Statistical Learning: Linear and Logistic Regression

Jesse Gronsbell
Department of Statistical Sciences, University of Toronto

What we learned yesterday

- Definitions of machine learning (ML), statistical learning, and data science
- Representing data in a matrix and corresponding mathematical notation
- Linear regression and least squares (LS) estimation

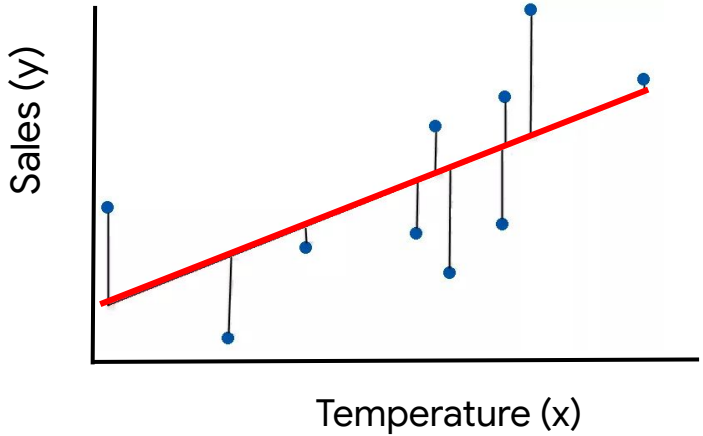
What we'll cover today

- How to compute or *estimate* m and b for the linear model with LS
- More on LS and linear regression
- Logistic regression

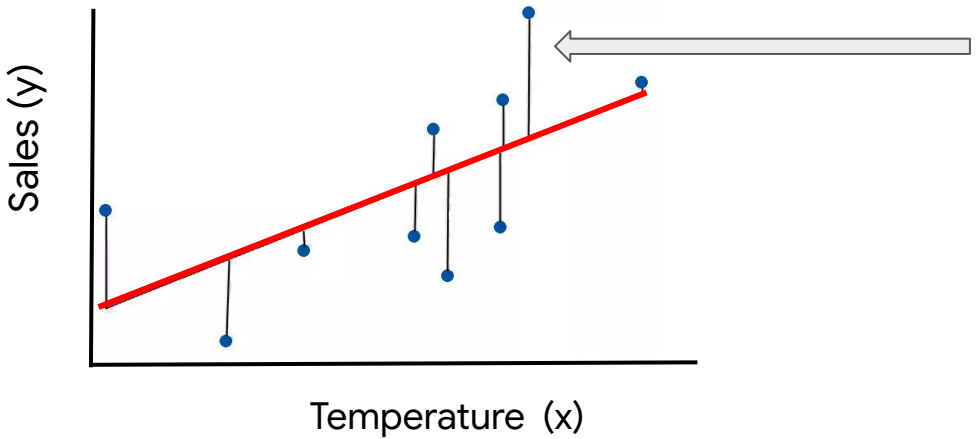
Part I: Least Squares cont’d

Recap: Linear regression problem from yesterday

Use the data on sales and temperature to estimate the unknown **parameters *m*** and ***b*** in the **linear model, expected sales = m*temperature + b**



How do we find m and b using the data?



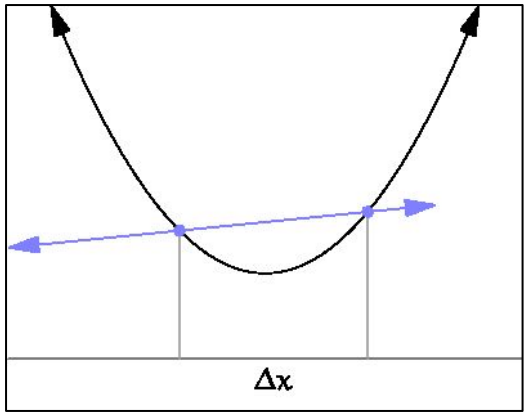
Error:
$[y_i - (mx_i + b)]^2$

Find ***m*** and ***b*** so that the sum of the errors is as small as possible:

$$\sum_{i=1}^{n} [y_i - (mx_i + b)]^2 = [y_1 - (mx_1 + b)]^2 + [y_2 - (mx_2 + b)]^2 + \ldots + [y_n - (mx_n + b)]^2$$

Calculus in 60 seconds: The derivative

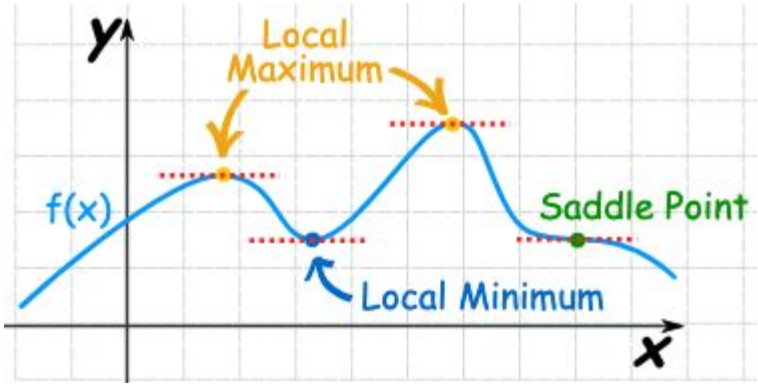
- The **derivative** is the **rate of change** of a function in a **tiny neighborhood** around a given point



https://en.wikipedia.org/wiki/Derivative

Calculus in 60 seconds: The derivative

- The **derivative** is the **rate of change** of a function in a **tiny neighborhood** around a given point



- When the **derivative is 0**, the function is flat and the point is either a:
  - Local maximum
  - Local minimum
  - Saddle point

https://www.mathsisfun.com/calculus/maxima-minima.html

Finding the LS estimates for m and b

1. Take the derivatives of our **objective function** $\sum_{i=1}^{n}[y_i-(mx_i+b)]^2$ with respect to **m** and **b**

2. Find the **m** and **b** where the derivatives are zero

3. Confirm that you have found a minimum value

The math… in case you're interested

Lecture notes:

https://www.amherst.edu/system/files/media/1287/SLR_Leastsquares.pdf

Step-by-step video:

https://www.youtube.com/watch?v=ewnc1cXJmGA

Let’s try least squares in R!

Linear regression: A little bit more detail

- Notice that I wrote *expected* sales = m*temperature + b
- More formally, this is written as

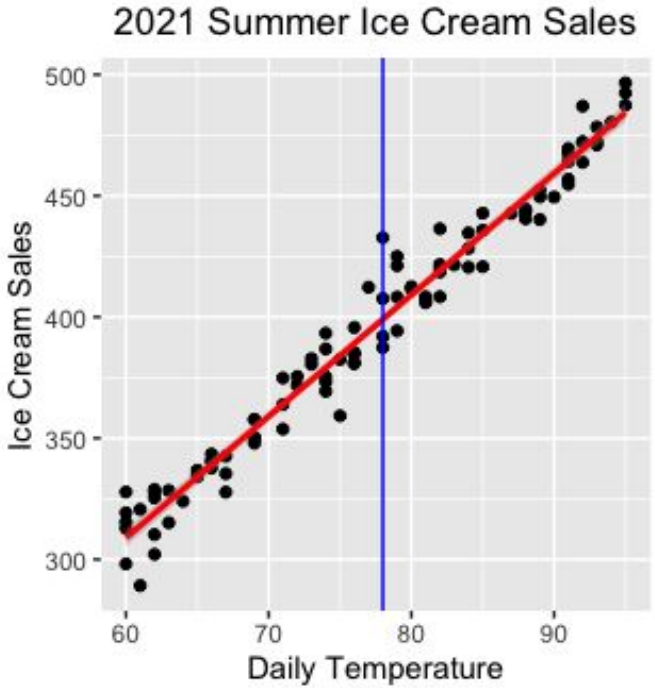
$$E(\text{sales} \mid \text{temperature}) = m*\text{temperature} + b$$

where E denotes *expectation*

- We interpret E(sales | temperature) as the average value of sales for a given temperature
    - Just like when we talk about what ‘tends to happen’ in everyday conversation

Why the expectation?

There is some randomness in the process we are observing!



We don’t always sell the exact same amount of ice cream on a 78 degree day. On average, we sell $399.09.

Another way to write a linear regression

- More generally, we have been writing the linear regression model as

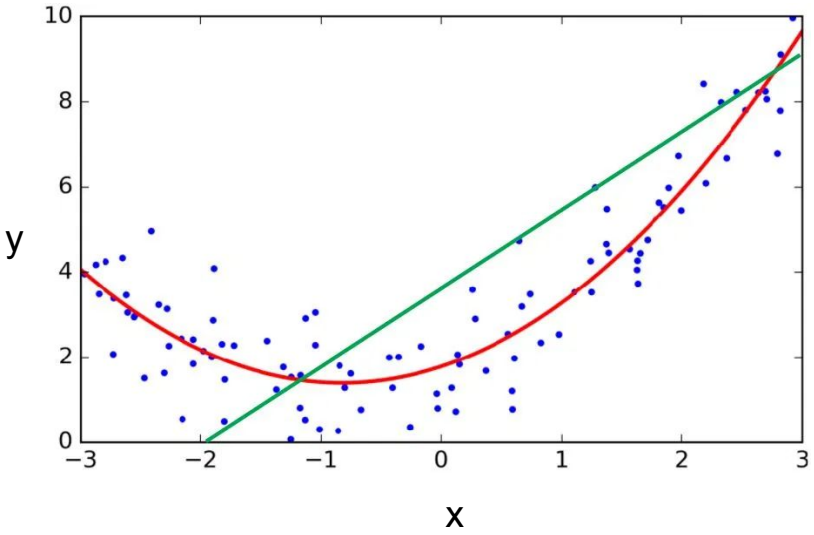
$$E(y \mid x) = m*x + b$$

- The model can also be expressed as

$$y = m*x + b + \varepsilon$$

where ε is random error

Flexibility of least squares: polynomial regression



- We can adapt our regression model for more **complex patterns** by adding additional terms

- **Ex**. we can fit a **polynomial model** by adding higher order terms

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$$

- We can use least squares to find estimates of β_0, β_1, and β_2

Generalizing the thinking: empirical risk minimization

- LS is an example of **empirical risk minimization (ERM)**

- **ERM:** Finding parameters that minimize a specified empirical loss function
  - Empirical = based on the data
  - **Ex**: LS is ERM with the squared loss function

- Just like LS, the intuition is to find the **parameters that match the data set well** based on the particular loss function

Part II: Logistic Regression

Choosing a model

- We saw how to use least squares to estimate a linear or polynomial model

- How would we actually choose between models IRL?

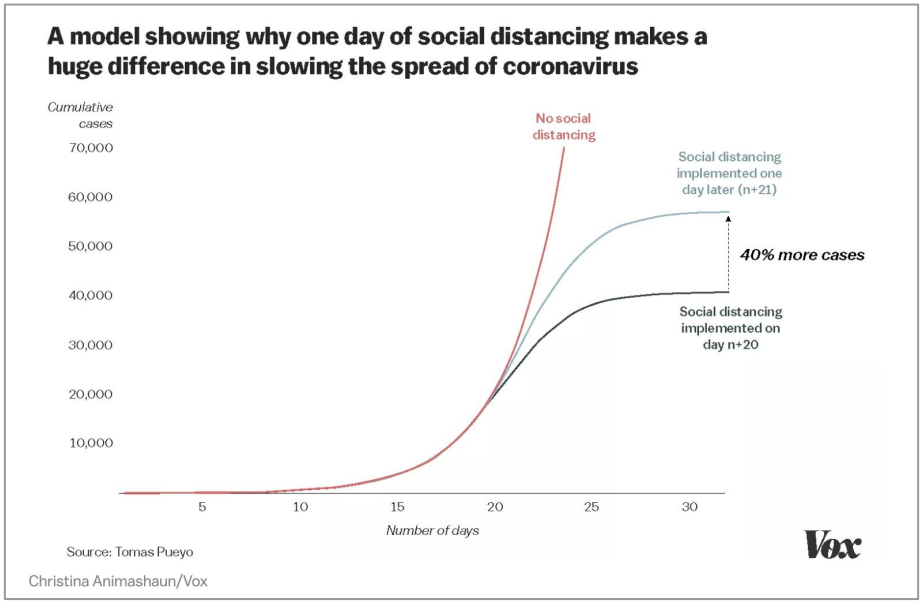
- Models are chosen to align with:
  - The question of interest
  - The outcome
  - The covariates

Thinking about the question of interest

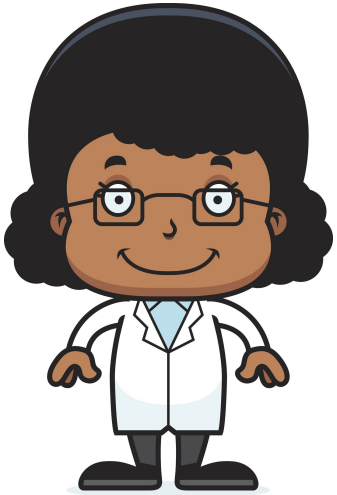
- In the ice cream example, we observed a linear trend and fit a linear model

- An exponential model would be more appropriate for things that speed up their growth rate through the time

  ex. An infectious disease like COVID-19



We often work with domain experts when modeling



Thinking about the outcome

- Different models accommodate **different types of outcomes**

- In the ice cream example, sales is a ***continuous outcome***
  - The value of sales for any given day is a value between (0,∞)
  - We won't observe the lower bound because ice cream is delicious
  - ... we might observe the upper bound if the ice cream was EXTREMELY tasty (JK!)

Other types of outcomes

- **Binary: 2 choices**
  - yes/no, did occur/didn't occur, blue/red, etc.
  - Typically represented as 0/1 or -1/1 in your data

Other types of outcomes

- **Binary: 2 choices**
  - yes/no, did occur/didn't occur, blue/red, etc.
  - Typically represented as 0/1 or -1/1 in your data

- **Categorical: Fixed number of choices**
  - yes/no/maybe, did occur/didn't occur/unclear/unobserved, blue/red/green/yellow/purple, etc.

Other types of outcomes

- **Binary: 2 choices**
  - yes/no, did occur/didn't occur, blue/red, etc.
  - Typically represented as 0/1 or -1/1 in a your data

- **Categorical: Fixed number of choices**
  - yes/no/maybe, did occur/didn't occur/unclear/unobserved, blue/red/green/yellow/purple, etc.

- **Count: Non-negative integer valued (i.e. 0, 1, 2, 3, ...)**
  - Number of events in a given time period

Note: Covariates are also classified in the same way.

Classification: modeling a binary outcome

- Going back to the ice cream example, suppose we instead want to model **whether we make $400** in a day based on the temperature

- The outcome, **y**, is **binary** as we either did or did not make $400 dollars

Classification: modeling a binary outcome

- Going back to the ice cream example, suppose we instead want to model **whether we make $400** in a day based on the temperature

- The outcome, **y**, is **binary** as we either did or did not make $400 dollars

- We now represent the outcome as

  y = 1 if we made $400 dollars **OR** 0 if we didn't make $400 dollars

Can we use least squares?

Let's try a simple example:

Temperature	Made $400?
72	0
78	1
59	0
86	1
91	1

Can we use least squares?

Let's try a simple example:

Temperature	Made $400?
72	0
78	1
59	0
86	1
91	1

LS yields the following linear model:

$$y = 0.03733x + -2.282$$

For a 60 degree day, our prediction is:

$$y = 0.03733(60) + -2.282 = -0.0422$$

Yikes - this isn't even a value between 0 and 1!

The quick fix - slap a link function on the model!

- The linear model took the form

$$E(y \mid x) = mx + b$$

and doesn't guarantee that a prediction will fall within [0,1]

The quick fix - slap a link function on the model!

- The linear model took the form

$$E(y \mid x) = mx + b$$

  and doesn't guarantee that a prediction will fall within [0,1]

- We can add a **link function**, g(), to ensure that our predictions will fall within the desired range

$$E(y \mid x) = g(mx + b)$$

The quick fix - slap a link function on the model!

- The linear model took the form

$$E(y \mid x) = mx + b$$

  and doesn't guarantee that a prediction will fall within (0,1)

- We can add a **link function**, g(), to ensure that our predictions will fall within the desired range

$$E(y \mid x) = P(y = 1 \mid x) = g(mx + b)$$

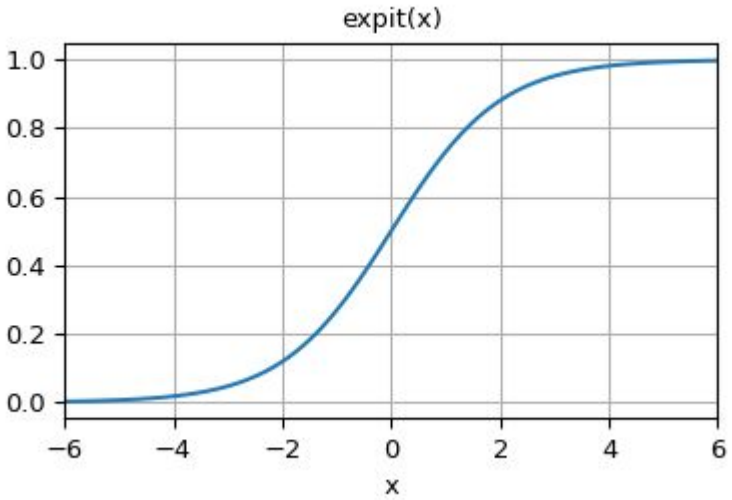
- With a **binary** outcome, we pick g() to **map values of mx + b to (0,1)**

Logistic regression

- With g() as the expit function, we are fitting a **logistic regression model**
- That is,

$$\mathbf{g(mx+b) = e^{mx+b}/(1+e^{mx+b})}$$

- Note that $e^{mx+b}/(1+e^{mx+b})$ is bounded between 0 and 1



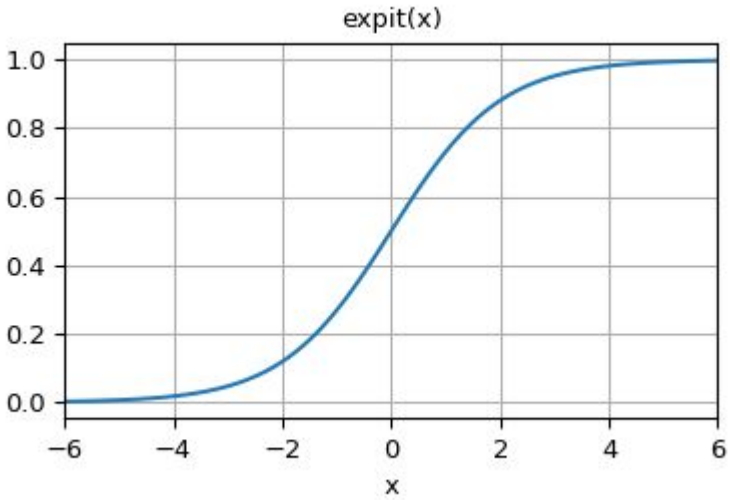
Logistic regression

- With g() as the expit function, we are fitting a **logistic regression model**
- That is,

$$\mathbf{g(mx+b) = e^{mx+b}/(1+e^{mx+b})}$$

- Note that $e^{mx+b}/(1+e^{mx+b})$ is bounded between 0 and 1

Fun fact: The name logistic comes from the fact that the inverse of g() is the logistic function



How do we estimate m and b?

- Estimation is a little more complicated now that we have g() floating around

- Instead of least squares, we often use the method of **maximum likelihood estimation**

- Basic idea is still to find an **m** and **b** that best fit the observed data
  - Instead of thinking about minimizing the error, we find the **m** and **b** that are most likely to have yielded the data we saw

Let’s try logistic regression in R!

Next up...

We will review some ways to think about covariates in our modeling!