Gradient Descent

Data Science Immersive

FLATIRON SCHOOL

Motivation

- OLS regression (one variable) $$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

- How do we actually find these beta values?
  - Linear algebra:

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_n \end{bmatrix} \qquad X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \cdots & \cdots \\ 1 & x_n \end{bmatrix} \qquad A = \begin{bmatrix} b \\ m \end{bmatrix} \qquad E = \begin{bmatrix} e_1 \\ e_2 \\ \cdots \\ e_n \end{bmatrix}$$

This gives us the matrix equation: $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{A} + \boldsymbol{E}$.

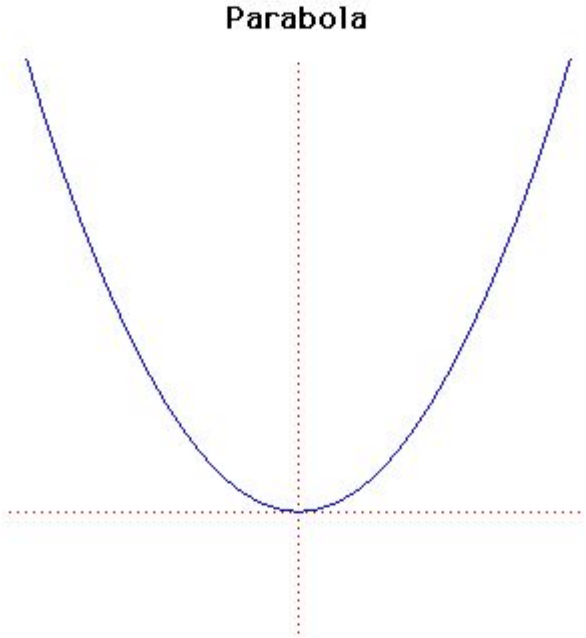
Actually, no.

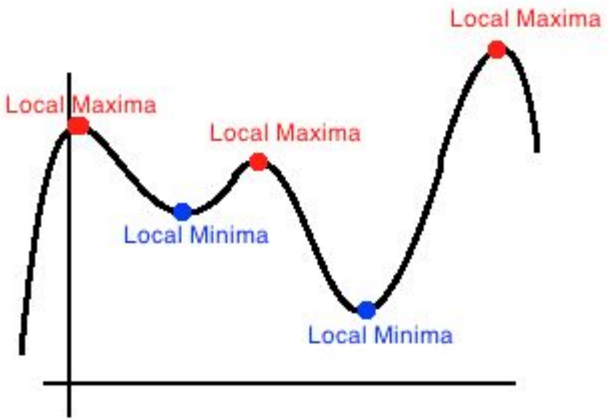
- As it turns out, solving in this way although very convenient, a direct solution becomes more and more computationally difficult as data sets get bigger.

- “Minimize cost function”

$$RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

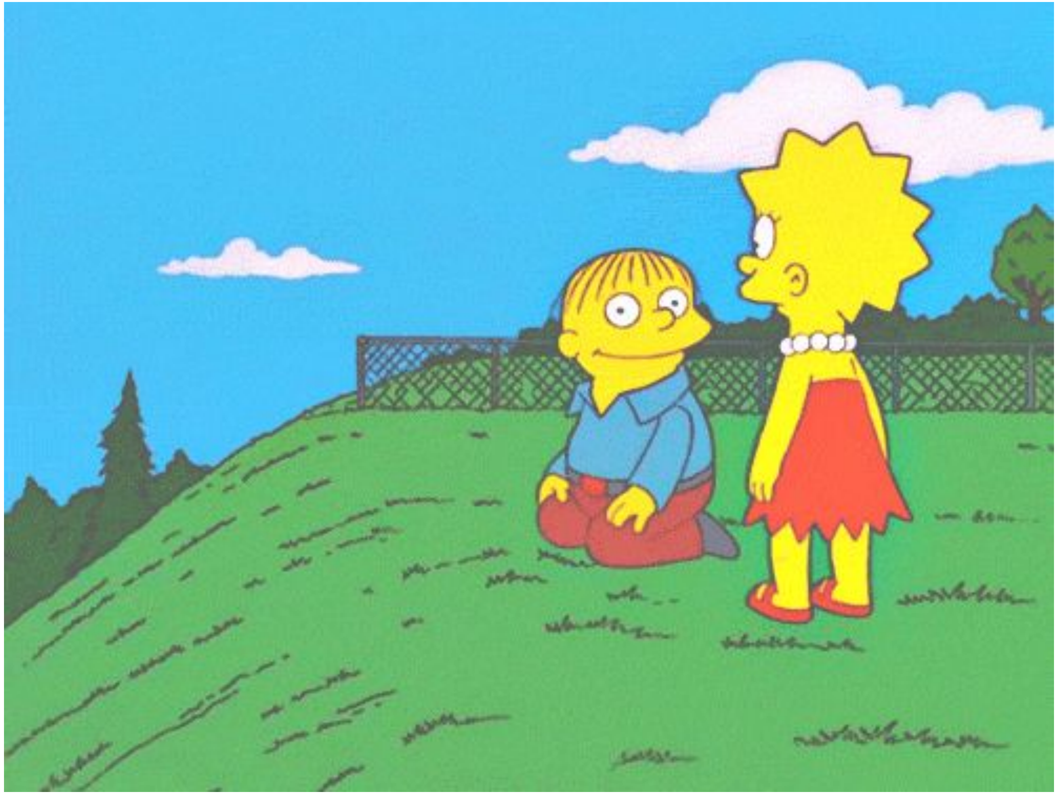
Minimizing a Function



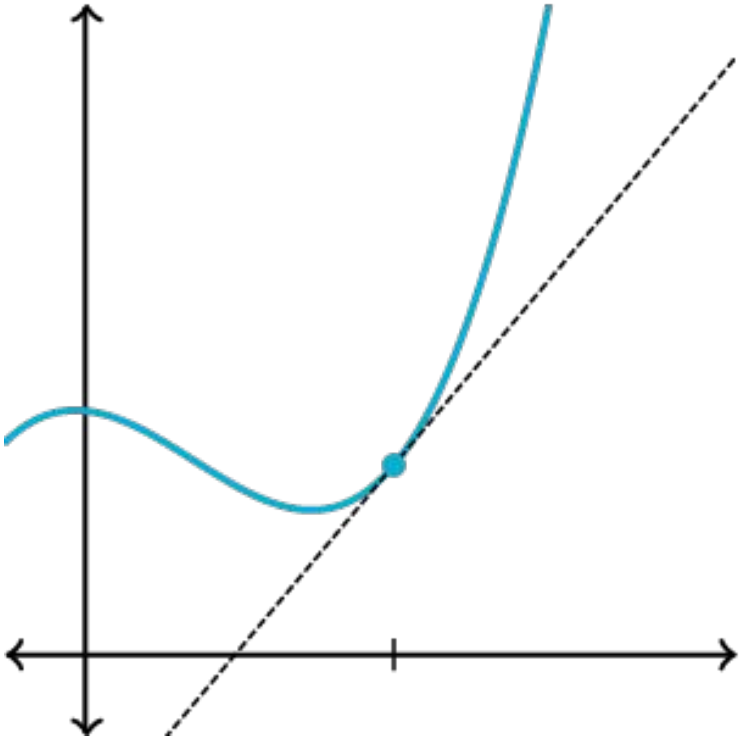


- To find the maxima and minima of a function, find where the derivative equals zero. Very easy for a convex function like the one on the left.
- But, what if you can't use this method?

Minimizing a Function



Derivatives



Derivatives

$$\lim_{x \to a} \frac{f(x) - f(a)}{x - a}$$

$$\lim_{h \to 0} \frac{f(a + h) - f(a)}{h}$$

Derivatives

- Chain Rule $$\frac{d}{dx}f(g(x)) = f'(g(x))g'(x)$$
- Power rule $$\frac{d}{dx}x^n = nx^{n-1}$$
- Example: $$\frac{d}{dx}x^2$$

Gradient Descent formula

Gradient descent algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \qquad (\text{for } j = 0 \text{ and } j = 1)$$

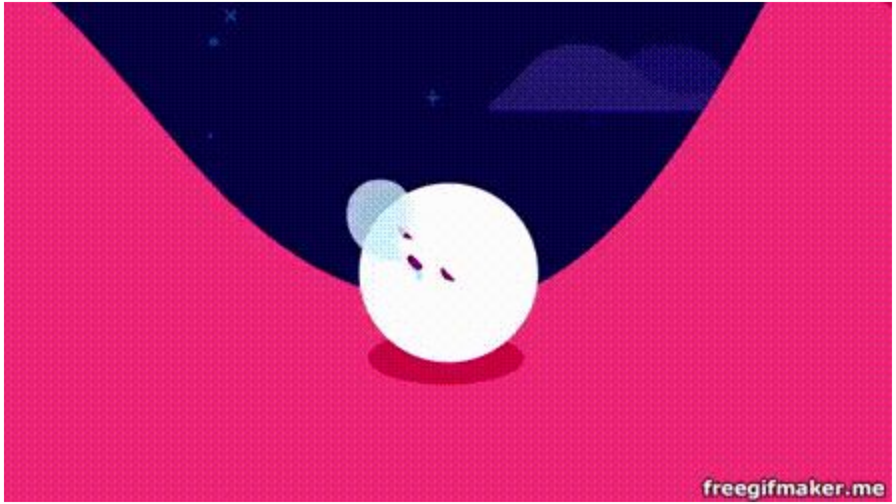
}

- The weird equal sign
- Thetas
- Partial derivative
- Alpha
- J

Learning rate, α

Potential problems:

- Overshooting the minimum so GD diverges instead of converging
- Getting stuck at a local minimum
- Not approaching minimum fast enough



Gradient Descent for Linear Regression

Gradient descent algorithm

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \qquad (\text{for } j = 0 \text{ and } j = 1)$$

}

Cost

$$J(\theta) = 1/2m \sum_{i=1}^{m} (h(\theta)^{(i)} - y^{(i)})^2$$

Simple Linear Regression with GD

$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i) * x_i$$

Code Example