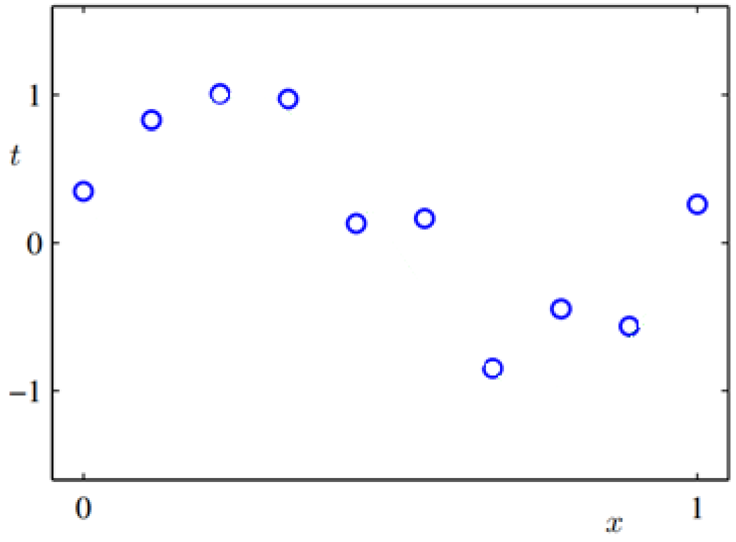
Linear Regression

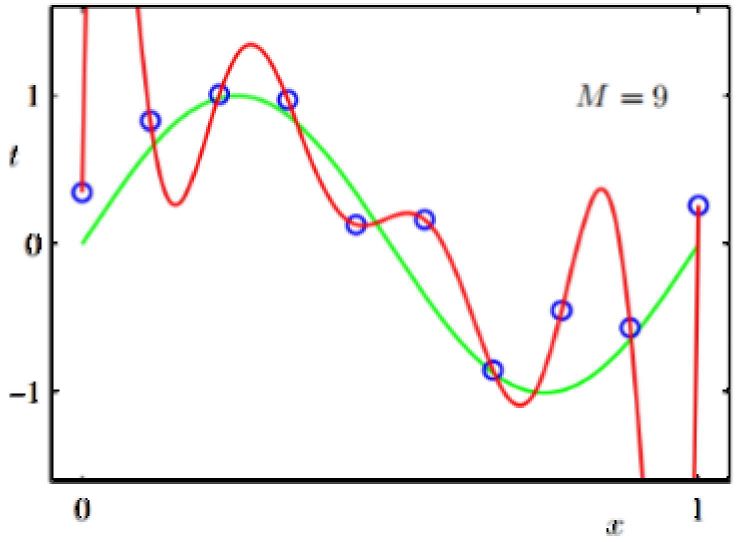
Example: Polynomial Curve Fitting

- Lets create some datapoints from the function $t(x) = \sin(2\pi x)$ with some added noise
- Green line represents the function
- Blue dots are datapoints (9)



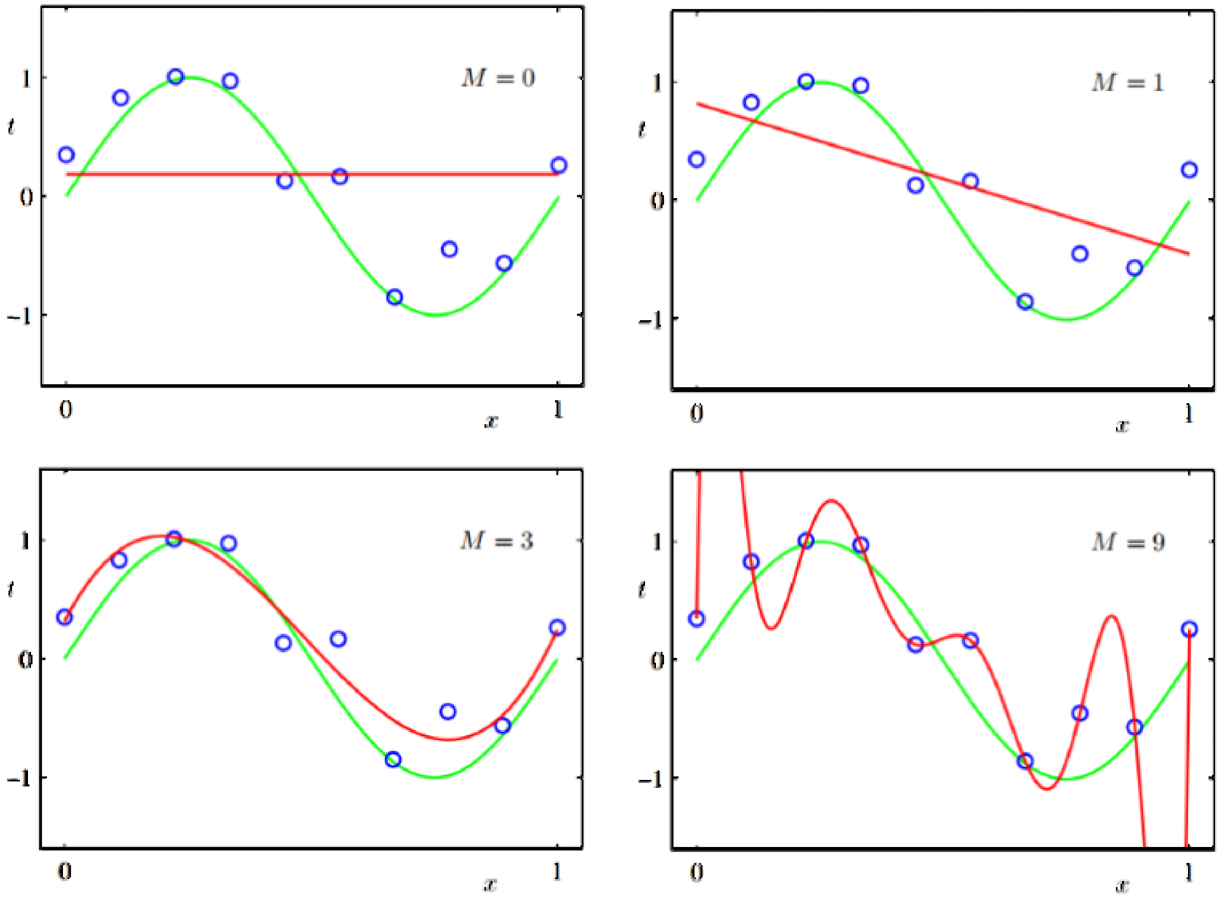
Example: Polynomial Curve Fitting

- We will try to extrapolate the original function so that we can predict values for other values of x. How?
- Lets start with a polynomial.
- What polynomial?
  - Degree 0
  - Degree 1
  - Degree 3
  - Degree 9



Example: Polynomial Curve Fitting

- How do we compute those red lines?
- Each of those polynomials is generated minimizing the error produced
- This is call **regression**



Linear Regression

- Approach for finding a linear relationship between input and output
- Linear: Predicted parameters are linear (Power = 1)
- Regression: Predicted parameters are real (Not discrete)
- Gradient Descent: Algorithm will look for the bottom of an error function

Linear Regression

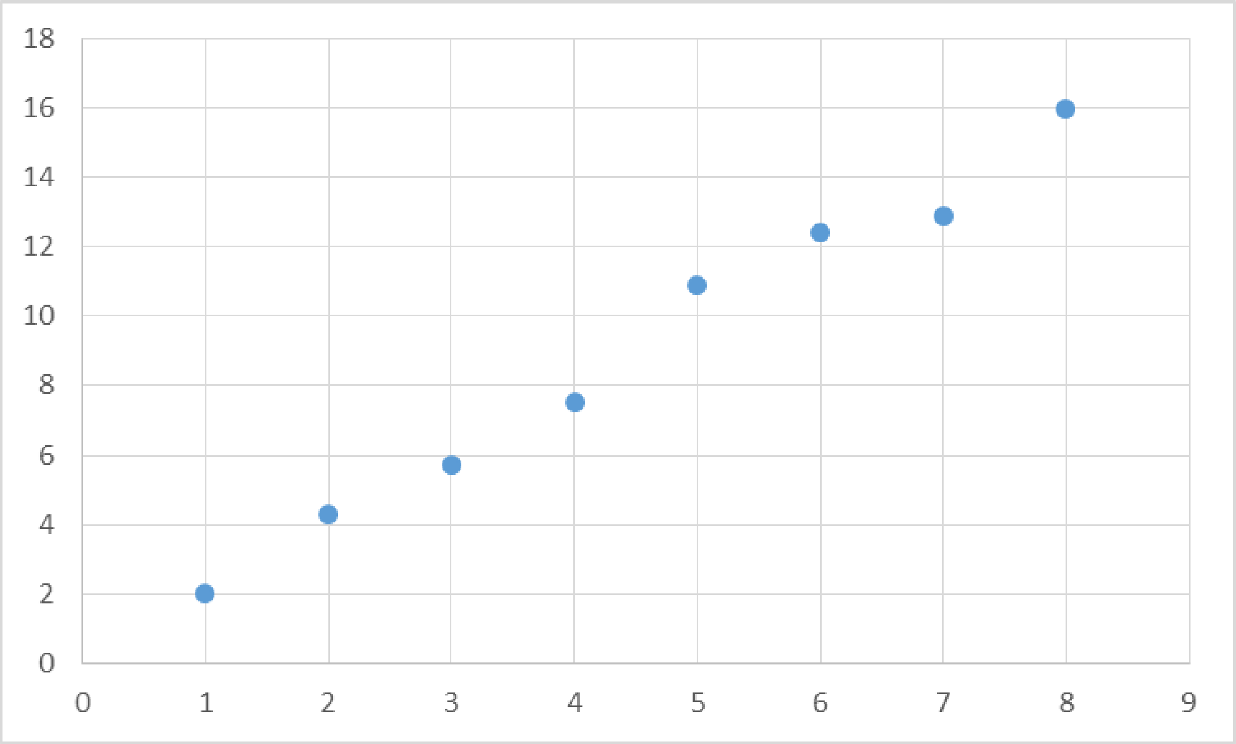
- Input: Training set or samples
  - Each sample has 1 input value x, and 1 output value y
- Output: A mathematical equation that, given an input, generates the expected/predicted output
  - $y = \theta * x$
  - Referred to as the "Hypothesis": $h_\theta(x) = y = \theta * x$

Linear Regression

- Hypothesis: $h_\theta(x) = y = \theta * x$
  - x = Input value
  - y = Output value
  - θ = Weight/Parameter
- The learning algorithm will learn θ (Theta)
  - Weight of x

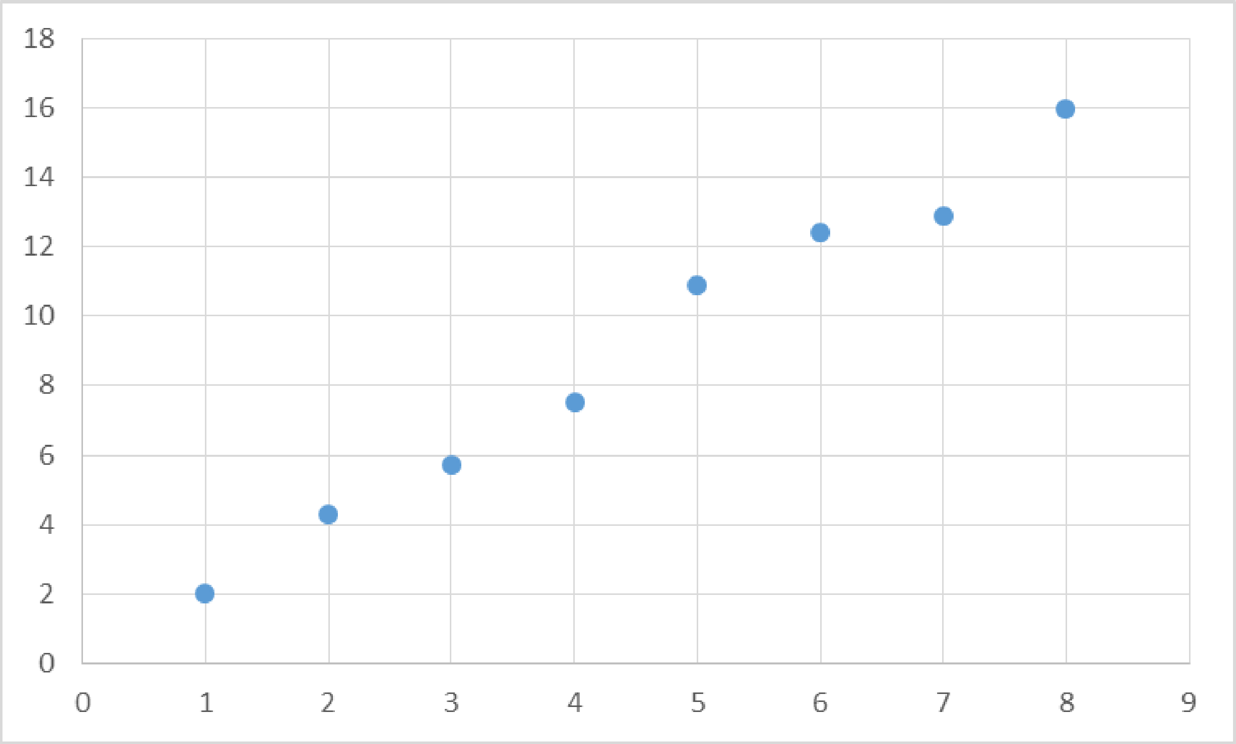
Linear Regression

- What is the value of θ?
  - $\theta \approx 2$



Linear Regression

- Linear regression will keep adjusting the value of θ
  - Until it's close to 2
  - Which represents y = 2x
- Mathematical equation won't go through all the points
  - They're not all the outcome of y = 2x
  - How to quantify how "correct" it is?
    - Cost Function



Cost

- The cost of a hypothesis is used to track how off it is
- The lower the cost, the more accurate it is
- Referred to as J(θ)
- Least Squares Cost equation:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$$

m: # of samples

i: Index of sample

Note: the $\frac{1}{2m}$ normalization might not always be there if your check for Least Squares in other sources (it actually has no effect, we just get half the cost)

Cost

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2$$

- Square: Because error could be positive or negative
- Divide by m: To determine the average of the cost

Cost

For the following input data, what is the cost with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2 y: 5.8
x_0: 8 y: 14.7
x_0: 12 y: 24.3
x_0: 20 y: 41.1

θ = 1 → $h_1(x) = y = x$

$$J(1) = \frac{1}{8}\sum_{i=1}^{4}\left(h_1\left(x^i\right) - y^i\right)^2$$

$$J(1) = \frac{1}{8}\left[(2 - 5.8)^2 + (8 - 14.7)^2 + (12 - 24.3)^2 + (20 - 41.1)^2\right] \approx 81.98$$

Cost

For the following input data, what is the cost with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2 y: 5.8
x_0: 8 y: 14.7
x_0: 12 y: 24.3
x_0: 20 y: 41.1

θ = 2 → $h_2(x) = y = 2x$

$$J(2) = \frac{1}{8}\sum_{i=1}^{4}(h_2(x^i) - y^i)^2$$

$$J(2) = \frac{1}{8}[(4 - 5.8)^2 + (16 - 14.7)^2 + (24 - 24.3)^2 + (40 - 41.1)^2] \approx 0.78$$

Cost

For the following input data, what is the cost with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2	y: 5.8
x_0: 8	y: 14.7
x_0: 12	y: 24.3
x_0: 20	y: 41.1

θ = 3 → $h_2(x) = y = 3x$

$$J(3) = \frac{1}{8}\sum_{i=1}^{4}(h_3(x^i) - y^i)^2$$

$$J(3) = \frac{1}{8}[(6 - 5.8)^2 + (24 - 14.7)^2 + (36 - 24.3)^2 + (60 - 41.1)^2] \approx 72.58$$

Minimizing Cost

- How to minimize the cost J(θ)?
  - The samples are constant, the only thing we can adjust is θ
    - Reminder: The learning algorithm is trying to learn θ
  - θ starts at a small random value
  - θ will keep adjusting until the cost reaches its minimum value
  - Minimum value is not guaranteed, at least with multivariable equations
    - Local vs global minimum
  - The value of θ will adjust using the **gradient descent** algorithm
  - Process so far: Adjust θ, check cost, adjust θ, check cost, and so on

Gradient Descent

- Observe the slope of the cost curve
  - If the slope is positive, θ needs to be decreased
  - If the slope is negative, θ needs to be increased
  - If the slope is 0:
    - A local or global minimum has been reached
    - θ shouldn't (and won't) change anymore

Gradient Descent

- θ is adjusted by the negative of the cost equation J(θ) derivative
- Decrease θ by the "Cost Derivative"
  - $\theta \ = \ \theta - Cost\ Derivative$

Cost Derivative

$$h_\theta(x) = y = \theta x$$

Cost Derivative

$$= \frac{\partial}{\partial_\theta} J(\theta)$$

$$= \frac{\partial}{\partial_\theta} [\frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2]$$

$$= \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i).x^i$$

Cost Derivative

For the following input data, what is the cost derivative with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2	y: 5.8
x_0: 8	y: 14.7
x_0: 12	y: 24.3
x_0: 20	y: 41.1

<u>θ = 1</u> → $h_1(x) = y = x$

$$\frac{\partial}{\partial_\theta} J(1) = \frac{1}{4}\sum_{i=1}^{4}\left(h_1\left(x^i\right) - y^i\right).x^i$$

$$\frac{\partial}{\partial_\theta} J(1) = \frac{1}{4}[(2 - 5.8) \cdot 2 + (8 - 14.7) \cdot 8 + (12 - 24.3) \cdot 12 + (20 - 41.1) \cdot 20] \approx -157.7$$

Cost Derivative

For the following input data, what is the cost derivative with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2	y: 5.8
x_0: 8	y: 14.7
x_0: 12	y: 24.3
x_0: 20	y: 41.1

θ = 2 → $h_2(x) = y = 2x$

$$\frac{\partial}{\partial_\theta} J(2) = \frac{1}{4}\sum_{i=1}^{4}\left(h_2\left(x^i\right) - y^i\right).x^i$$

$$\frac{\partial}{\partial_\theta} J(2) = \frac{1}{4}[(4 - 5.8)\cdot 2 + (16 - 14.7)\cdot 8 + (24 - 24.3)\cdot 12 + (40 - 41.1)\cdot 20] \approx -9.4$$

Cost Derivative

For the following input data, what is the cost derivative with θ = 1, θ = 2 and θ = 3 for hypothesis $h_\theta(x) = y = \theta * x$?

x_0: 2	y: 5.8
x_0: 8	y: 14.7
x_0: 12	y: 24.3
x_0: 20	y: 41.1

<u>θ = 3</u> → $h_3(x) = y = 3x$

$$\frac{\partial}{\partial_\theta} J(3) = \frac{1}{4}\sum_{i=1}^{4}\left(h_3\left(x^i\right) - y^i\right).x^i$$

$$\frac{\partial}{\partial_\theta} J(3) = \frac{1}{4}\left[(6 - 5.8)\cdot 2 + (24 - 14.7)\cdot 8 + (36 - 24.3)\cdot 12 + (60 - 41.1)\cdot 20\right] \approx 444.9$$

Gradient Descent

- The change in θ's value could be too fast or too slow
- Adjust θ by - "Learning Rate" * "Cost Derivative"
  - α = Learning Rate
  - $\theta = \theta - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^i\right) - y^i\right). x^i$
- Repeat until convergence

Learning Rate

- Determines how fast θ converges to the minimum
- A smaller α leads to a smaller θ change per iteration
  - This might lead to the gradient descent being very slow
- A larger α leads to a larger θ change per iteration
  - This might lead a cost increase after each iteration
  - Overshoot the minimum

Learning Rate

- Should α be decreased overtime?
  - As the cost approaches a local/global minimum, smaller reductions are needed
  - As the derivative approaches 0, smaller θ changes are needed
    - We're very close to the minimum!
  - It seems intuitive to decrease α over time
  - But it should NOT
  - As J(θ) approaches a minimum, its derivative becomes smaller
    - (It's 0 at the minimum)
  - The steps will automatically become smaller

Gradient Descent Algorithm

- Set θ to a random small value
- Adjust θ by – α*Cost Derivative
- Repeat, or break if the max # of iterations has been reached or the algorithm converged

Multivariable Linear Regression

- When the input (x) is 0 is the output (y) necessarily 0?
  - NO
- Also referred to as multiple linear regression
- Consequently, the fitted curve won't go through the origin (0,0)
  - $y = ax+b$
- Add θ_0 as a constant
  - $y = \theta_0 + \theta_1 x_1$
    - As if $x_0 = 1$
  - Referred to as the "intercept"

Cost Derivative

$$h_\theta(x) = y = \theta_0 + \theta_1 x$$

Cost Derivative

$$= \frac{\partial}{\partial_{\theta_0}} J(\theta)$$

$$= \frac{\partial}{\partial_{\theta_0}} [\frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2]$$

$$= \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)$$

Multivariable Linear Regression

- Now we have multiple θ's
- Gradient descent should adjust them all **simultaneously**
- For each feature θ_j:
  - $\theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^i\right) - y^i\right). x_j^i$
  - i: Sample index
  - j: Feature index
- Except for θ_0 when it is an intercept

Multivariable Linear Regression

- Updating θ_0 requires the original values of all θ's
- Updating θ_1 requires the original values of all θ's
- Updating θ_2 requires the original values of all θ's
- …

- Compute the new values of all θ's, then update them

Multivariable Linear Regression

- Interpretation of having multiple features
  - More than 1 feature contribute to the final output
  - Identify the relationship of any single feature (x_1, x_2, etc.) and the output, when all other features are held fixed
    - The unique effect of x_i on y
    - Assuming features are not correlated with one another

Multivariable Linear Regression

- Same feature used multiple times, raised to a different power each time
  - $h_\theta(x) = y = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 \quad (x_0 = 1)$
- Multiple features, multiple powers
  - $h_\theta(x) = y = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_2 \; (x_0 = 1)$
- General cost function becomes:
  - $J(\theta) = J(\theta_1, \theta_2, \dots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$
  - m: # of samples, n: # of features

Multivariable Regression

- Interpretation of having multiple features that might not be linear.
  - Identify the relationship of any single feature (x_1, x_2, etc.) and the output, some feature might be related to quadratic, cubic
  - Example: The best hypothesis might look like the following

$$h_\theta(x) = y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 {x_3}^3 + \theta_4 {x_4}^2$$

  - Is anything different for features x_3 and x_4?
    - Cost derivative function

Cost Derivative

$$h_\theta(x) = y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 {x_3}^3 + \theta_4 {x_4}^2$$

Cost Derivative

$$= \frac{\partial}{\partial_{\theta_3}} J(\theta)$$

$$= \frac{\partial}{\partial_{\theta_3}} [\frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2]$$

$$= \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i) \cdot {x_3}^3$$

Multivariable Regression Algorithm

- Pick the features (x_1, x_2, etc.)
- Pick a value for the learning rate α
- Set the equation's complexity (the power for each feature)
  - Example: $y = h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$, $y = h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2^2$
- For each θ_j, where $0 < j < n$:
  - Compute the derivative of the cost function with respect to θ_j
  - Compute θ_j's new value: – α*Cost Derivative
    - Don't update θ_j's value yet
    - Save to a temporary location
- For each θ_j :
  - Update θ_j's value from the temporary location

Vector Representation

- Each hypothesis can be represented as a vector multiplication
  - $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$
- Assume θ and x are vectors: $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$ and $x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$
- $h_\theta(x) = \theta^T x = [\theta_0 \quad \theta_1 \quad \theta_2 \quad \theta_3] * \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$

Hints

- Choosing a biggest converging learning rate will help getting to the result fast (when debugging)
- What if the algorithm converges to a local minima?
  - No way out from there, but it might be a good enough model
- What if it is not?
  - Change to different initial θ values
- How do I choose correct initial θ values?
  - Test out different initial values and inspect the different convergence values
- Can I get the global minima?
  - Depends on the complexity of your equation, single variable linear regression always converges to global minima, might not if we use multivariable regressions (start from different initial θ values to find different solutions)

Assignment #1

Regression

References

- Notes by Antoine Abi Chacra, DigiPen Institute of Technology