LECTURE 13

Simple Linear Regression

Introducing (re-visiting) a more nuanced model.

Data 100/Data 200, Fall 2021 @ UC Berkeley
Fernando Pérez and Alvin Wan
Content credit: Suraj Rampure, Ani Adhikari

Recap: Modeling

Recap of the modeling process

- **Choose a model.**
  - So far, we've seen the constant model $\hat{y} = \theta$.
    - θ is called a parameter.
- **Choose a loss function.**
  - So far, the options have been squared loss or absolute loss.
    - Squared loss: $L_2(y, \hat{y}) = (y - \hat{y})^2$.
    - Absolute loss: $L_1(y, \hat{y}) = |y - \hat{y}|$.
  - Loss functions tell us how much to penalize a single prediction.
- **Minimize average loss across our entire dataset, to determine the optimal parameters.**
  - Smaller average loss values mean a better fit; thus, we find the parameters that minimize average loss.

$$MSE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad MAE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

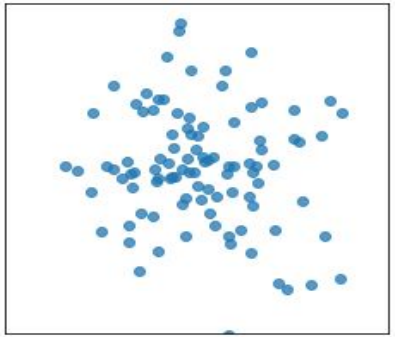
Correlation

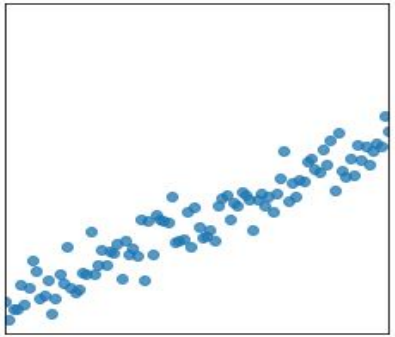
Exploring relationships between two variables

- The constant model we saw in the last lecture was only able to capture the distribution of a single variable. It was a **summary statistic**.
- More commonly, we create models that try to explain the relationships between **multiple variables** (which we will now denote with x and y).
- When using the Tips dataset as a motivator in the last lecture, we looked at a histogram of the data (with an overlaid KDE).
- When we have two continuous variables, we have several choices.
    - **Scatter plot.** These are the simplest choice.
    - Hexbin plot.
    - Contour plot.

Exploring relationships between two variables

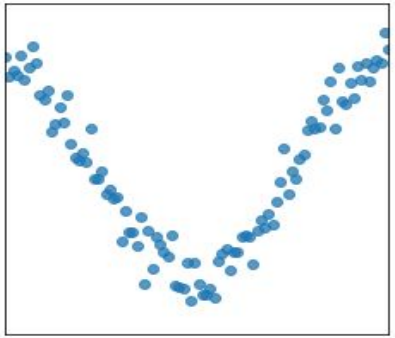
Looks like random noise.

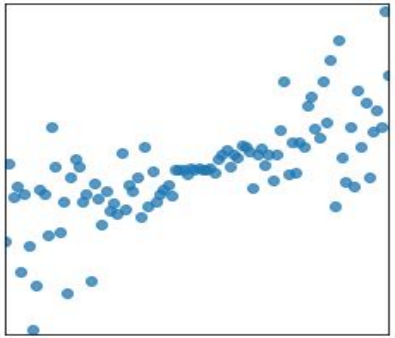




Looks like there's a strong linear relationship between x and y.

Looks like x and y are related, but not linearly.





Looks like there's somewhat of a linear relationship, but the points are more spread out away from the center.

Correlation coefficient

The Pearson's **correlation coefficient** (**r**) measures the **strength of the linear association** between two variables.

- It is a unitless quantity.
- It ranges between -1 and 1.
  - r = 1 indicates a perfect positive linear association (x and y lie exactly on a straight line that is sloped upwards).
  - r = -1 indicates a perfect negative linear association between x and y.
  - The closer r is to 0, the weaker the linear association between x and y is.
- It says nothing about **causation** or **non-linear association.**
  - Even if r = 1, it does not mean that x causes y! Correlation does not imply causation.
  - When **r = 0**, we say our two variables are **uncorrelated**. They could be related through some non-linear association, though.
- Very sensitive to outliers, as you will see in discussion.

Correlation coefficient

From Data 8: **r** is the **average** of the **product** of x and y, both measured in **standard units**.

Suppose our data looks like $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$. Then, $x_i \text{ in standard units } = \frac{x_i - \bar{x}}{\sigma_x}$.

We then have:

$$r = \underbrace{\frac{1}{n}\sum_{i=1}^{n}}_{\text{average}} \underbrace{\left(\frac{x_i - \bar{x}}{\sigma_x}\right)\left(\frac{y_i - \bar{y}}{\sigma_y}\right)}_{\text{product}}$$

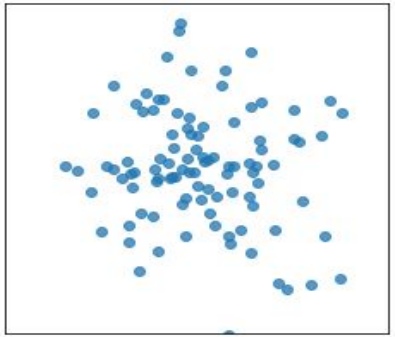
Note: Since σ_x and σ_y are constants, we can pull them out of the sum, and write

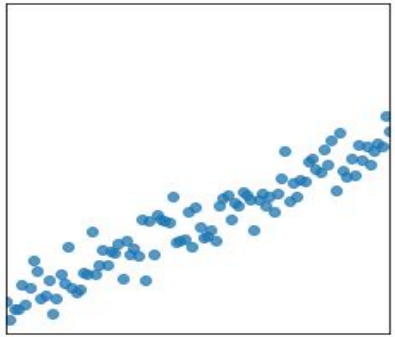
$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = r\sigma_x\sigma_y$$

This quantity is called the **covariance** of two variables.

Correlation coefficient

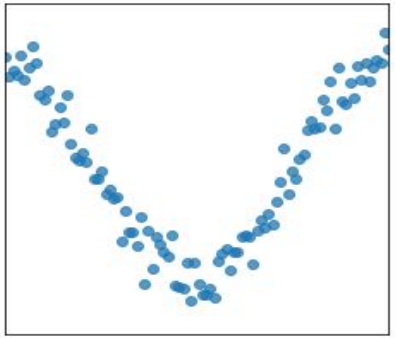
r = -0.121

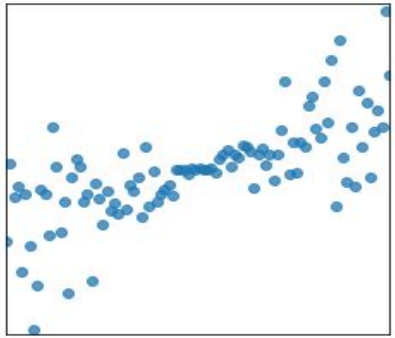




r = 0.949

r = 0.052





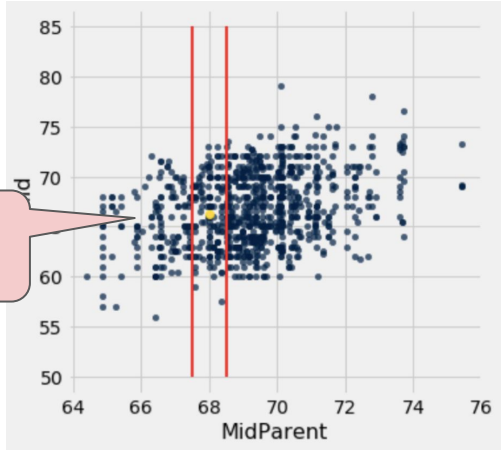
r = 0.704

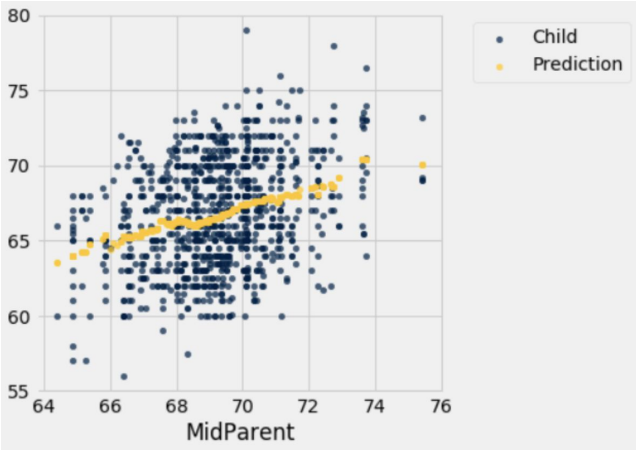
Simple linear regression

Graph of averages

Suppose we want to now **predict** the value of y, for any given x. One reasonable thing to do might be to compute the **average value of y for each x**, and predict that.

Bucket the x-axis into bins.





Doing this yields predictions that look like a line. So, let's model this relationship with a line!

Equation of the regression line

A simple linear model (with a slope and intercept) is of the form

$$\hat{y} = \theta_0 + \theta_1 x$$

Note, we have two parameters now. For simplicity's sake, we will instead say (for now):

$$\hat{y} = a + bx$$

We call this the **simple linear regression** model.

To determine the optimal model parameters $\hat{a}$ and $\hat{b}$, we need to **choose a loss function.** Choosing **squared loss** (and hence MSE) gives the following optimal parameters:

$$\hat{b} = r\frac{\sigma_y}{\sigma_x} \qquad\qquad \hat{a} = \bar{y} - \hat{b}\bar{x}$$

Note: these are defined in terms of the correlation coefficient, **r**!

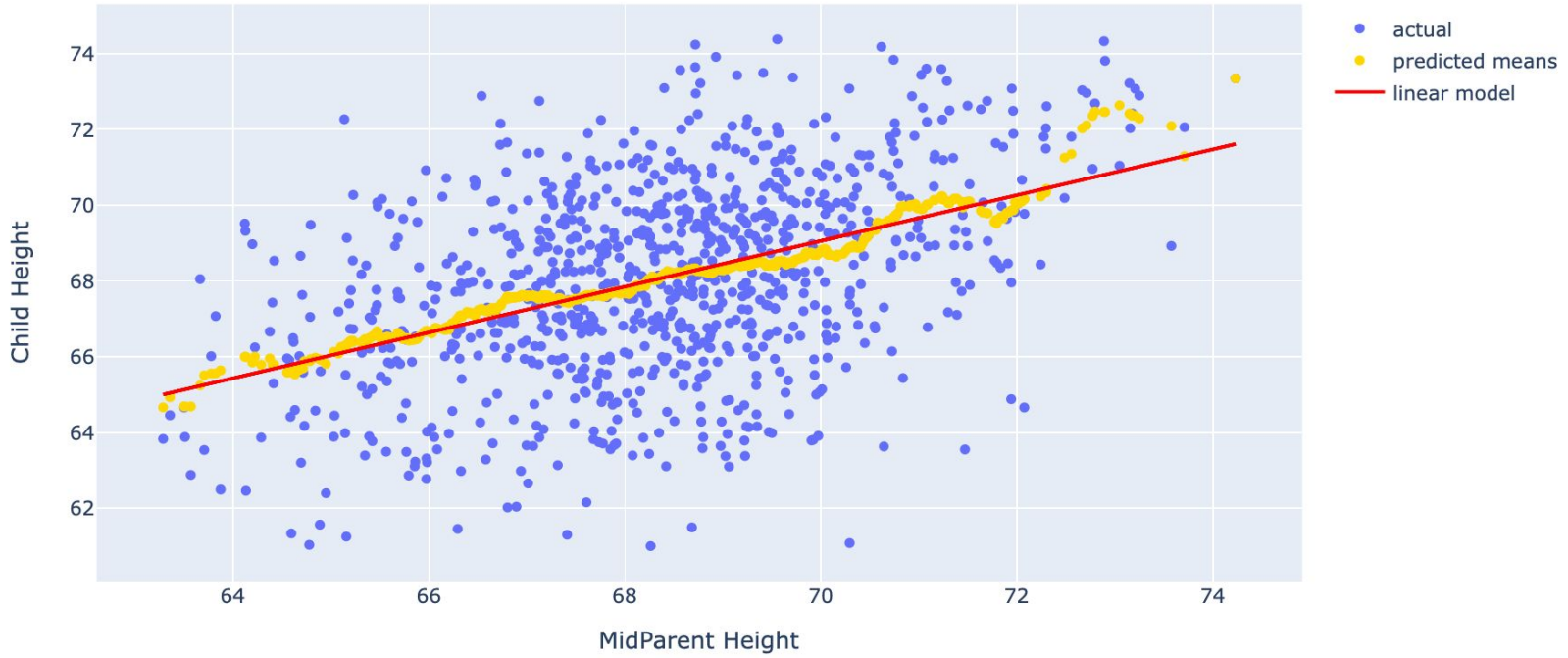
Equation of the regression line

Let's look at a **demo**. Specifically, we'll:

- Create the graph of averages you saw before.
- Implement the optimal parameters by hand.
    - In the future, we will resort to in-built packages to do this for us.

Question: We implemented the graph of averages in Data 8. Why are we doing it again?

- Answer: Because we're going to do this **yet again** when we get to logistic regression.
- This is helpful in understanding **why** we've chosen the models that we have.



Our linear model matches the predicted means quite well.

Minimizing MSE for the SLR model

Minimizing MSE for the SLR model

We will now walk through the calculus of determining the optimal parameters for the SLR model, using **squared loss**. Recall, mean squared error is of the form $MSE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$.

Since our model (for a given observation) is $\hat{y}_i = a + bx_i$, the quantity we want to minimize is:

$$R(a, b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (a + bx_i))^2$$

- Note, there are now **two** parameters we need to optimize over. We want the best **combination** of a and b such that average loss is minimized.
  - This gives us a model that fits the data as best as possible.
- We refer to this combination of model and loss as "**least squares** linear regression."

Minimizing MSE for the SLR model

One slight simplification we can make: the pair (a, b) that minimizes $R(a,b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (a + bx_i))^2$ is also the same pair (a, b) that minimizes

$$R(a,b) = \sum_{i=1}^{n}(y_i - (a + bx_i))^2$$

- The value of the objective function will be different, but what we're looking for are the optimal parameters. Those won't change when we multiply the function by a constant.
- To determine the pair (a, b) that minimizes our objective function, we need to take **partial derivatives** with respect to both parameters (a, b), set them equal to 0, and solve both equations.
- Remember, our data points $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ are constants here – they are not variables!
- The video for this lecture walks through all of the steps in the next slide(s) in depth.

Minimizing MSE for the SLR model

First, we rearrange our objective function to be slightly more convenient. We then take the derivative with respect to a, and set it equal to 0.

$$R(a,b) = \sum_{i=1}^{n}(y_i - (a + bx_i))^2 = \sum_{i=1}^{n}(y_i - bx_i - a)^2$$

$$\frac{\partial R}{\partial a} = \sum_{i=1}^{n} 2(y_i - bx_i - a)(-1) = -2\sum_{i=1}^{n}(y_i - bx_i - a)$$

$$0 = -2\sum_{i=1}^{n}(y_i - bx_i - a)$$

Then, using the properties of summations, we rearrange to solve for $\hat{a}$ Note, this is in terms of our choice of b.

$$0 = -2\sum_{i=1}^{n}(y_i - bx_i - a)$$

$$0 = \sum_{i=1}^{n}(y_i - bx_i - a)$$

$$0 = \sum_{i=1}^{n} y_i - b\sum_{i=1}^{n} x_i - na$$

$$na = \sum_{i=1}^{n} y_i - b\sum_{i=1}^{n} x_i$$

$$\hat{a} = \frac{1}{n}\sum_{i=1}^{n} y_i - b \cdot \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\hat{a} = \bar{y} - b\bar{x}$$

We now need to substitute this value into the objective function when we solve for the optimal b.

Minimizing MSE for the SLR model

First, we substitute our expression for $\hat{a}$ into the objective function:

$$\begin{aligned} R(a,b) &= \sum_{i=1}^{n}(y_i - (a + bx_i))^2 \\ &= \sum_{i=1}^{n}(y_i - (\bar{y} - b\bar{x} + bx_i))^2 \\ &= \sum_{i=1}^{n}(y_i - \bar{y} - b(x_i - \bar{x}))^2 \end{aligned}$$

Then, we take the partial derivative w.r.t. b and set it to 0:

$$\begin{aligned} \frac{\partial R}{\partial b} &= \sum_{i=1}^{n} 2 \cdot (y_i - \bar{y} - b(x_i - \bar{x})) \cdot (-1) \cdot (x_i - \bar{x}) \\ \frac{\partial R}{\partial b} &= -2\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y} - b(x_i - \bar{x})) \\ 0 &= -2\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y} - b(x_i - \bar{x})) \end{aligned}$$

And finally, we rearrange and solve for $\hat{b}$

$$\begin{aligned} 0 &= \left[\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) - b(x_i - \bar{x})^2\right] \\ 0 &= \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) - b\sum_{i=1}^{n}(x_i - \bar{x})^2 \\ 0 &= nr\sigma_x\sigma_y - bn\sigma_x^2 \\ b\sigma_x^2 &= r\sigma_x\sigma_y \\ \hat{b} &= \frac{r\sigma_x\sigma_y}{\sigma_x^2} = r\frac{\sigma_y}{\sigma_x} \end{aligned}$$

Looks familiar!

Minimizing MSE for the SLR model

We've now shown that when using **squared loss** as our loss function, the optimal parameters for the model $\hat{y} = \hat{a} + \hat{b}x$ are given by

$$\hat{b} = r\frac{\sigma_y}{\sigma_x} \qquad \hat{a} = \bar{y} - \hat{b}\bar{x}$$

- If we used a loss function other than squared loss, we'd end up with different optimal parameters!
- This process of determining optimal model parameters by hand is something you should be able to do on your own.
  - Change the model, change the loss, and try it yourself!
- Note: We can also rewrite our model as follows, showing that r is the slope of the regression line in standard units:

$$\frac{\hat{y} - \bar{y}}{\sigma_y} = r\left(\frac{x - \bar{x}}{\sigma_x}\right)$$

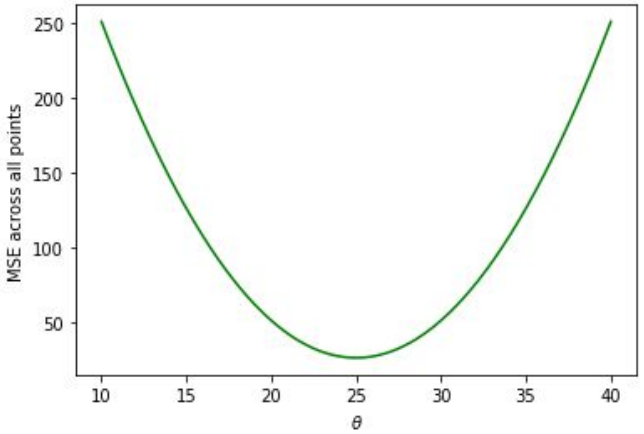
Visualizing loss surfaces

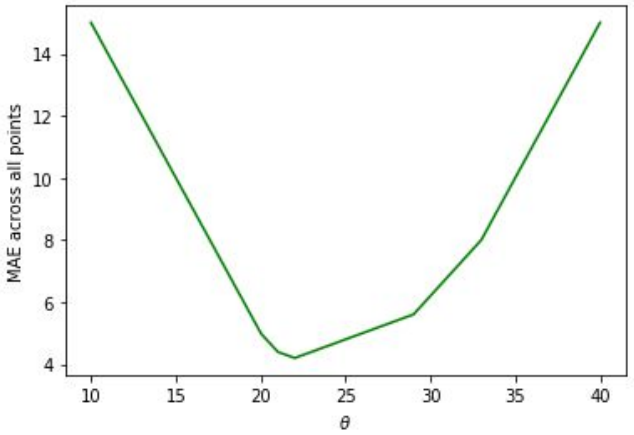
Visualizing loss surfaces

On the left, we have the plots of the **loss surfaces** for the constant model (from last lecture).

- Top: squared loss (so average loss = MSE).
  - The y-axis shows the MSE for each value of theta on the x-axis.
- Bottom: absolute loss (so average loss = MAE).

The simple linear regression model has two parameters, a and b (or equivalently, θ_0 and θ_1). This means the loss surface will be **3D!**





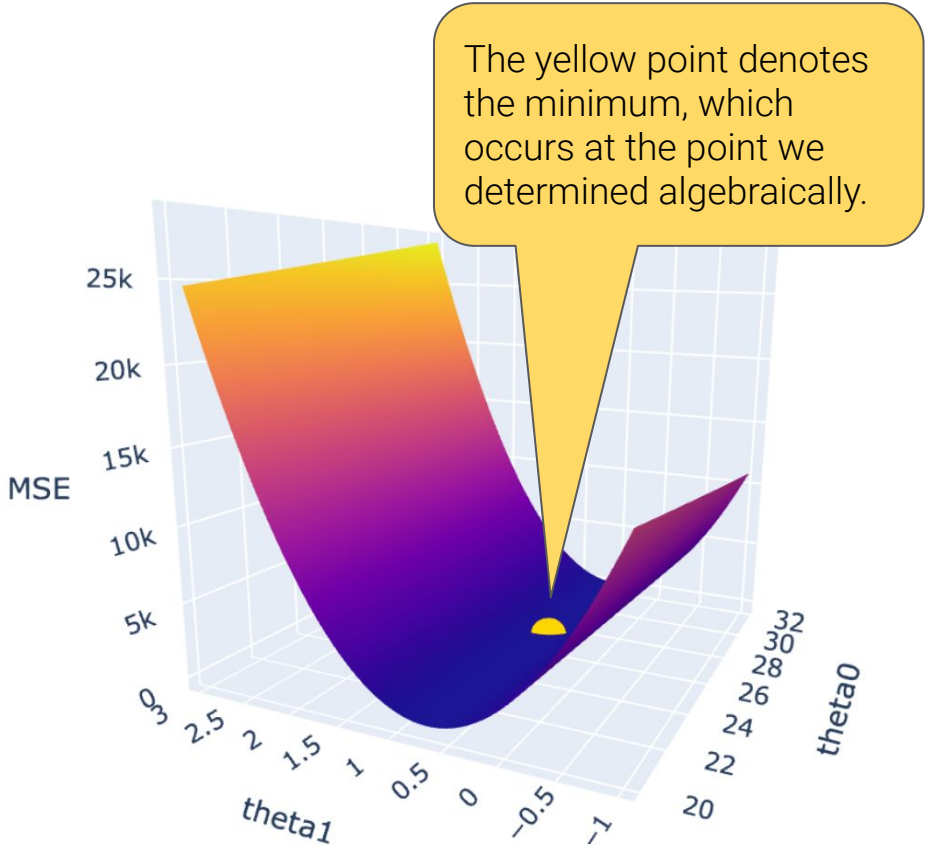
Visualizing loss surfaces

Here, we have 3 axes.

- One for θ_0.
- One for θ_1.
- One that tells us the mean squared error on our dataset, using the model $\hat{y} = \theta_0 + \theta_1 x$.

The loss surface is nice and smooth (which we touted as a property of the squared loss in the last lecture).

Let's look at a **demo** of this in code.



Model interpretation

Interpreting slopes

$$\text{slope } = r\frac{\sigma_y}{\sigma_x}$$

The slope is measured in **units of y per unit of x.**

- For instance, suppose we survey several individuals for their weight and height, and we want to use weight (x) to predict height (y).
  - Another way of saying this is “regressing height on weight.”
- The units of our slope could be **inches per pound**.
  - In a standard line $y = a + bx$, the slope (b) measures the increase in y for a 1 unit increase in x.

Using the above example, suppose our model turns out to be

$$\text{predicted height} = 56 + 0.09 \cdot \text{weight}$$

Interpreting slopes

$$\text{predicted height} = 56 + 0.09 \cdot \text{weight}$$

Does this mean that if someone in the dataset puts on 1 pound, we estimate that they will get 0.09 inches taller? **No!**

- The model we created shows **association**, not causation.
- The data we collected is a snapshot of several people at one instance of time (cross-sectional), not snapshots of people over time (longitudinal).

What does this mean, then?

- 0.09 inches is the estimated height difference between two people whose weights are one pound apart.

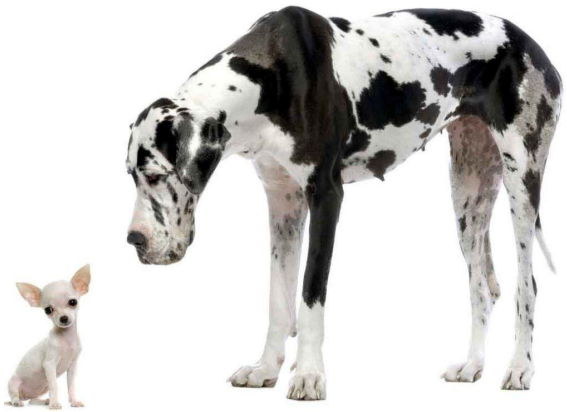
New data needs to be similar to original data

Suppose we fit a model that predicts a Chihuahua's weight given its length.

$$\text{predicted weight} = 3 + 2 \cdot \text{length}$$

Should we use this model to predict the weight of Great Danes?

- No – we have no indication that the weight vs. length relationship for Great Danes are the same as Chihuahuas.
- Great Danes' weights and lengths are well outside of the range of weights and lengths we fit our model on.
- **If the new data we test our model on looks nothing like the data we fit our model on, there's no guarantee that it will be any good.**
    - This is a notion we will formalize in a few lectures.

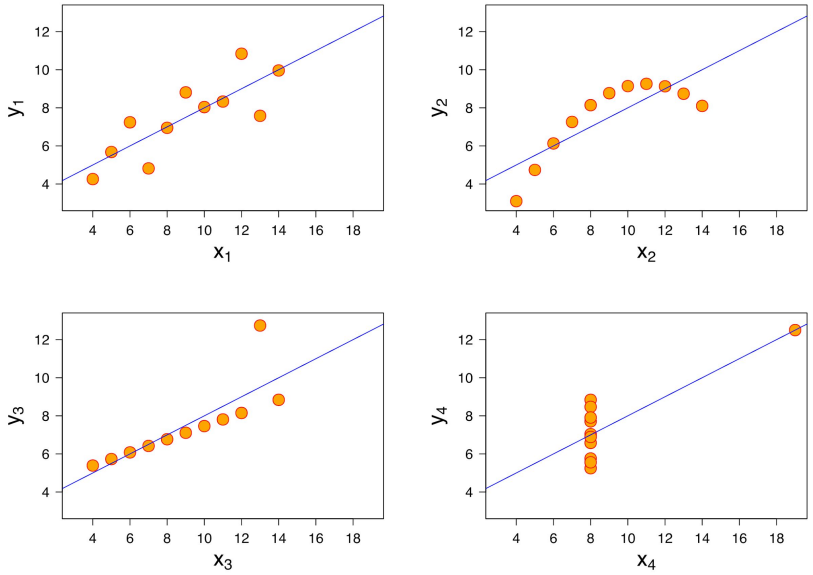


Chihuahuas (left) range from 3-6 pounds, and 9.5-15 inches in length. **Great Danes** (right) range from 110-175 pounds, and 35.5-43 inches in length.

Visualize, **then** quantify!

Anscombe's quartet refers to the following four sets of points on the right.

- They each have the same mean of x, mean of y, SD of x, SD of y, and r value.
- Since our optimal SLR model only depends on those quantities, they all have the same regression line.
- However, the SLR model only makes sense as a model for one of these four sets of points.
- **Before modeling, you should always visualize your data first!**



$$\bar{x} = 9, \bar{y} = 7.501$$

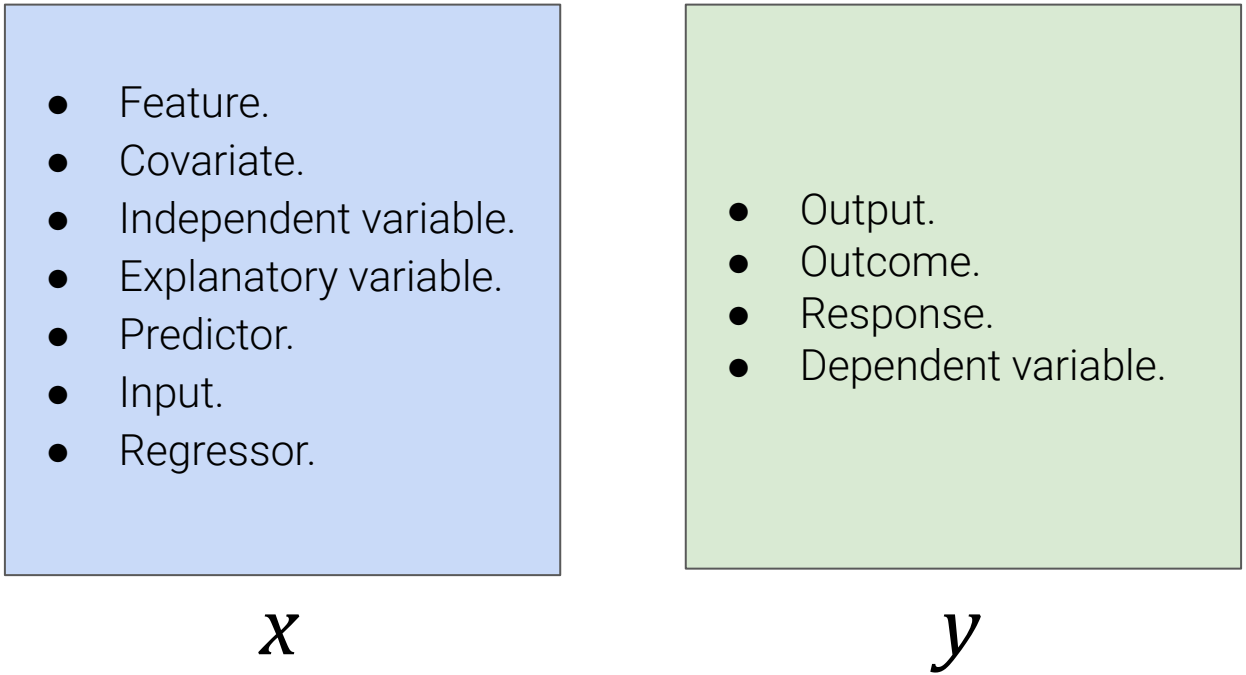
$$\sigma_x = 3.162, \sigma_y = 1.937$$

$$r = 0.816$$

Multiple linear regression

Terminology

There are several equivalent terms in the regression context. You should be aware of them.



Adding independent variables

First, some terminology. For our purposes, all of these terms mean the same thing:

- Feature.
- Covariate.
- Independent variable.
- Explanatory variable.
- Predictor.
- Input.
- Regressor.

In the regression context, each of the above things has a "**weight**" assigned to it, given by the **parameter**. We also call these weights "**coefficients**." For instance, in $\hat{y} = \theta_0 + \theta_1 x$, we might say the "weight" associated with the constant/intercept term is θ_0 , and the "weight" associated with the x term is θ_1 .

Adding independent variables

A linear regression model with two features (and thus, three parameters), is of the form

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

For example, suppose we want to create a linear regression model that predicts the number of points a player in the NBA averages (PTS). Using just the number of assists (AST) they average might yield a model of the form

$$\text{predicted PTS} = 3.98 + 2.4 \cdot \text{AST}$$

If we use both AST and the number of 3PT field goal attempts they make (3PA), we may have

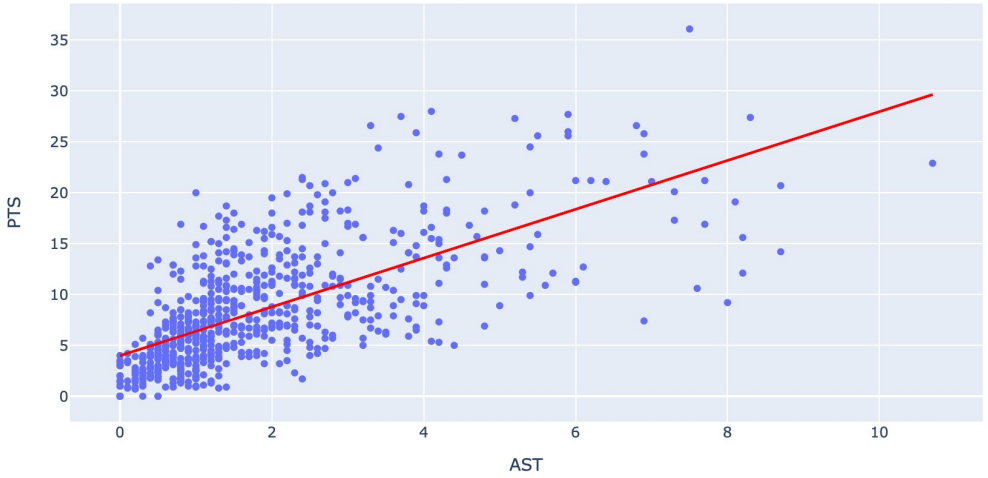
$$\text{predicted PTS} = 2.163 + 1.64 \cdot \text{AST} + 1.26 \cdot \text{3PA}$$

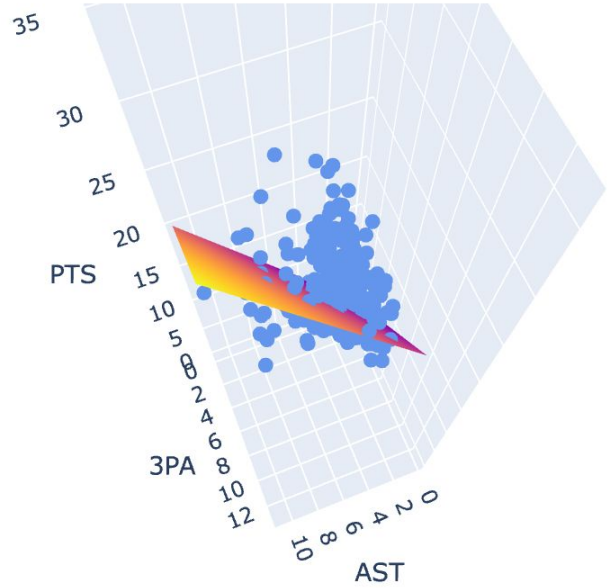
(These coefficients were determined by minimizing average squared loss, in the companion notebook.)

Visualizing higher-dimension models

In both of the below plots, the blue circles represent the true observations.

- On the left, the red line represents the model obtained by using only AST.
- On the right, since we now have two independent variables, our model is a plane in 3D.





Multiple linear regression

In general, the **multiple linear regression** model is of the form

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \ldots + \theta_p x_p = \theta_0 + \sum_{j=1}^{p} \theta_j x_j$$

- We say this model has p features, plus an intercept term.
- The weight associated with feature x_j is θ_j .
- If we set $x_0 = 1$ for each observation, then we can simplify further:

$$\hat{y} = \sum_{j=0}^{p} \theta_j x_j$$

- This is the notation we will use moving forward.
  - Think about how you can rewrite this in terms of a vector multiplication!

Be careful: x_j here refers to feature j, not data point j.

Multiple linear regression

Model 1: $\text{predicted PTS} = 3.98 + 2.4 \cdot \text{AST}$

different!

Model 2: $\text{predicted PTS} = 2.163 + 1.64 \cdot \text{AST} + 1.26 \cdot \text{3PA}$

These are different models! In general, $\hat{\theta}_j$ in one model will not be equal to $\hat{\theta}_j$ in another model.

- 2.4 is the slope of the relationship between AST and PTS, when only considering those two variables.
  - Parameters [3.98, 2.4] minimize average squared loss for Model 1.
- 1.64 is the slope of the relationship between AST and PTS, when also considering 3PA.
  - Parameters [2.163, 1.64, 1.26] minimize average squared loss for Model 2.

General notation

Our models can be expressed as a function $\hat{y} = f_\theta(x)$ of an input variable, x.

Constant model: $f_\theta(x) = \theta$

Simple linear regression model: $f_\theta(x) = \theta_0 + \theta_1 x$

Multiple linear regression model: $f_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_p x_p$

- Note: In the latter two models, θ is a vector, not a scalar! In the last model, x is a vector too. We will formalize this in the next lecture.
- We denote the prediction function that uses the optimal choice of parameters for a given model with $f_{\hat{\theta}}(x)$. For instance, for the SLR model, $f_{\hat{\theta}}(x) = \hat{\theta}_0 + \hat{\theta}_1(x)$.

RMSE and Multiple R^2

Evaluating models

What are some ways to determine if our model was a good fit to our data?

- Look at MSE or RMSE.
- Look at the correlations.
- Look at a residual plot.
  - Residuals are defined as being the difference between actual and predicted y valu $e_i = y_i - \hat{y}_i$
    - ■ .
  - Next lecture!

Root Mean Squared Error (RMSE)

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

Root mean squared error is defined as being the square root of the mean squared difference between predictions and their true values.

- It is the square root of MSE, which is the average loss that we've been minimizing to determine optimal model parameters.
- RMSE is in the same units as y.
- A lower RMSE indicates more "accurate" predictions.
    - Lower average loss across the dataset.

Comparing RMSEs

- For the constant model with squared loss, RMSE is σ_y .
  - MSE(sample mean) = sample variance.
  - This is a good baseline to compare with.
- Using just the data we **trained our model on**, it is impossible for RMSE to go up by adding features.
  - If a new feature (e.g. “does a player like the color red?”) we’ve added doesn’t help lower average loss, its weight will just be set to 0.
  - When we start evaluating models on unseen data, this is no longer true.
    - We will see why in ~3 lectures.
- Soon, we will look at “training error” and “testing error”. The errors that we look at are RMSEs.

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

$$\text{predicted PTS} = 3.98 + 2.4 \cdot \text{AST}$$

Has an RMSE of **4.29** on the NBA dataset.

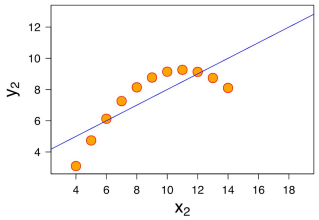
$$\text{predicted PTS} = 2.163 + 1.64 \cdot \text{AST} + 1.26 \cdot \text{3PA}$$

Has an RMSE of **3.64** on the NBA dataset.

Multiple R^2

When we had just one feature (x), we were able to look at the correlation coefficient r to get a sense of how strong the linear association between x and y was.

- The further r was from 0, the stronger the linear association between x and y.
    - Looking at r alone isn't enough. See Anscombe's quartet.
- Here we have multiple features. We *could* (and sometimes do!) look at the correlation between each feature and our true y values individually.
- However, we are also interested in measuring the strength of the linear association between our actual y and predicted y.
    - We want this relationship to be as close to the line $y = x$ as possible.



$r = 0.816$

Multiple R^2

We define the **multiple R^2** value as the square of the **correlation** between the true y and predicted y. This is also referred to as the **coefficient of determination**.

$$R^2 = [r(y, \hat{y})]^2$$

Since it is the square of a correlation coefficient (which ranged between -1 and 1), R^2 ranges between 0 and 1. Another way of expressing R^2, in linear models that have an intercept term, is

$$R^2 = \frac{\text{variance of fitted values}}{\text{variance of } y} = \frac{\sigma^2_{\hat{y}}}{\sigma^2_y}$$

Thus, we can interpret R^2 as the **proportion of variance** in our true y that our **fitted values** (predictions) capture, or "the proportion of variance that the **model explains**."

Multiple R^2

- As we add more features, our fitted values tend to become closer and closer to our actual y values. Thus, R^2 increases.
  - The simple model (AST only) explains 45.7% of the variance in the true y.
  - The AST & 3PA model explains 60.9%.
- Adding more features doesn't always mean our model is better, though!
  - We are a few lectures away from understanding why.
  - "Adjusted R^2" accounts for this (see Stat 151A).

$$R^2 = \frac{\text{variance of fitted values}}{\text{variance of } y}$$

$$\text{predicted PTS} = 3.98 + 2.4 \cdot \text{AST}$$

$R^2 = 0.457$

$$\text{predicted PTS} = 2.163 + 1.64 \cdot \text{AST} + 1.26 \cdot \text{3PA}$$

$R^2 = 0.609$

Summary

Summary

- We now know of three models, $\hat{y} = f_\theta(x)$.
  - The constant model, $f_\theta(x) = \theta$.
  - The simple linear regression model $f_\theta(x) = \theta_0 + \theta_1 x$.
  - The multiple linear regression model $f_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_p x_p$.
  - A model with optimal parameters is denoted $f_{\hat{\theta}}(x)$.
- We looked at the correlation coefficient, r, and studied its properties.
- We solved for the optimal parameters for the simple linear model by hand, by minimizing average squared loss (MSE).

$$\hat{\theta}_1 = r\frac{\sigma_y}{\sigma_x} \qquad \hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x}$$

- We introduced the notion of a feature, and how we can have multiple in our models.
- We discussed the multiple R^2 coefficient and RMSE as methods of evaluating the quality of a linear model.

Next time

In the next lecture, we will…

- Express the multiple linear regression model using matrix-vector notation.
- Explicitly solve for the optimal parameters.
  - Thus far, we've done this by hand for the constant and simple linear models.
  - The process of solving for the optimal parameters will inform us of several properties of linear models!
  - **There will be a lot of linear algebra!**
- Look at residuals and their properties.
- Think about what it means for a model to be "linear."