031      3.09   0.002     0.035    0.156
```

Defining "new" data

```
# Create data frame of "new" houses
new_houses <- data_frame(
  log10_size = c(2.9, 3.6),
  condition = factor(c(3, 4))
)
new_houses

# A tibble: 2 x 2
  log10_size condition
       <dbl> <fct>
1        2.9 3
2        3.6 4
```

Making predictions using new data

```
# Make predictions on new data
get_regression_points(model_price_3, newdata = new_houses)

# A tibble: 2 x 4
     ID log10_size condition log10_price_hat
  <int>      <dbl> <fct>               <dbl>
1     1        2.9 3                    5.34
2     2        3.6 4                    5.94

# Make predictions in original units by undoing log10()
get_regression_points(model_price_3, newdata = new_houses) %>%
  mutate(price_hat = 10^log10_price_hat)

# A tibble: 2 x 5
     ID log10_size condition log10_price_hat price_hat
  <int>      <dbl> <fct>               <dbl>     <dbl>
1     1        2.9 3                    5.34   219786.
2     2        3.6 4                    5.94   870964.
```

MODELING WITH DATA IN THE TIDYVERSE

Let's practice!