Formalizing Linear Models

Dr. Maria Tackett

10.15.19

STA 199

[Click for PDF of slides]

Announcements

- Complete Reading 05 (if you haven't already done so)
- Project topic ideas **due Wednesday at 11:59p**

Characterizing relationships with models

Data & packages

```r
library(tidyverse)
library(broom)
```

```r
pp <- read_csv("data/paris_paintings.csv",
               na = c("n/a", "", "NA"))
```

Want to follow along?

Go to RStudio Cloud -> make a copy of "Modeling Paris Paintings"

Height & width

```
(m_ht_wt <- lm(Height_in ~ Width_in, data = pp))
```

```
##
## Call:
## lm(formula = Height_in ~ Width_in, data = pp)
##
## Coefficients:
## (Intercept)     Width_in
##      3.6214       0.7808
```

$$\widehat{Height_{in}} = 3.62 + 0.78\ Width_{in}$$

- **Slope**: For each additional inch the painting is wider, the height is expected to be higher, on average, by 0.78 inches.
- **Intercept**: Paintings that are 0 inches wide are expected to be 3.62 inches high, on average.
  - This is a nonsense interpretation!

The linear model with a single predictor

- We're interested in the β_0 (population parameter for the intercept) and the β_1 (population parameter for the slope) in the following model:

$$\hat{y} = \beta_0 + \beta_1~x$$

- Tough luck, you can't have them...

- So we use the sample statistics to estimate them:

$$\hat{y} = b_0 + b_1~x$$

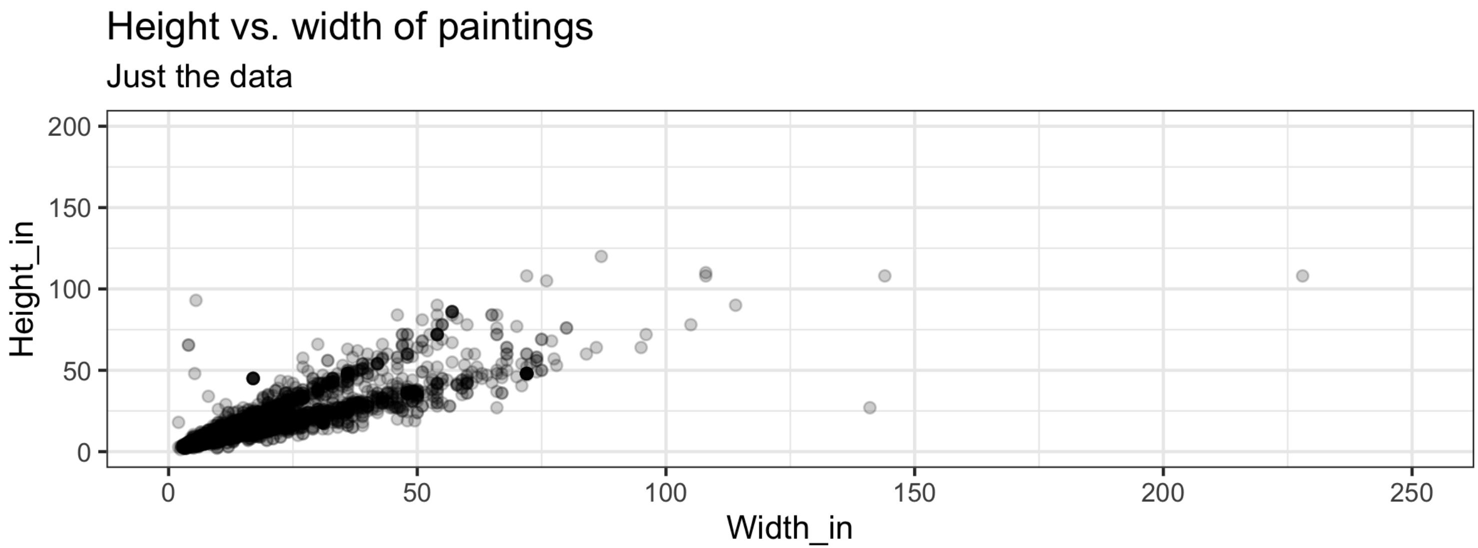
Least squares regression

The regression line minimizes the sum of squared residuals.

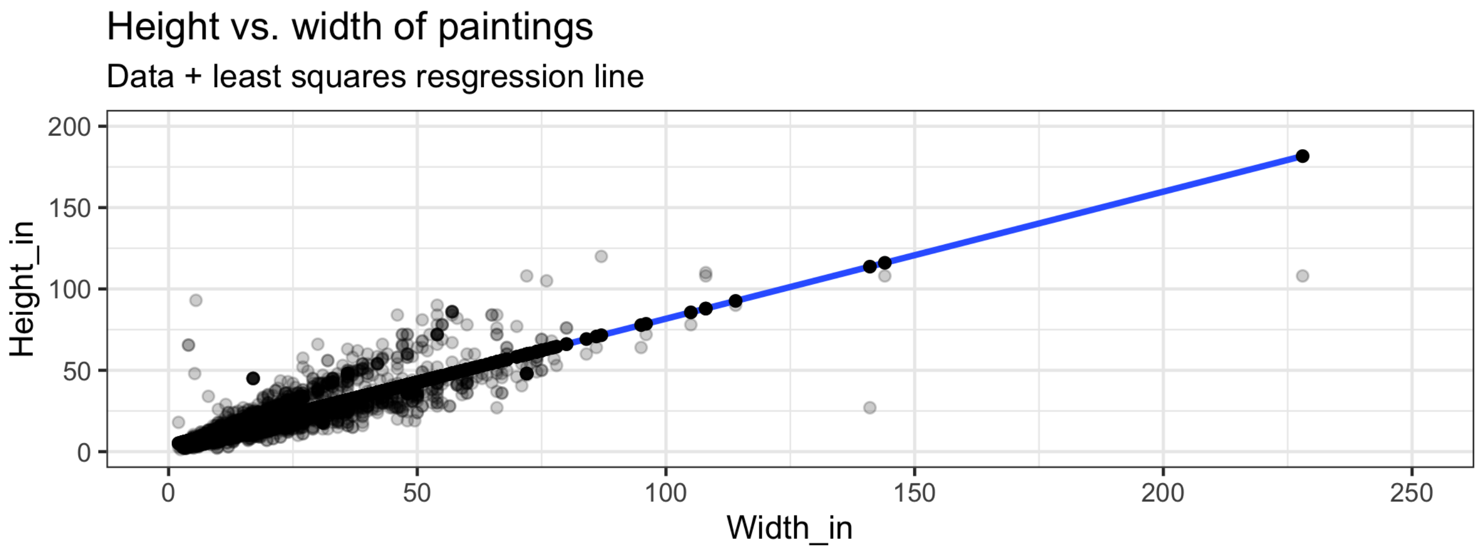
If $e_i = y - \hat{y}$,

then, the regression line minimizes $\sum_{i=1}^{n} e_i^2$.

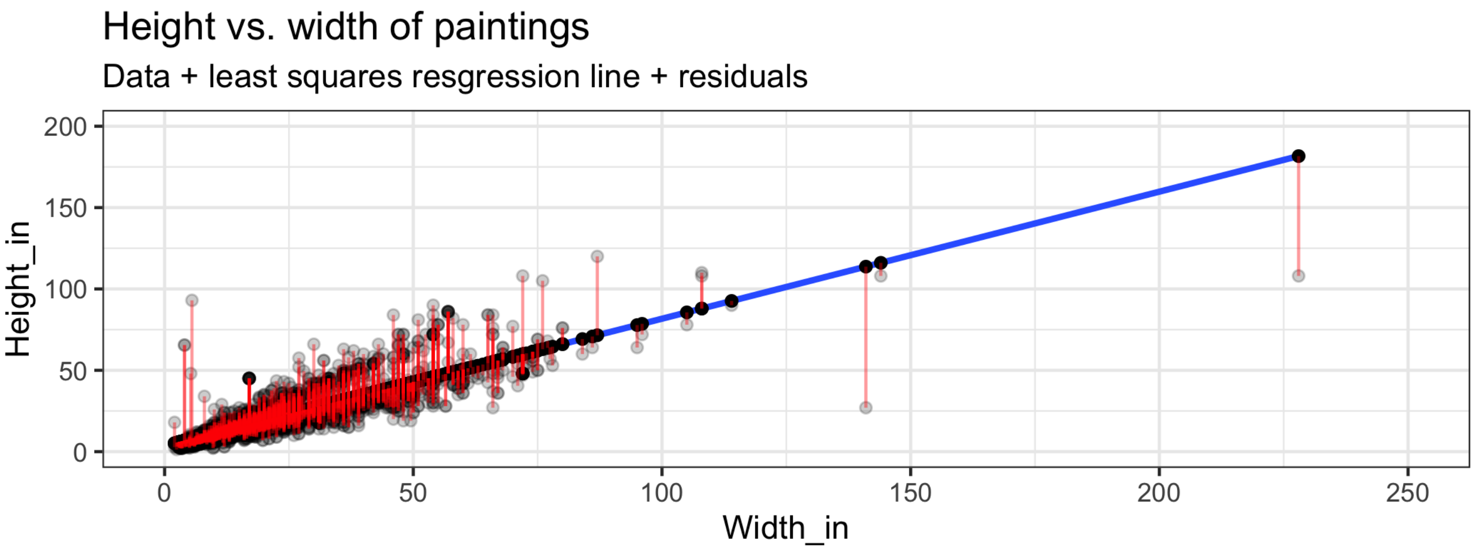
Visualizing residuals



Visualizing residuals (cont.)



Visualizing residuals (cont.)



Properties of the least squares regression line

- The slope has the same sign as the correlation coefficient:

$$b_1 = r\frac{s_y}{s_x}$$

- The regression line goes through the center of mass point, the coordinates corresponding to average x and average y: $(\bar{x}, \bar{y})$.

$$\hat{y} = b_0 + b_1 x \quad \Rightarrow \quad b_0 = \bar{y} - b_1 \bar{x}$$

Properties of the least squares regression line

- The sum of the residuals is zero:

$$\sum_{i=1}^{n} e_i = 0$$

- The residuals and x values are uncorrelated.

Height & landscape features

```r
(m_ht_lands <- lm(Height_in ~ factor(landsALL), data = pp))
```

```
##
## Call:
## lm(formula = Height_in ~ factor(landsALL), data = pp)
##
## Coefficients:
##      (Intercept)  factor(landsALL)1
##           22.680             -5.645
```

$$\widehat{Height}_{in} = 22.68 - 5.65\ landsALL$$

Height & landscape features (cont.)

- Slope: Paintings with landscape features are expected, on average, to be 5.65 inches shorter than paintings that without landscape features.
    - Compares baseline level (`landsALL = 0`) to other level (`landsALL = 1`).
- Intercept: Paintings that don't have landscape features are expected, on average, to be 22.68 inches tall.

Categorical predictor with 2 levels

```
## # A tibble: 8 x 3
##   name     price landsALL
##   <chr>    <dbl>    <dbl>
## 1 L1764-2    360        0
## 2 L1764-3      6        0
## 3 L1764-4     12        1
## 4 L1764-5a     6        1
## 5 L1764-5b     6        1
## 6 L1764-6      9        0
## 7 L1764-7a    12        0
## 8 L1764-7b    12        0
```

Relationship between height and school

```
(m_ht_sch <- lm(Height_in ~ school_pntg, data = pp))
```

```
##
## Call:
## lm(formula = Height_in ~ school_pntg, data = pp)
##
## Coefficients:
##     (Intercept)  school_pntgD/FL     school_pntgF     school_pntgG
##          14.000            2.329           10.197            1.650
##    school_pntgI     school_pntgS     school_pntgX
##          10.287           30.429            2.869
```

- When the categorical explanatory variable has many levels, they're encoded to dummy (indicator) variables.

- Each coefficient describes the expected difference between heights in that particular school compared to the baseline level.

Categorical predictor with >2 levels

Show 10 entries Search:

	school_pntg	D_FL	F	G	I	S	X
1	A	0	0	0	0	0	0
2	D/FL	1	0	0	0	0	0
3	F	0	1	0	0	0	0
4	G	0	0	1	0	0	0
5	I	0	0	0	1	0	0
6	S	0	0	0	0	1	0
7	X	0	0	0	0	0	1

Showing 1 to 7 of 7 entries

Previous 1 Next

Correlation does not imply causation!

Remember this when interpreting model coefficients

Prediction with models

Predict height from width

On average, how tall are paintings that are 60 inches wide?

$$\widehat{Height_{in}} = 3.62 + 0.78\ Width_{in}$$

```
3.62 + 0.78 * 60
```

```
## [1] 50.42
```

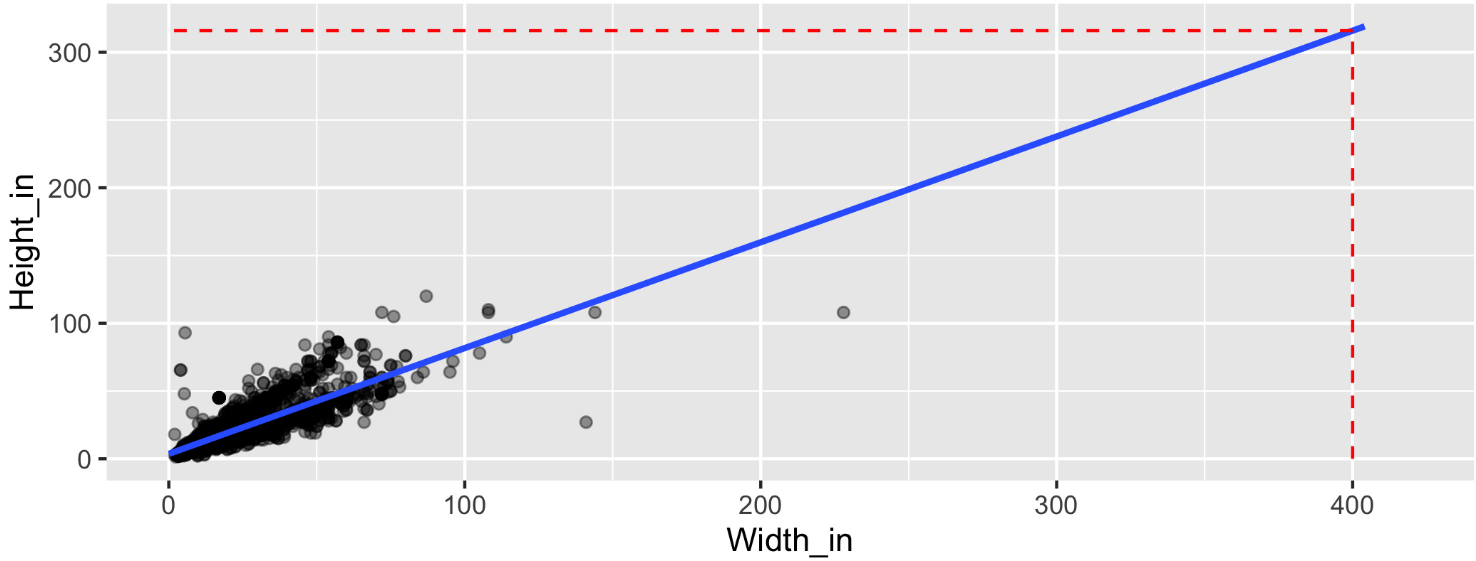
"On average, we expect paintings that are 60 inches wide to be 50.42 inches high."

Warning: We "expect" this to happen, but there will be some variability. (We'll learn about measuring the variability around the prediction later.)

Prediction vs. extrapolation

On average, how tall are paintings that are 400 inches wide?

$$\widehat{Height_{in}} = 3.62 + 0.78\ Width_{in}$$



Watch out for extrapolation!

> "When those blizzards hit the East Coast this winter, it proved to my satisfaction that global warming was a fraud. That snow was freezing cold. But in an alarming trend, temperatures this spring have risen. Consider this: On February 6th it was 10 degrees. Today it hit almost 80. At this rate, by August it will be 220 degrees. So clearly folks the climate debate rages on."[1]
> Stephen Colbert, April 6th, 2010

[1] OpenIntro Statistics. "Extrapolation is treacherous." OpenIntro Statistics.

Measuring model fit

Measuring the strength of the fit

- The strength of the fit of a linear model is most commonly evaluated using R^2.
- It tells us what percent of variability in the response variable is explained by the model.
- The remainder of the variability is explained by variables not included in the model.
- R^2 is sometimes called the coefficient of determination.

Obtaining R^2 in R

- Height vs. width

```
glance(m_ht_wt)
```

```
## # A tibble: 1 x 11
##   r.squared adj.r.squared sigma statistic p.value    df  logLik    AIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>   <dbl>  <dbl>
## 1     0.683         0.683  8.30     6749.       0     2 -11083. 22173.
## # … with 3 more variables: BIC <dbl>, deviance <dbl>, df.residual <int>
```

```
glance(m_ht_wt)$r.squared # extract R-squared
```

```
## [1] 0.6829468
```

Roughly 68% of the variability in heights of paintings can be explained by their widths.

Tidy regression output

Let's revisit the model predicting heights of paintings from their widths:

```
m_ht_wt <- lm(Height_in ~ Width_in, data = pp)
```

Not-so-tidy regression output

- You might come across these as you read work from others, but we'll try to stay away from them
- Not because they are wrong, but because they don't result in tidy data frames as results.

Not-so-tidy regression output (1)

Option 1:

```
m_ht_wt
```

```
## 
## Call:
## lm(formula = Height_in ~ Width_in, data = pp)
## 
## Coefficients:
## (Intercept)     Width_in  
##      3.6214       0.7808
```

Not-so-tidy regression output (2)

Option 2:

```
summary(m_ht_wt)
```

```
##
## Call:
## lm(formula = Height_in ~ Width_in, data = pp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -86.714  -4.384  -2.422   3.169  85.084
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.621406   0.253860   14.27   <2e-16 ***
## Width_in    0.780796   0.009505   82.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.304 on 3133 degrees of freedom
##   (258 observations deleted due to missingness)
## Multiple R-squared:  0.6829,    Adjusted R-squared:  0.6828
## F-statistic:  6749 on 1 and 3133 DF,  p-value: < 2.2e-16
```

Review

What makes a data frame tidy?

1. Each variable forms a column.
2. Each observation forms a row.
3. Each type of observational unit forms a table.

Tidy regression output

Achieved with functions from the broom package:

- **`tidy`**: Constructs a data frame that summarizes the model's statistical findings: coefficient estimates, *standard errors, test statistics, p-values*.
- **`augment`**: Adds columns to the original data that was modeled. This includes predictions and residuals.
- **`glance`**: Constructs a concise one-row summary of the model. This typically contains values such as R^2, adjusted R^2, *and residual standard error that are computed once for the entire model*.

Tidy your model's statistical findings

```r
tidy(m_ht_wt)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    3.62    0.254        14.3 8.82e-45
## 2 Width_in       0.781   0.00950      82.1 0.
```

```r
tidy(m_ht_wt) %>%
  select(term, estimate)
```

```
## # A tibble: 2 x 2
##   term        estimate
##   <chr>          <dbl>
## 1 (Intercept)    3.62
## 2 Width_in       0.781
```

Augment data with model results

New variables of note (for now):

- **`.fitted`**: Predicted value of the response variable
- **`.resid`**: Residuals

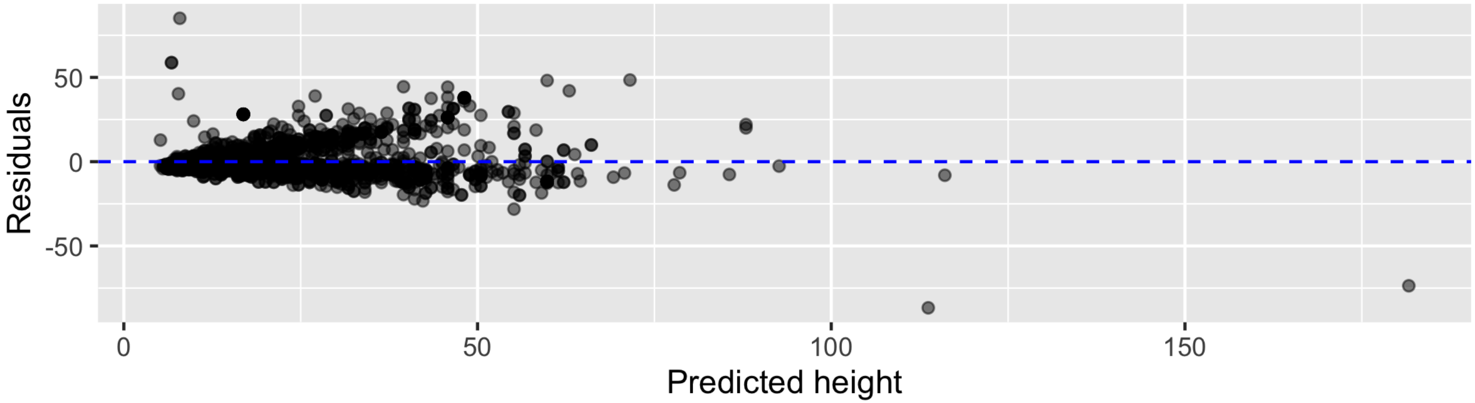
```
augment(m_ht_wt) %>%
  slice(1:5)
```

```
## # A tibble: 5 x 10
##   .rownames Height_in Width_in .fitted .se.fit .resid     .hat .sigma
##   <chr>         <dbl>    <dbl>   <dbl>   <dbl>  <dbl>    <dbl>  <dbl>
## 1 1                37     29.5    26.7   0.166  10.3  3.99e-4   8.30
## 2 2                18     14      14.6   0.165   3.45 3.96e-4   8.31
## 3 3                13     16      16.1   0.158  -3.11 3.61e-4   8.31
## 4 4                14     18      17.7   0.152  -3.68 3.37e-4   8.31
## 5 5                14     18      17.7   0.152  -3.68 3.37e-4   8.31
## # … with 2 more variables: .cooksd <dbl>, .std.resid <dbl>
```

Why might we be interested in these new variables?

Residuals plot

```
m_ht_wt_aug <- augment(m_ht_wt)
ggplot(m_ht_wt_aug, mapping = aes(x = .fitted, y = .resid)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, color = "blue", lty = 2) +
  labs(x = "Predicted height", y = "Residuals")
```



What does this plot tell us about the fit of the linear model?

Glance to assess model fit

```
glance(m_ht_wt)
```

```
## # A tibble: 1 x 11
##   r.squared adj.r.squared sigma statistic p.value    df  logLik    AIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>   <dbl>  <dbl>
## 1     0.683         0.683  8.30     6749.       0     2 -11083. 22173.
## # … with 3 more variables: BIC <dbl>, deviance <dbl>, df.residual <int>
```

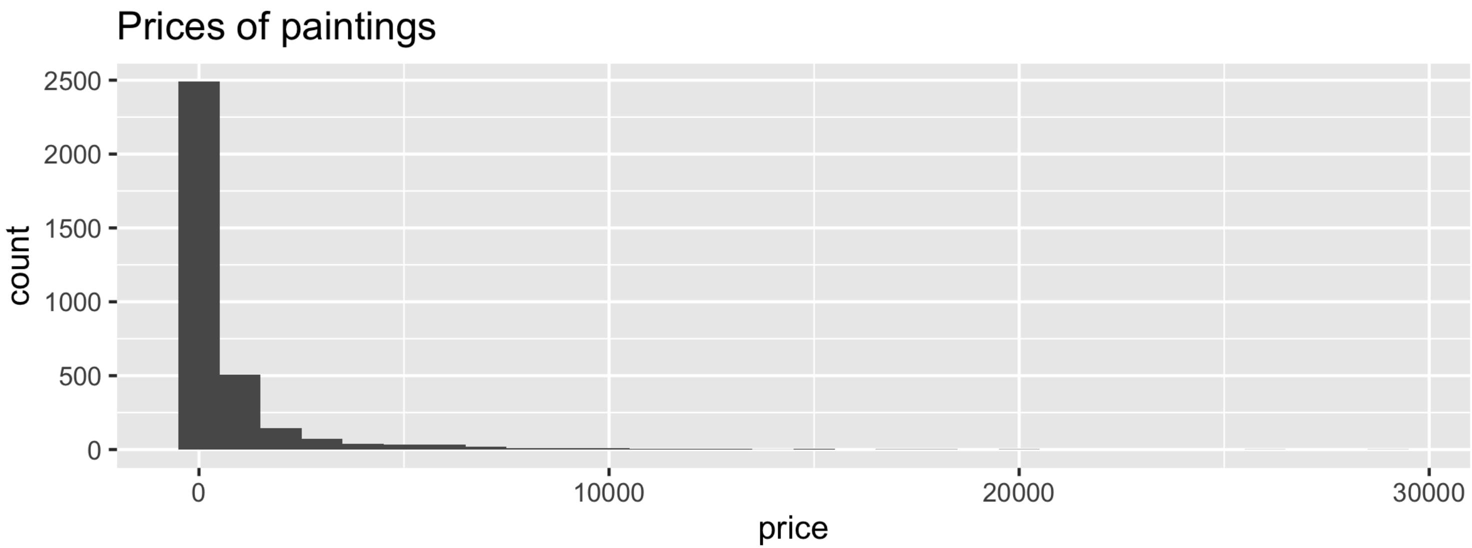
```
glance(m_ht_wt)$r.squared
```

```
## [1] 0.6829468
```

The R^2 is 68.29%.

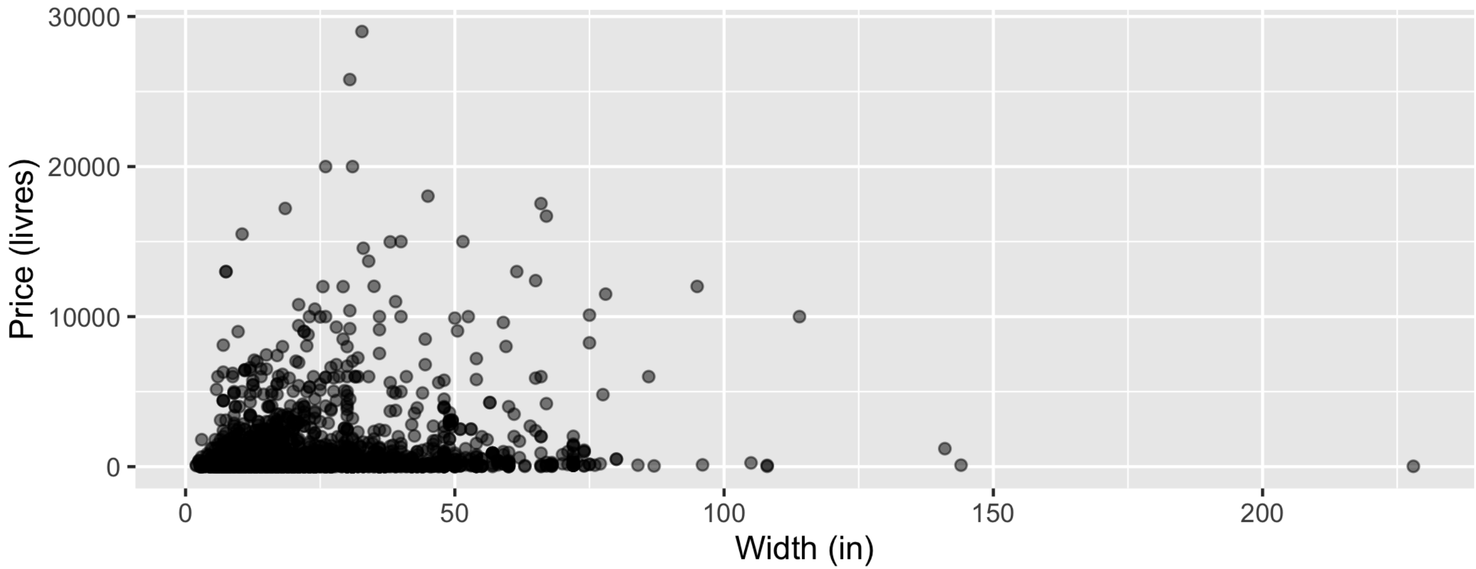
Exploring linearity

Data: Paris Paintings



Price vs. width

Describe the relationship between price and width of painting.

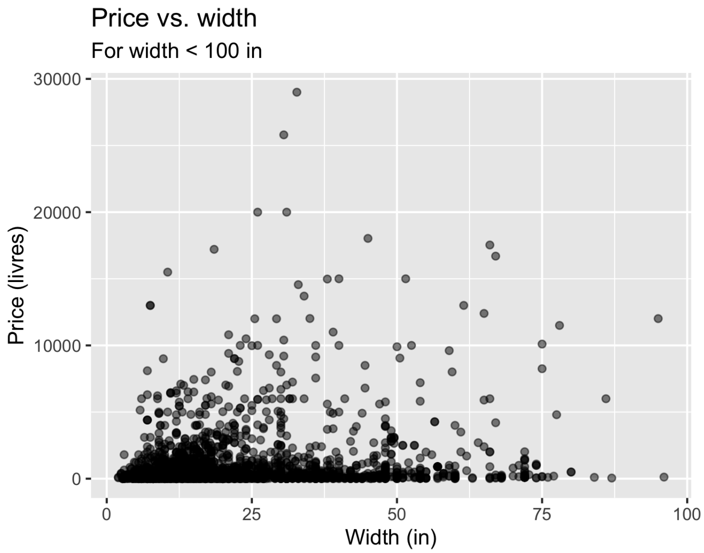


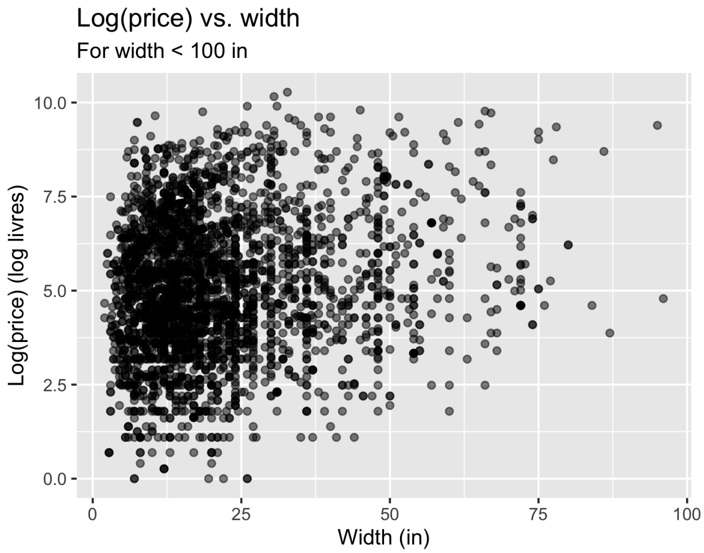
Let's focus on paintings with `Width_in < 100`

```
pp_wt_lt_100 <- pp %>%
  filter(Width_in < 100)
```

Price vs. width

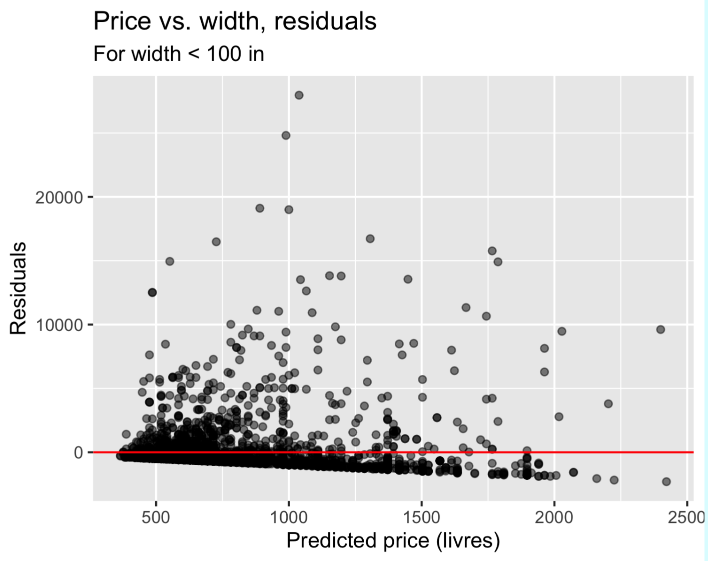
Which plot shows a more linear relationship?

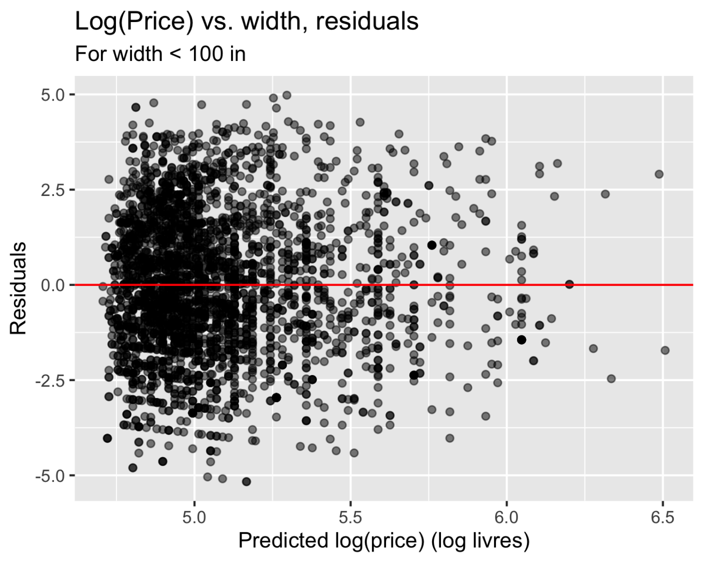




Price vs. width, residuals

Which plot shows a residuals that are uncorrelated with predicted values from the model?





What's the unit of residuals?

Transforming the data

- We saw that `price` has a right-skewed distribution, and the relationship between price and width of painting is non-linear.
- We also observed signs of the model violation, non-constant variance.
- In these situations a transformation applied to the response variable (y) may be useful.
    - The most common transformation is the log transformation $(\log(y) = ln(y))$
- This is beyond the scope of the course, but I'm happy to provide guidance if you want to try modeling a response that requires transformation in your final project