Correlation and Regression

Stat 250

Click here for PDF version

Overview: Fuel Economy Data

- Many factors go into determining what gas mileage a car will achieve
- For now, we will focus on the weight of a car
- It’s generally understood that heavier cars will get worse fuel economy, but it is not clear how much of an increase in weight will lead to a decrease in fuel economy

Overview: Fuel Economy Data

```
head(mpg)
##    weight mpg
## 1    3436  18
## 2    3433  16
## 3    3449  17
## 4    3086  14
## 5    2372  24
## 6    2833  22

dim(mpg)
## [1] 289    2
```

Notation:

y_i = MPG for i^{th} car

x_i = weight for i^{th} car

$i = 1, \ldots, n$

Exploratory Data Analysis

Scatterplots

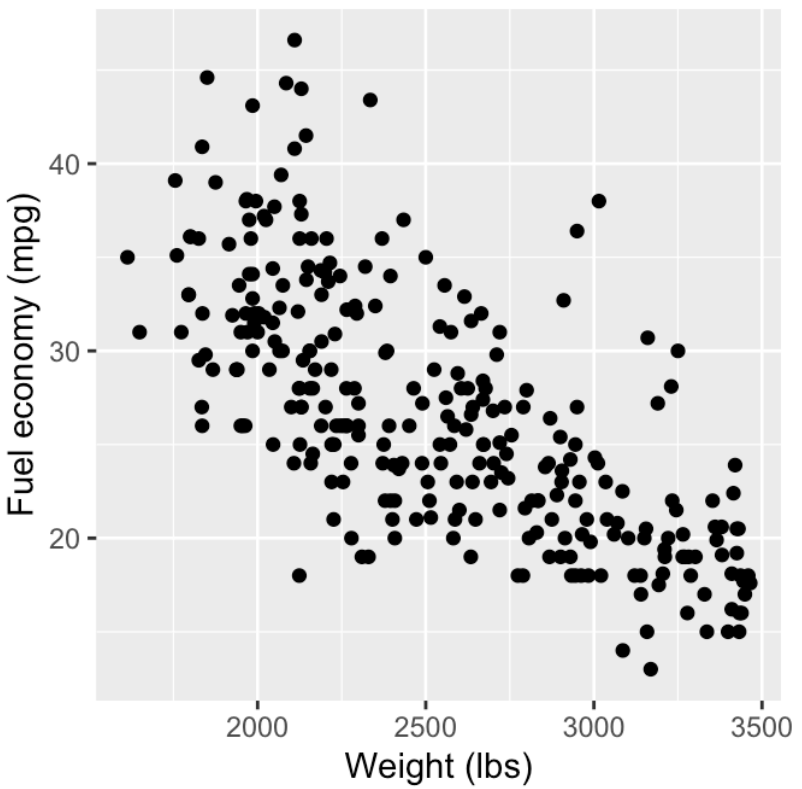
Form?

```
gf_point(mpg ~ weight, data = mpg,
         xlab = "Weight (lbs)",
         ylab = "Fuel economy (mpg)")
```

Direction?

Strength?

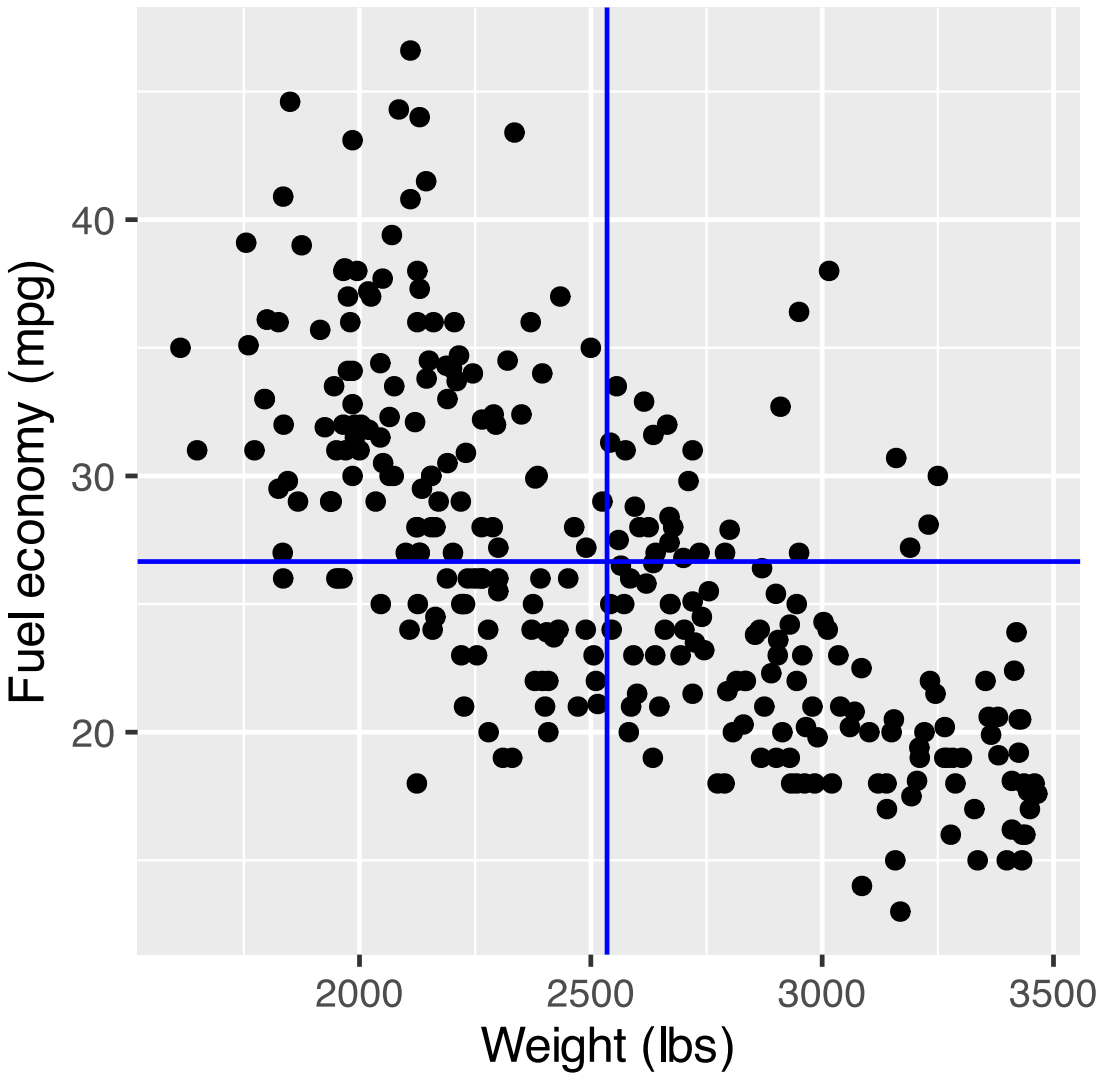
Outlier/unusual features?



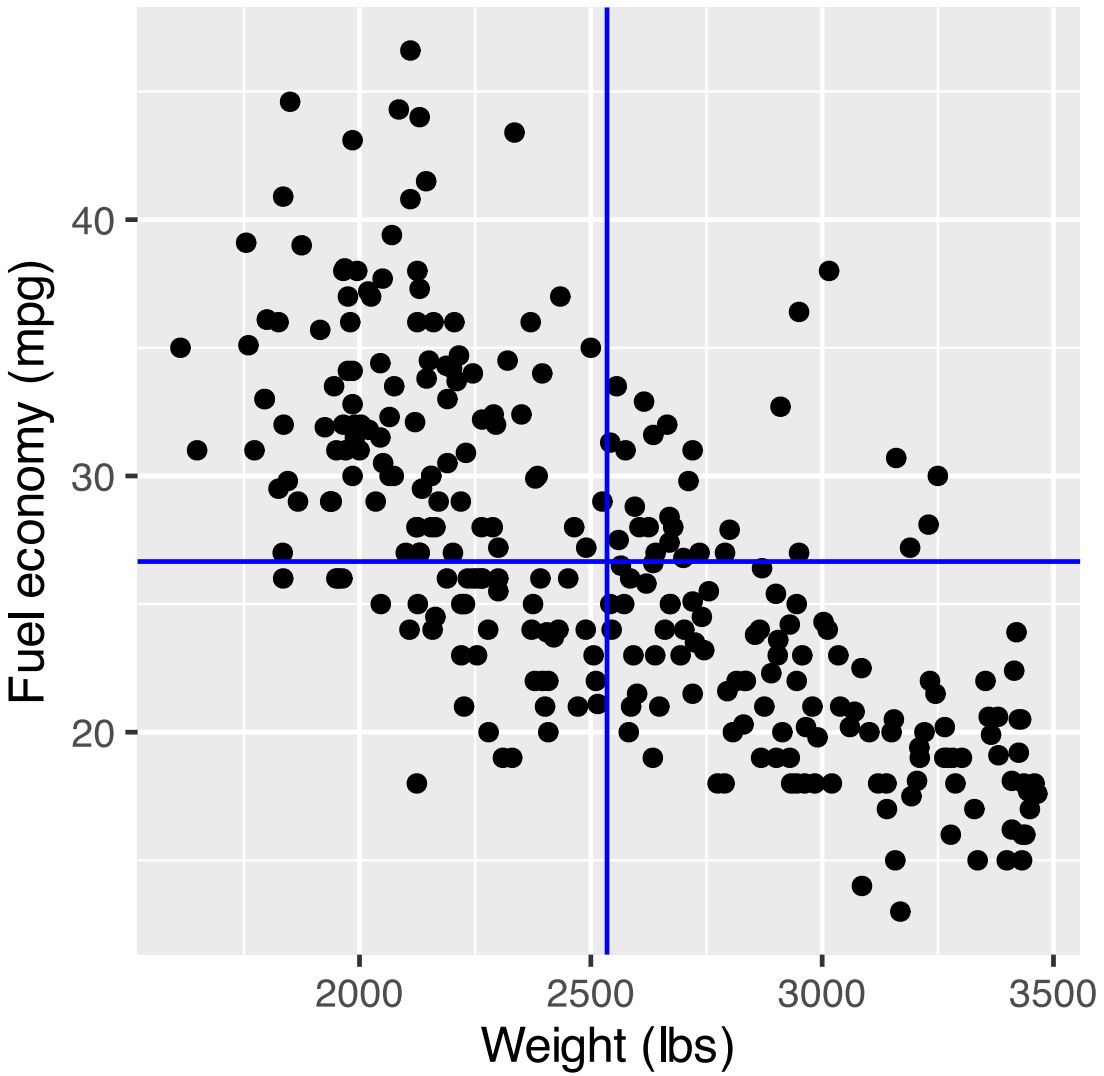
Strength of association

Perception of the strength of the association can change just by changing the scale on the vertical axis

Covariance



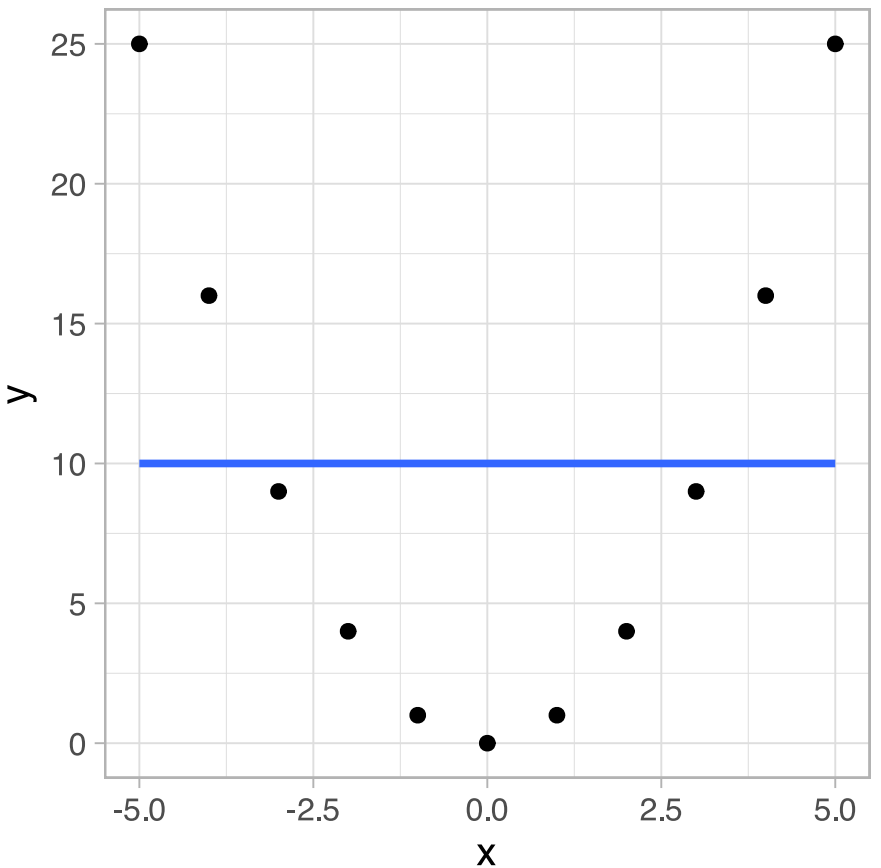
Correlation



Caution

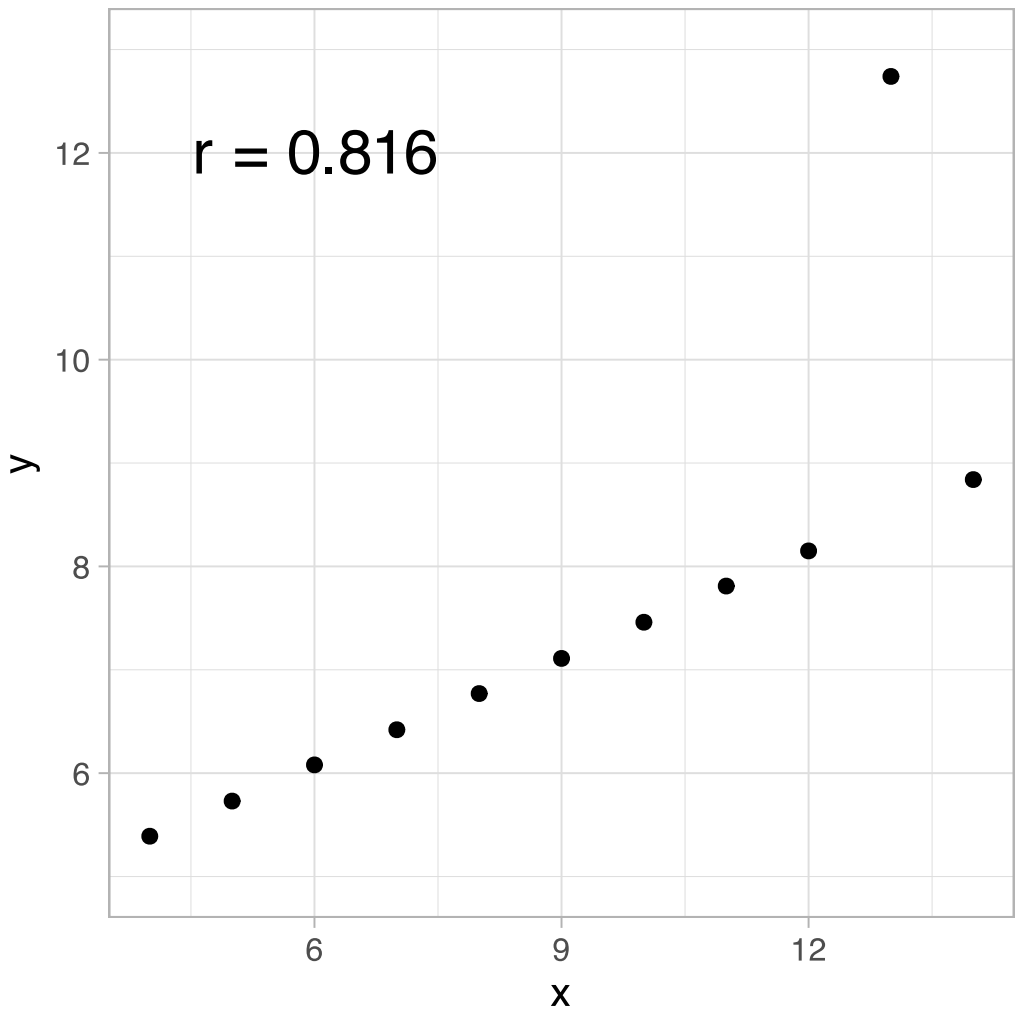
Correlation only measures the strength/direction of **linear relationships**

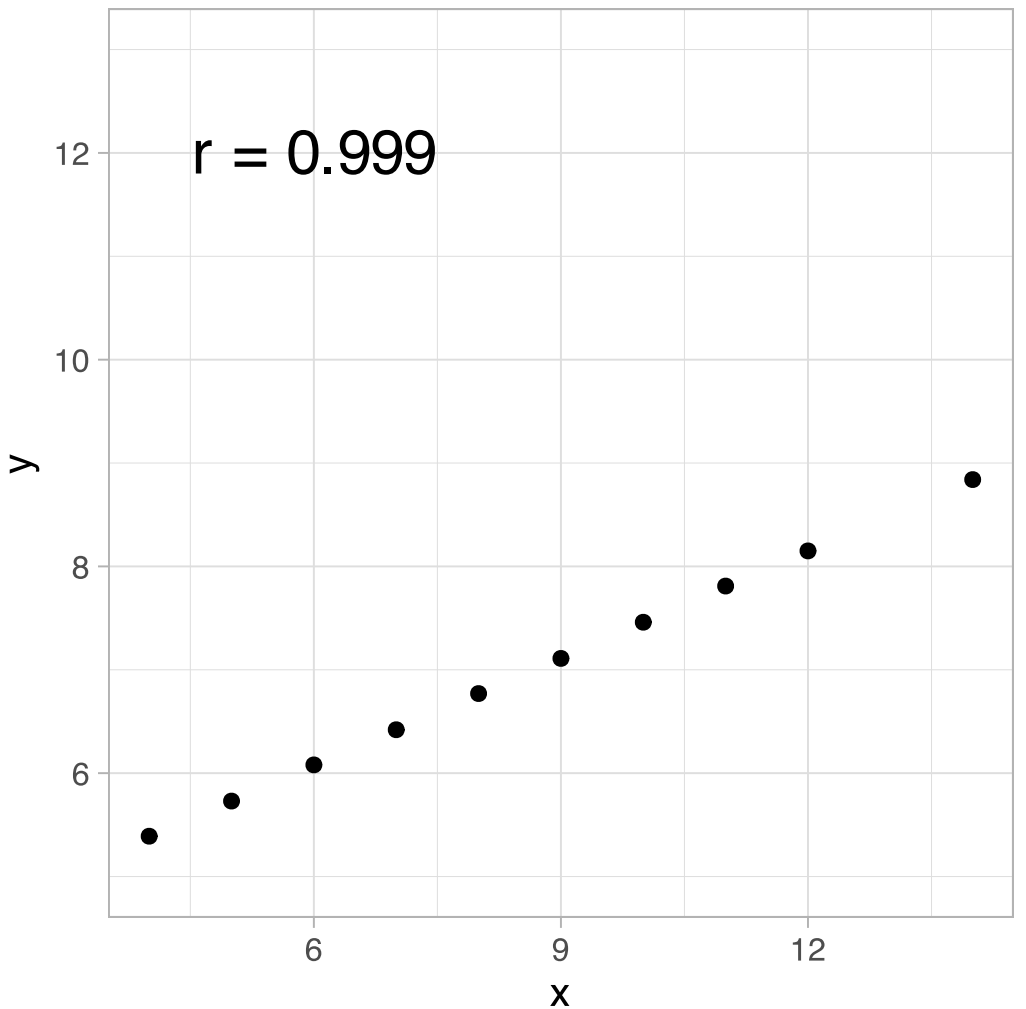
```
x <- -5:5
y <- x^2
cor(x, y)
## [1] 0
```



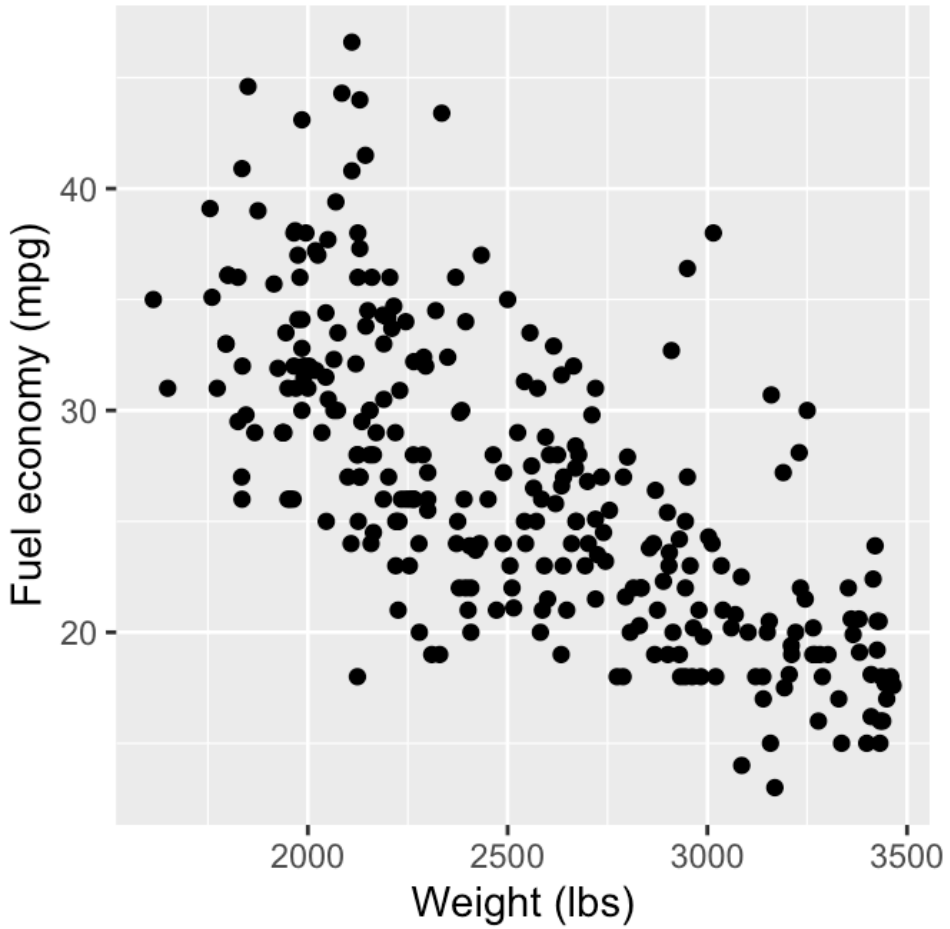
Caution

Correlation is **sensitive to outliers**





Model selection



$$y_i = f(x_i) + \varepsilon_i$$

What is a good model for the fuel economy data?

Simple Linear Regression

Predicting fuel economy

Task: predict the fuel economy of a vehicle based on its weight—i.e. find $\hat{\alpha}$ and $\hat{\beta}$

$$\hat{y}_i = \hat{\alpha} + \hat{\beta} x_i$$

Approach: minimize the residual sums of squares

$$\text{RSS} = \sum_{i=1}^{n} (y_i - a - b x_i)^2$$

- This is called least squares (LS) estimation

Linear models in R

`lm` is our workhorse function

```
mod <- lm(mpg ~ weight, data = mpg)
```

- The formula is of the form `response ~ predictor`
- The result is an object of class `lm`

```
names(mod)
##  [1] "coefficients"  "residuals"     "effects"
##  [5] "fitted.values" "assign"        "qr"
##  [9] "xlevels"       "call"          "terms"
```

Linear models in R

You have a few options to the results

1. **Print**: print `mod` to see the estimated regression coefficients
2. **Summary**: `summary(mod)` displays the most useful information about the model
3. **Attributes**: extract the attribute of interest using the `$` operator

summary()

```
summary(mod)
```

```
Call:
lm(formula = mpg ~ weight, data = mpg)

Residuals:
     Min       1Q   Median       3Q      Max 
-12.7011  -3.3404  -0.5987   2.3588  16.0605 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 51.5871689  1.4835394   34.77   <2e-16 ***
weight      -0.0098334  0.0005749  -17.11   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretations

$\widehat{\beta}$

$\widehat{\alpha}$

Making predictions

Once we have our estimated regression coefficients, $\hat{\alpha}$ and $\hat{\beta}$, obtaining a prediction is easy.

Example predict the MPG for a car weighing 2,500 lbs

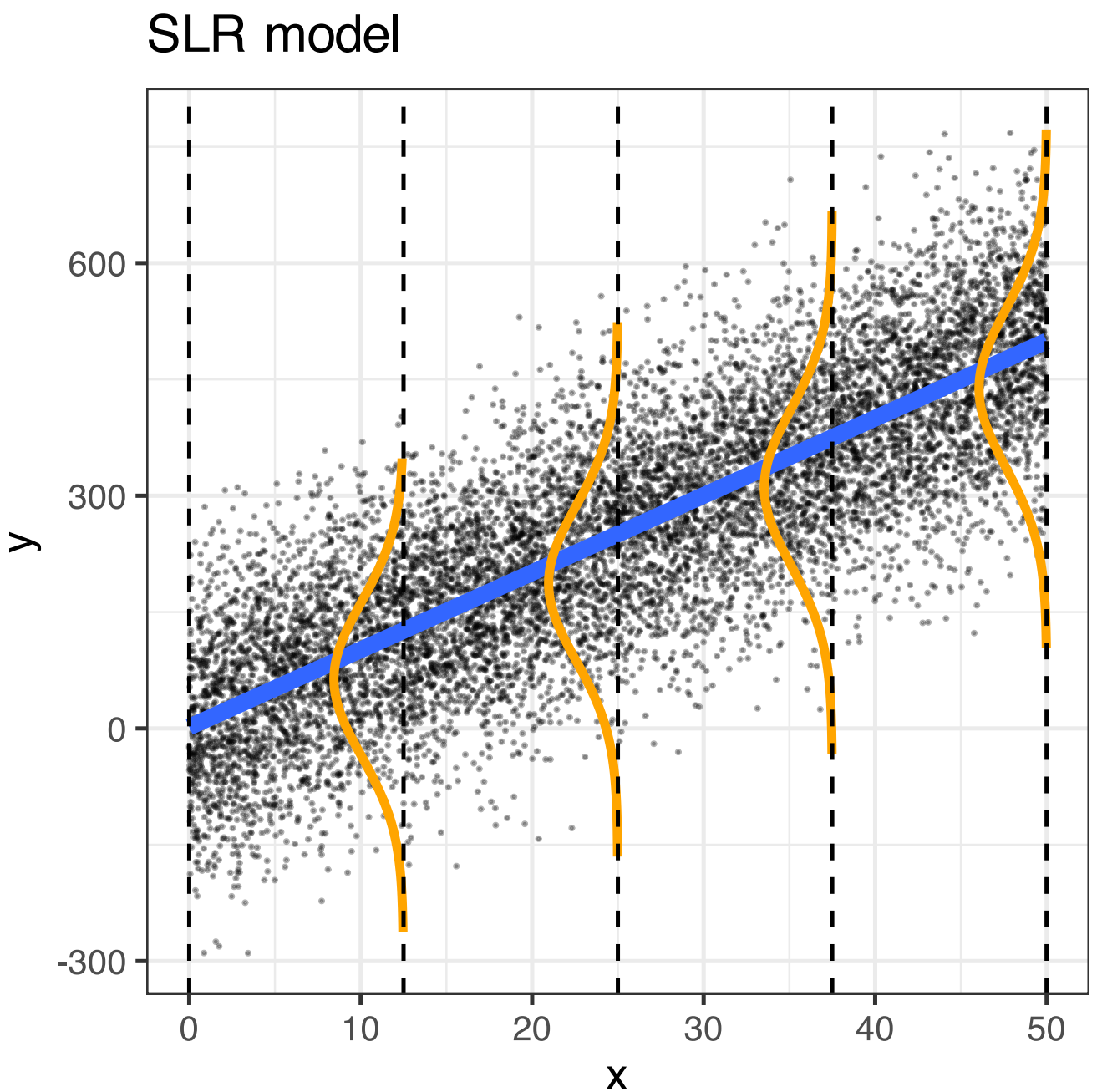
$$\hat{y} = \hat{\alpha} + \hat{\beta}(2500)$$

In R, we use the `predict` function

```
predict(mod, newdata = data.frame(weight = 2500))
##         1
## 27.00371
```

The full SLR model

- LS only assumes that there is a linear relationship between x and y
- Additional assumptions are needed to understand the uncertainty of our predictions
- The SLR model can be written in a few forms
  - $Y_i = \alpha + \beta x_i + \varepsilon_i$ where $\varepsilon \overset{iid}{\sim} N(0, \sigma^2)$
  - $Y_i \overset{iid}{\sim} N(\alpha + \beta x_i, \sigma^2)$


SLR model
600
300
0
-300
y
0
10
20
30
40
50
x

Regression conditions/assumptions

1. **Linearity**: $E(Y_i|X = x_i) = \mu_i = \alpha + \beta x_i$
2. **Independence**: $\varepsilon_1, \ldots, \varepsilon_n$ are independent
3. **Constant error variance**: $Var(\varepsilon_1) = \cdots = Var(\varepsilon_n) = \sigma^2$
4. **Normal error terms**: $\varepsilon_i \sim N(0, \sigma^2)$

ML estimation

We cannot obtain an estimate of σ^2 through LS, so instead we can use maximum likelihood (ML)

To do this, we simply maximize the likelihood function

$$L(\alpha, \beta, \sigma) = \prod_{i=1}^{n} f(Y_i | x_i, \alpha, \beta, \sigma) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-(Y_i - \alpha - \beta x_i)/2\sigma^2}$$

Idea: finding the values of α, β, and σ that make our data most likely

ML estimation

It's easier to work with the log likelihood

$$\ell(\alpha, \beta, \sigma) = \sum_{i=1}^{n} \log(\sigma) - \frac{1}{2}\log(2\pi) - (Y_i - \alpha - \beta x_i)^2/2\sigma^2$$

$$\frac{\partial \ell}{\partial \alpha} = \frac{1}{\sigma^2} \sum_{i=1}^{n} (Y_i - \alpha - \beta x_i) \qquad \frac{\partial \ell}{\partial \beta} = \frac{1}{\sigma^2} \sum_{i=1}^{n} (Y_i - \alpha - \beta x_i) x_i$$

[Math Processing Error]

where $e_i = Y_i - (\alpha + \beta x_i) = Y_i - \widehat{Y}_i$

ML estimation

Setting the derivatives to 0 and solving yields

$$\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$

$$\hat{\beta} = \frac{\sum(x_i - \bar{x})(Y_i - \bar{Y})}{\sum(x_i - \bar{x})^2}$$

$$\hat{\sigma}^2 = \frac{\sum e_i^2}{n}, \text{ where } e_i = y_i - (\hat{\alpha} + \hat{\beta}x_i)$$

$\hat{\sigma}^2$ is biased, so we must make an adjustment to obtain an unbiased estimator

$$S^2 = \frac{\sum e_i^2}{n - 2}$$

Properties of our estimators

- $\hat{\alpha}$ and $\hat{\beta}$ are **unbiased estimates** of α and β
- $\hat{\alpha}$ and $\hat{\beta}$ are the **best linear unbiased estimates** (BLUE); that is, they have the smallest variance of all linear unbiased estimates
- S^2 is an unbiased estimate of σ^2