Linear Models

DATA 299

Based on Chapters 5 and 6 in *Analytic Method in Sports, 2nd ed.*, Severini

Hitchman

Spring '25

Setting the Scene for Linear Regression

We assume that there are two components that contribute to a response variable Y:

1. A function that relates the expected (or average) value of Y to **explanatory variables** $X_1, X_2, \ldots, X_p$. That is,

$$E(Y) = f(X_1, X_2, \ldots X_p).$$

   In linear regression, we assume this function is linear:

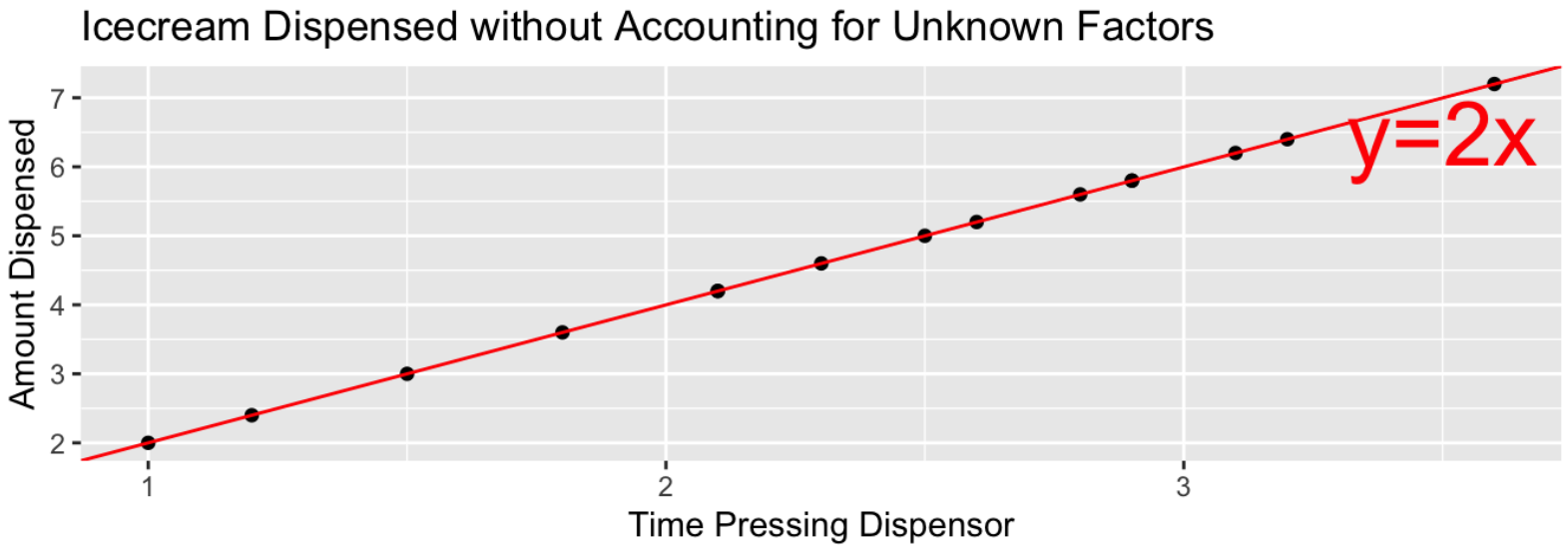
$$E(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p.$$

2. Random, unexplained (by the model) variability ϵ that results in an individual response Y_i differing from the one expected from a set of input values.

The first component is called the **signal** and the second component the **noise**.

Ice Cream Example

Suppose an ice cream machine is made to dispense 2 oz. of ice cream per second, on average. If each person using the machine got exactly 2 oz. per second, the relationship between time and amount dispensed would look like this:



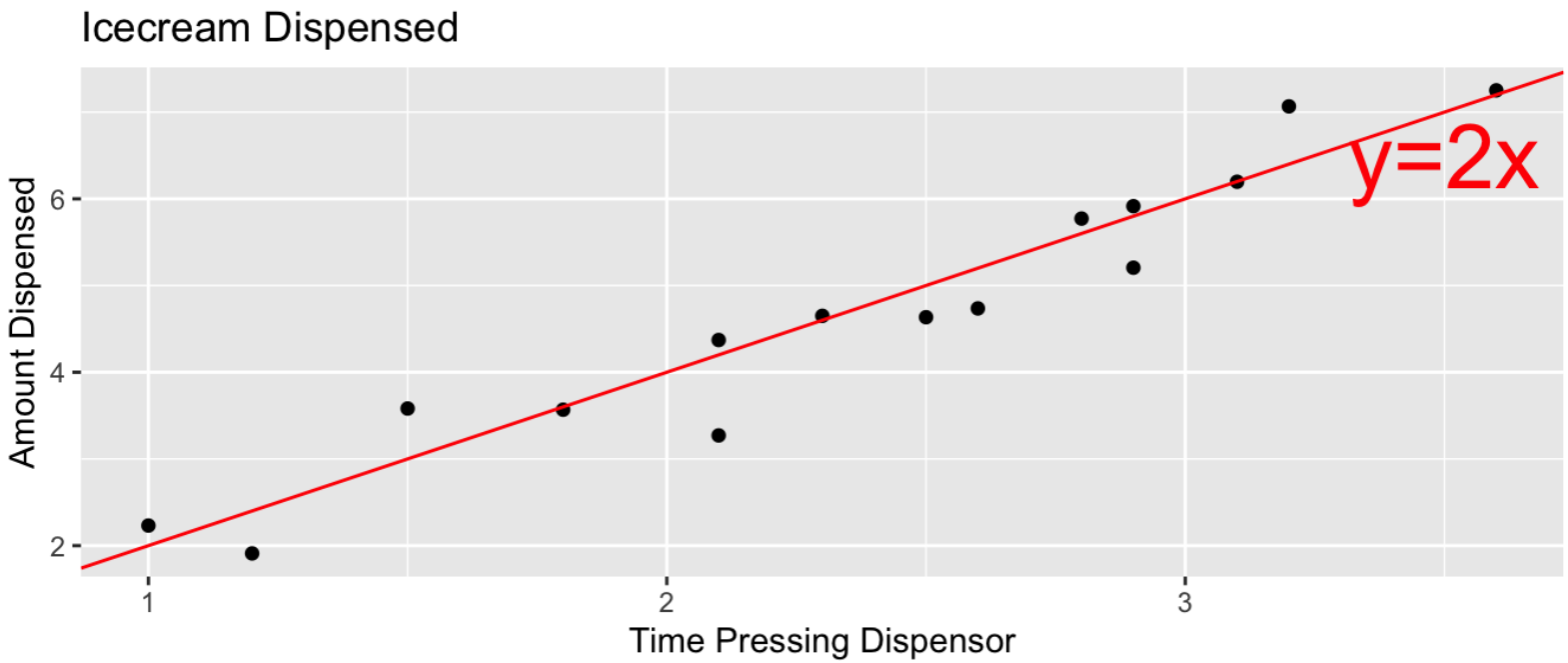
https://bookdown.org/ajsage/statistics_for_data_science_notes/the-normal-error-linear-regression-model.html

Ice Cream Example

In reality, however, the actual amount dispensed each time it is used will vary due to unknown factors like:

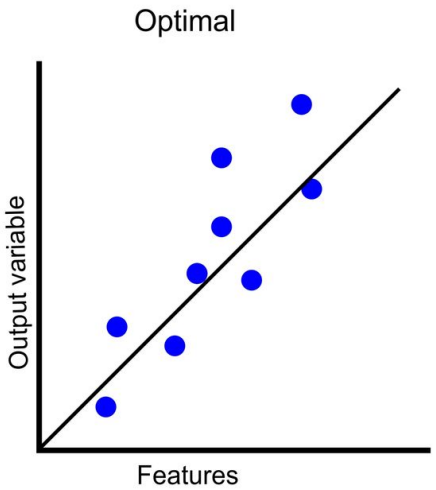
- force applied to dispenser
- temperature of ice cream
- other unknown factors

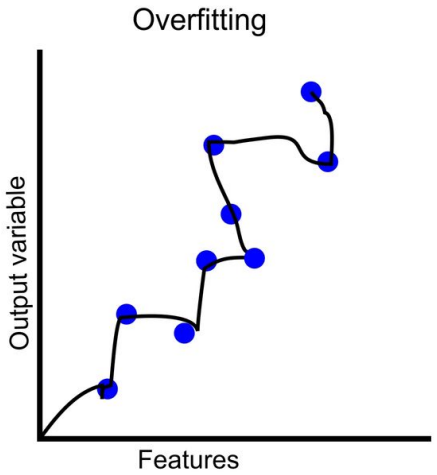
So, the data will probably look more like this:



Beware Overfitting

We build two models $E(Y) = f(X)$ based on data $(x_1, y_1), \ldots (x_9, y_9)$.
Which is a better model?





Modeling the Noise

It is common in statistical theory to assume that the noise ϵ is normally distributed with mean 0 and some positive standard deviation σ:

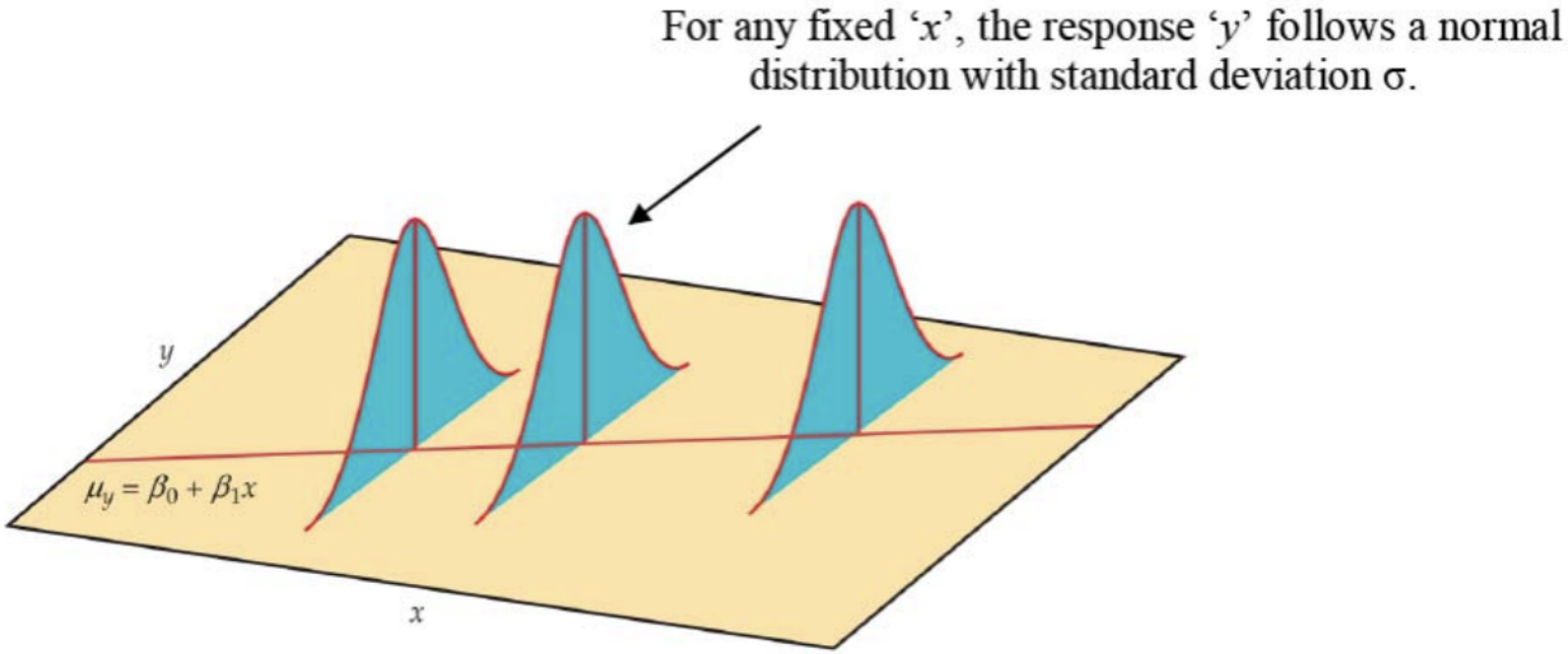


image found at: https://faculty.ksu.edu.sa/sites/default/files/stat_332-393_1_0.pdf

Summary of assumptions

- We believe that explanatory variables (the X_i) tell us what value we should expect some response variable Y to take.
- We believe we have a linear relationship, with some 'noise' due to unexplained factors:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k + \epsilon$$

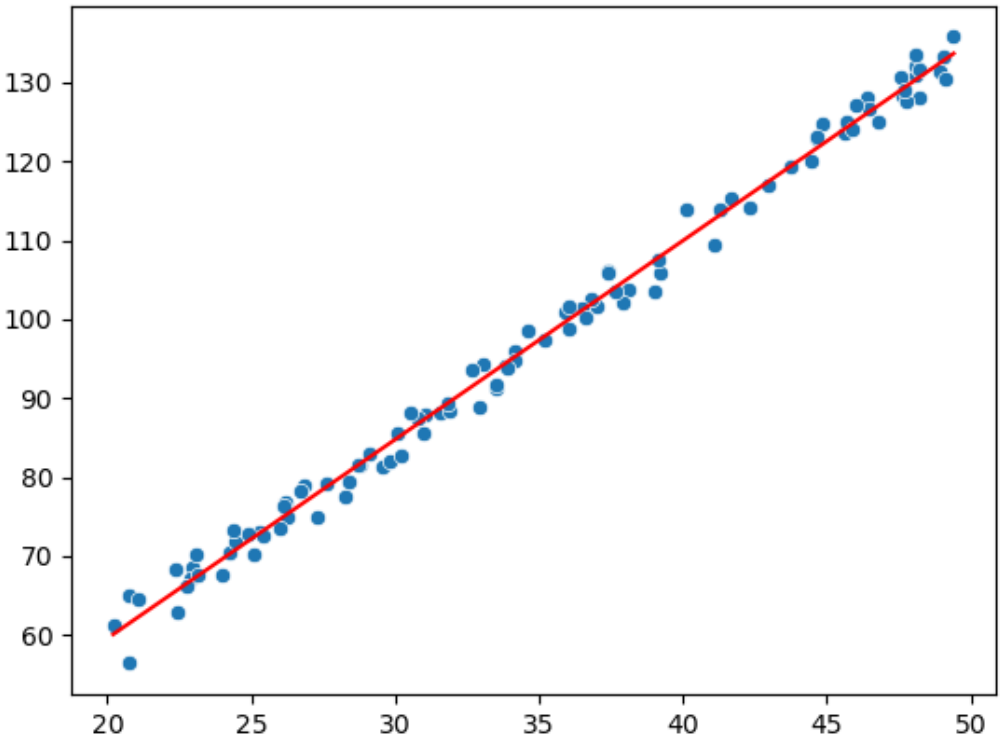
- We believe the 'noise' ϵ to be normally distributed with mean 0 and some variance that is independent of the X_i values.

Simple Linear Regression

We have a single explanatory variable X, so our model looks like

$$Y = \beta_0 + \beta_1 X + \epsilon.$$

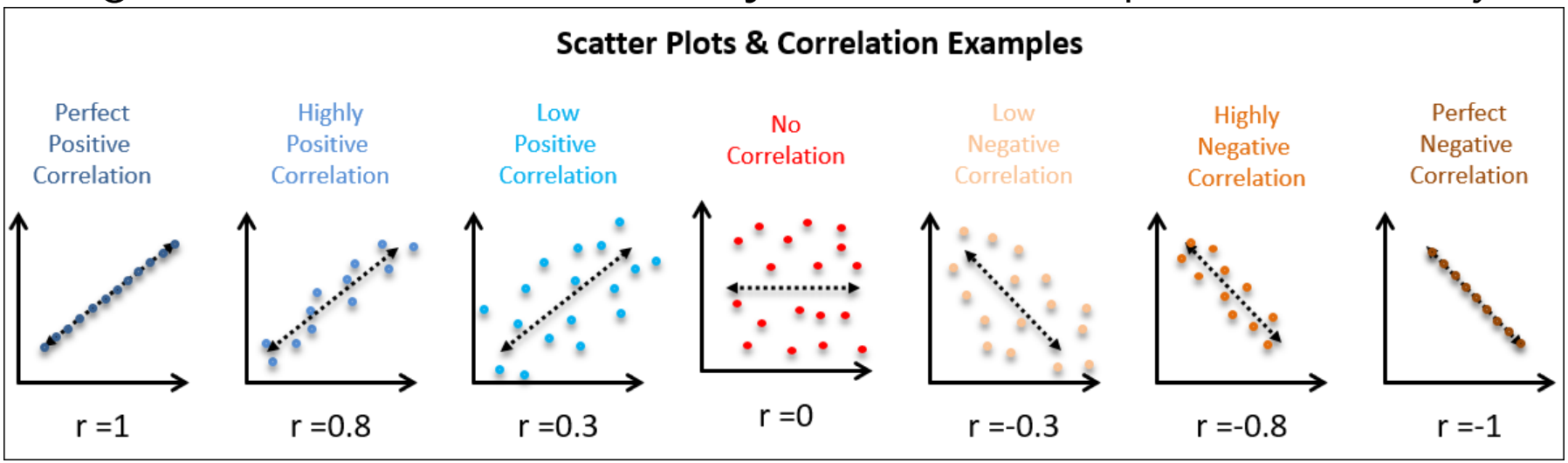
In this case we have a simple visual of the relationship: a line in the xy-plane with slope β_1 and y-intercept β_0.



Our goal: find the 'best' line to fit the data.

Correlation coefficient

Do x and y have a linear relationship?
The **correlation coefficient** is a number (r) between -1 and 1 that measures the strength and direction of the **linearity** of the relationship between x and y.



https://cqeacademy.com/wp-content/uploads/2018/06/Scatter-Plots-and-Correlation-Examples.png

- If the direction is negative, $r < 0$
- if the direction is positive, $r > 0$
- the closer the points hug a single line, the closer r gets to ± 1.
- If there is really no linear form of any kind, $r \approx 0$.

Guess the correlation

Click below if you want practice guessing the correlation!
Guess the correlation Link

Facts about r

- r does not depend on units.
- It only measures the strength of a linear relationship.
- r is strongly affected by outliers, because it is calculated using means and standard deviations of the variables involved
- r is the same whether we regress x on y or y on x.

Formula for r

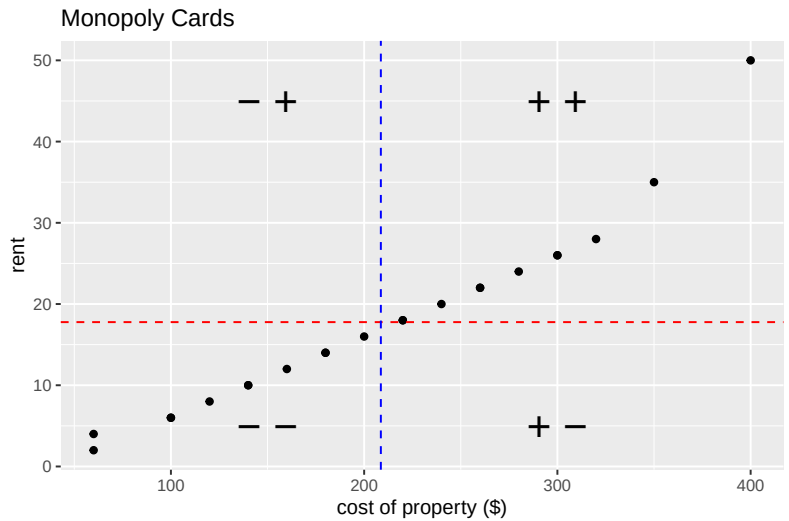
$$r = \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{x_i - \overline{x}}{s_x} \right) \left(\frac{y_i - \overline{y}}{s_y} \right),$$

where

- $\overline{x}$ - the mean of the x_i
- s_x - the standard deviation of the x_i
- $\overline{y}$ - the mean of the y_i
- s_y - the standard deviation of the y_i

Making sense of the formula

We can mark our plot into quarters by using a vertical line through the value of $\overline{x}$ on the x-axis, and a horizontal line through the value of $\overline{y}$ on the y-axis.
Quadrant signs $\leftrightarrow$ signs of $(x_i - \overline{x})/s_x$ term and $(y_i - \overline{y})/s_y$ for a point in that quadrant.
The quadrants marked $++$ and $--$ will contribute positively to the sum for r; the other quadrants contribute negatively to the sum.



Least-Squares Regression Line

We assume the association between X and Y is modelled by

$$Y = \beta_0 + \beta_1 X + \epsilon.$$

From observed data

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$

we may approximate the model parameters β_0 and β_1.

- The **Least-Squares Regression Line** is the line that minimizes the sum of the squares of the vertical distances between the data points and the line.

Picturing the Least-Squares Regression Line

A nice demonstration on Desmos:
https://www.geogebra.org/m/XUkhCJRj

Determining the Least-squares regression line

Given n points of the form (x_i, y_i) we need to know:

- $\overline{x}$ - the mean of the x_i
- s_x - the standard deviation of the x_i
- $\overline{y}$ - the mean of the y_i
- s_y - the standard deviation of the y_i
- r - the correlation coefficient of the scatter plot

The equation of the least-squares regression line has the form

$$y = \hat{\beta}_0 + \hat{\beta}_1 x$$

where the slope is

$$\hat{\beta}_1 = r\frac{s_y}{s_x}$$

and the y-intercept is

$$\hat{\beta}_0 = \overline{y} + \hat{\beta}_1 \overline{x}.$$

The values $\hat{\beta}_i$ are called the regression coefficients and they serve as estimators for the theoretical model parameters β_0 and β_1.

Summary: Least-squares Regression

- We have data $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, and we suspect x associates with y in a linear way.
- We build the least-squares regression line from these data.
- This serves as a model for predicting the value of y from a given x. Let $\hat{y}$ denote a predicted y:
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$
- The residual of a data point (x_i, y_i) is the quantity $y - \hat{y}_i$ (**observed** y associated with the input x_i **minus predicted** $\hat{y}_i$ for this input).
For instance,

Black box approach

We can use software (R is great with regression, Excel, python, TI-83 :)) to generate the least-squares regression line.
In R, we can find the slope and y-intercept of the least squares regression line for two vectors y (response) and x (explanatory) by running

$$\text{lm}(y \sim x)$$

For instance, you could run the above code after defining two small vectors (of equal length):

```
x <- c(0,1,2,3)
y <- c(1,1,3,4)
```

This produces the least-squares line $\hat{y} = 1.1x + 0.6$.
And a visual of the data and line, in base R:

```
plot(x,y,pch=16)
abline(a=0.6,b=1.1) #a = intercept, b = slope!
```

Residuals in this simple model

The LSR line built from

```
x <- c(0,1,2,3)
y <- c(1,1,3,4)
```

is $\hat{y} = 1.1x + 0.6$.
The residual for the data point (2,3) is the observed y minus the predicted y-value from the line:

$$\text{residual for (2,3) is } 3 - (1.1 \cdot 2 + 0.6) = 0.2,$$

which means the point $(2, 3)$ lives above the least-squares line.
The point $(1, 1)$ has residual $1 - (1.1 \cdot 1 + 0.6) = -0.7$. A negative residual means the point lives below the least-squares line.

Residual Plots

It can be useful to make a point plot of the observed inputs x_i against their corresponding residuals $(\hat{y}_i - y_i)$. Such a plot is called a **residual plot** and helps us assess whether the assumptions of the linear model are reasonable.
A few things to keep in mind.

- If the residual plot has a clear pattern, a line might not be the best model
- If the residual plot shows a changing spread of values as x changes, then the assumption that the random error ϵ has constant variance for all x.

Least Squares Regression on two data frame columns

We often build a linear model from data frame columns. For instance in the tidyverse **starwars** data frame, we might consider

$$\text{lm}(\text{mass} \sim \text{height}, \text{data} = \text{starwars})$$

if we want to predict mass (y, response) from height (x, explanatory variable). Run the code above to find least-squares regression line is

$$\hat{y} = 0.624x - 11.487.$$

Better yet, we might store all sorts of useful information about the linear model by creating a model object:

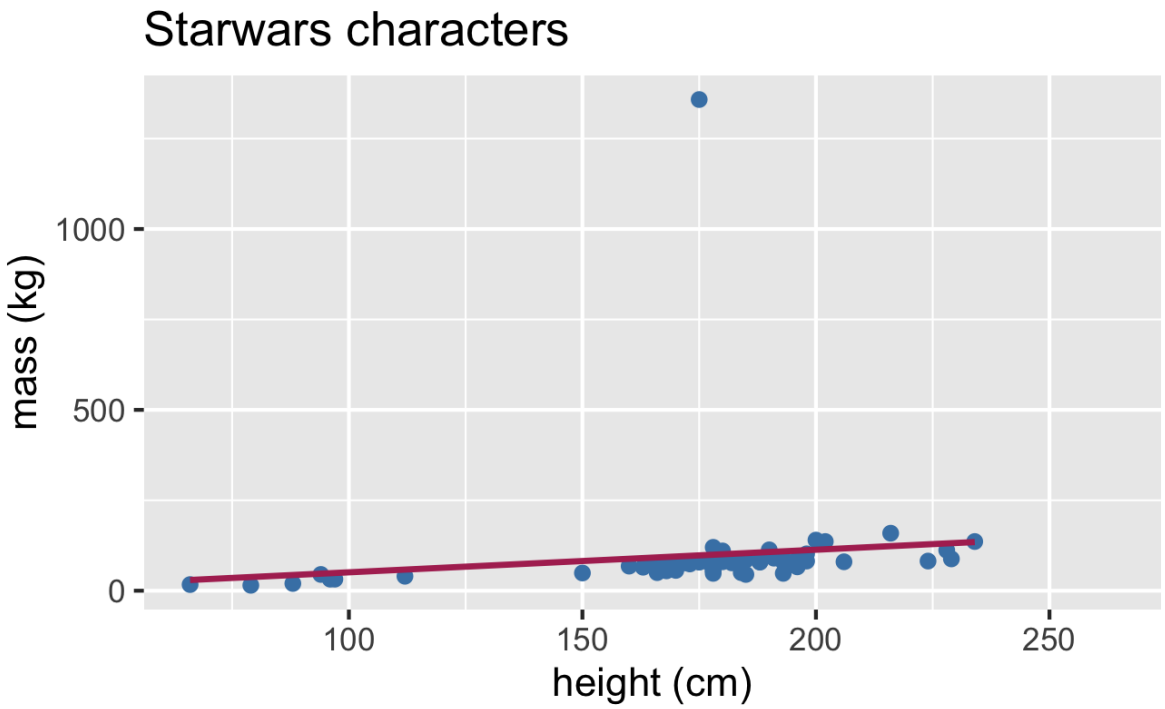
```
model <- lm(mass ~ height, data = starwars)
```

Then we can retrieve the slope and y-intercept of the least squares line by running

```
model$coefficients
```

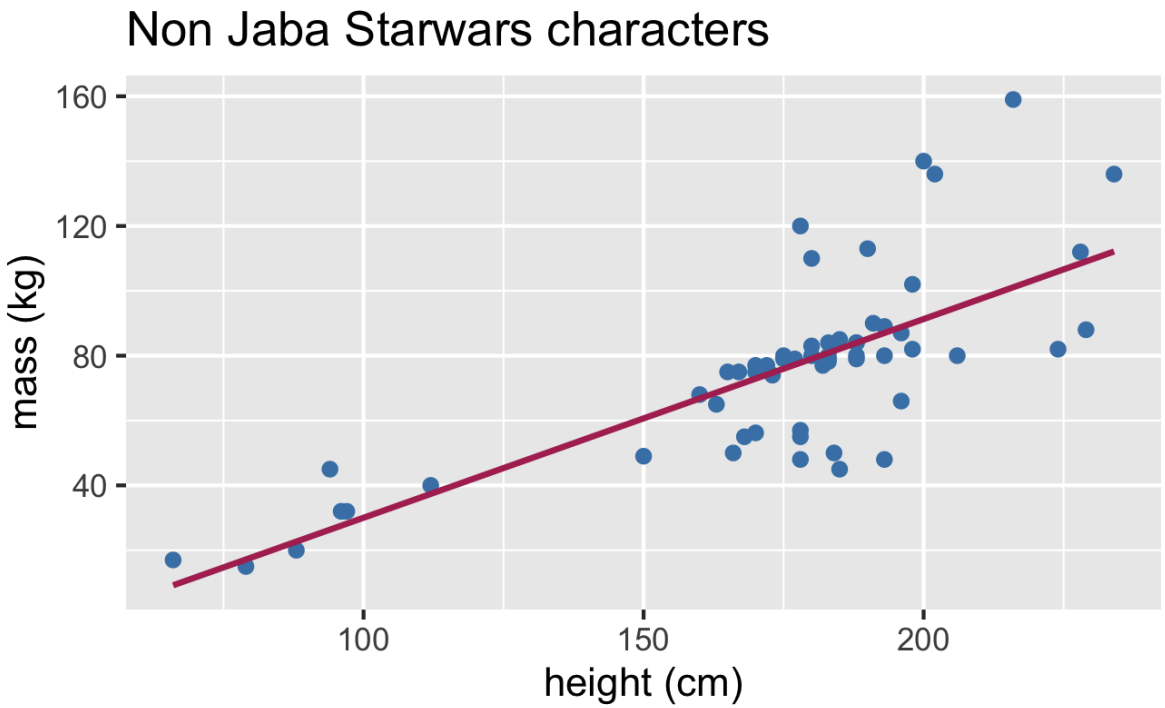
Starwars height vs mass. Where's Jaba?

```
ggplot(starwars, aes(x = height, y = mass))+
  geom_point(col = "steelblue") +
  geom_smooth(method='lm', formula=y~x, col="maroon",linewidth = 0.8
  labs(x = "height (cm)",
       y = "mass (kg)",
       title = "Starwars characters")
```



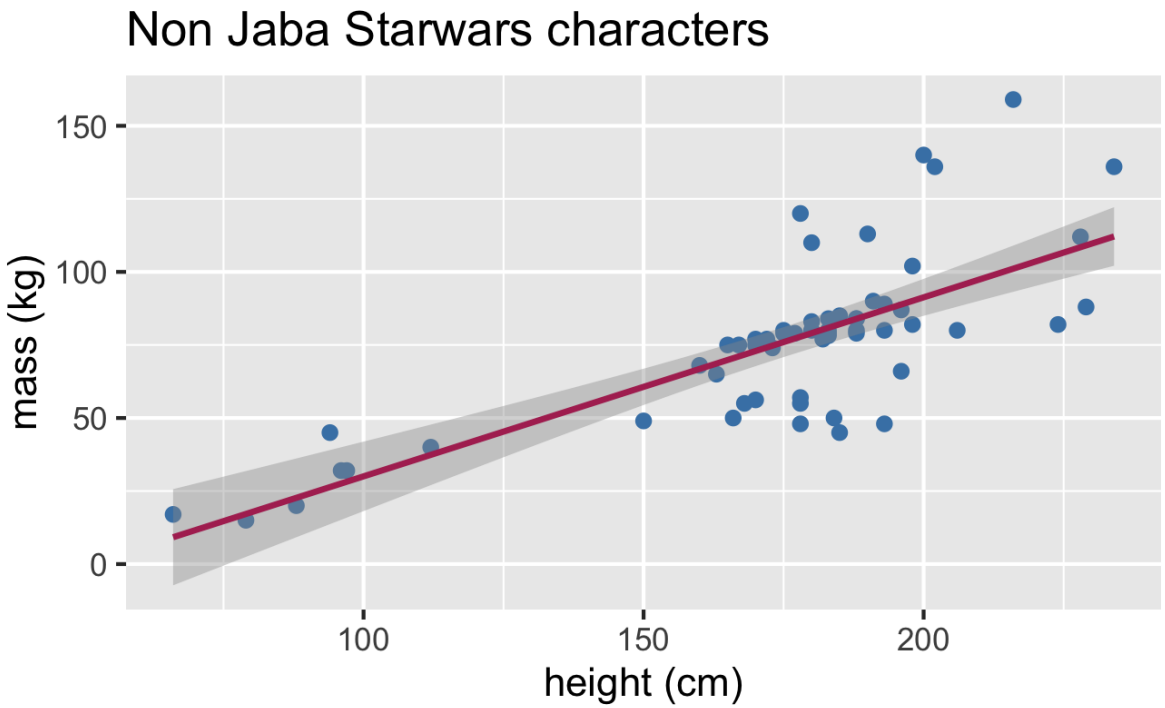
Starwars height vs mass without Jaba.

```
ggplot(starwars |> filter(mass < 1000), aes(x = height, y = mass))+
  geom_point(col = "steelblue") +
  geom_smooth(method='lm', formula=y~x, col="maroon",linewidth = 0.8
  labs(x = "height (cm)",
       y = "mass (kg)",
       title = "Non Jaba Starwars characters")
```



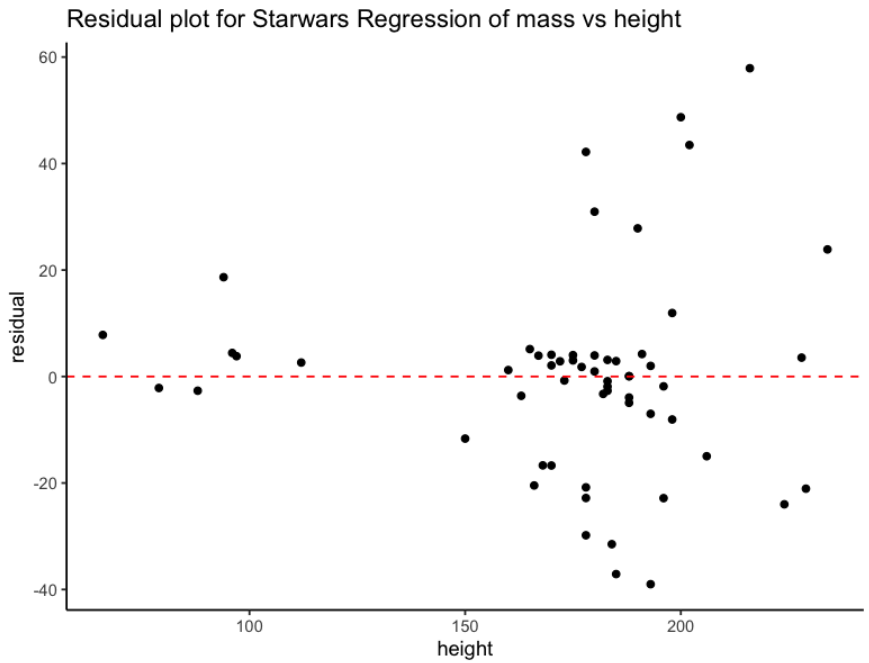
Same plot with confidence bands

Set `se = TRUE` in the `geom_smooth()` command (or omit it since TRUE is its default value) generates confidence bands around the least-squares line. Generally, the further x gets from the mean of the x_i, the less reliable the line becomes as a model for estimating y.



Residual Plot for the non-Jaba model

Looks like more variation about the line at the high end of the height spectrum, constant variance assumption may not be reasonable in our model.



Evaluating the Linear Model

We have three standard metrics for evaluating a linear model.

(1) r^2 measures the proportion of the variation in y that is explained by its linear fit with x. r^2 is always between 0 and 1. The closer to 1, the better the linear model is at predicting y from x.

From the object `model` from the starwars example, we can find r^2:

```
summary(model)
```

Evaluating the Linear Model

(2) *p-values* - Each parameter estimate $\hat{\beta}_i$ has a p-value associated with it. This p-value captures the probability of observing the data that we observed under the assumption that the value of β_i is actually 0. This means that if we have a low p-value, it is highly likely that the value of β_i is nonzero, and thus 'significant'. p-values can also be found via `summary(model)`.

Evaluating the Linear Model

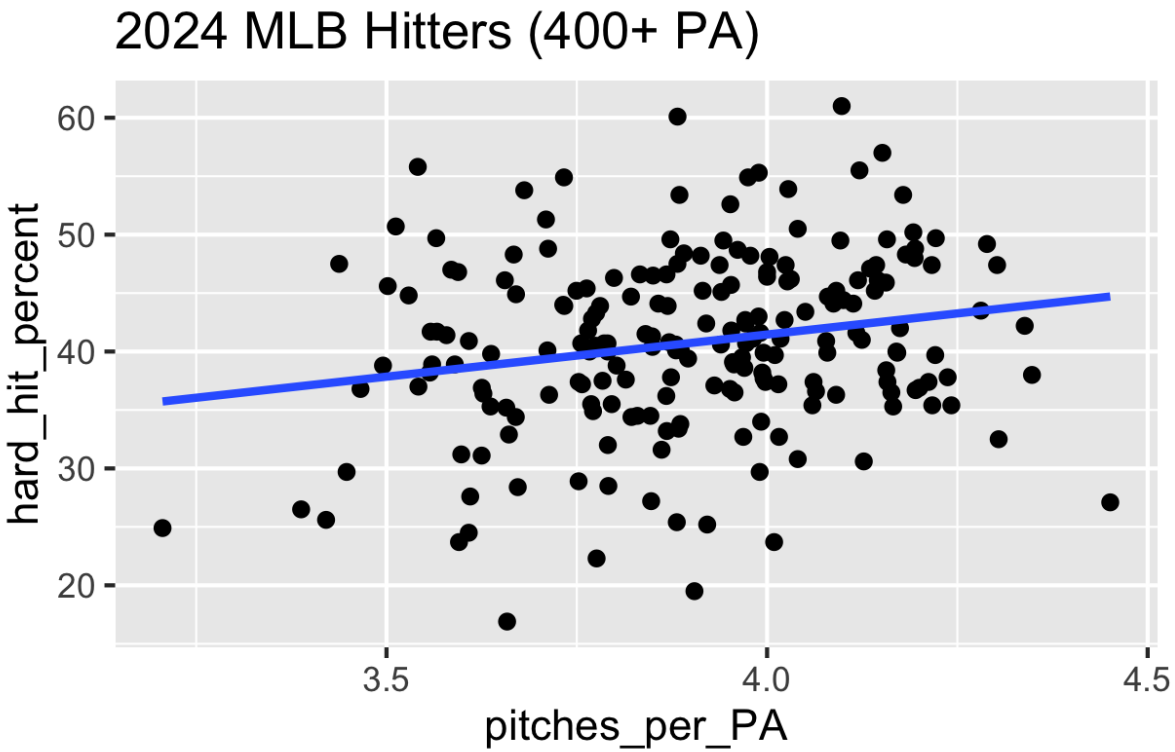
(3) *Cross-validation*. Randomly divide the data into a training set (80%) and a testing set (20%). Fit the model on the training set, then look at the *mean squared error* on the test set and compare it to that on the training set. Make this comparison across different sample sizes as well. If the mean squared errors are approximately the same, then the model generalizes well, and we're not in danger of overfitting.

The *mean squared error* is a number that measures how much the predicted values $\hat{y}_i$ vary from the observed values y_i:

$$\text{MSE} = \frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$$

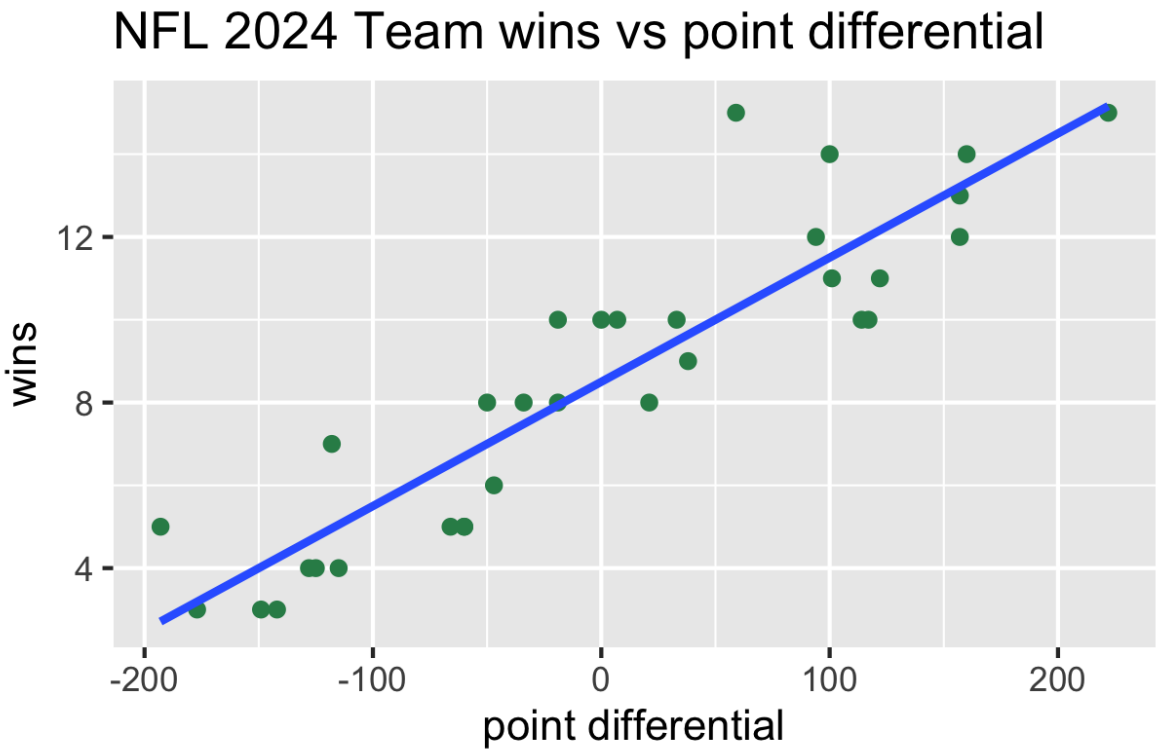
Example 1: Are batters more successful if they see more pitches?

We look at the code in R. Here's a plot:



Example 2: Does point differential predict wins in Football?

We look at the code in R. Here's a plot:



Multiple Linear Regression

Perhaps we believe Y is associated with a linear combination of several other variables:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \epsilon.$$

See the code associated with these slides on our course resource page for examples

Example 5: MLB Career Trajectory

This discussion based on Chapter 7 of ABDR3e.
We can fit a quadratic model for a player's home run hitting ability (Y) as a function of their age (X):
$Y = \beta_0 + \beta_1(X - 30) + \beta_2(X - 30)^2 + \epsilon$
We work in R, but share this plot of Ken Griffey Jr.'s home run hitting prowess over the course of his career.