Machine Learning

Lecture 2

Participation and Assignment

- Pilot News Updates
- Pop quizzes (NAME)
- Assignment 1 will be on linear regression (today’s class)

Categories Within Supervised ML

Classification machine learning systems: Systems where we seek a yes-or-no prediction, such as “Is this tumer cancerous?”, “Does this cookie meet our quality standards?”

a. Binary or Multiclass classifier : Output y in {-1, 1} or y in {1,..k}

Regression machine learning systems: Systems where the value being predicted falls somewhere on a continuous spectrum. These systems help us with questions of “How much?” or “How many?”.

Example

Independent variable? Target variable?

Example

Independent variable? Target variable?

Some Terms in Supervised Learning

Training Data

Data used to “learn” the relationship between the independent and target variables

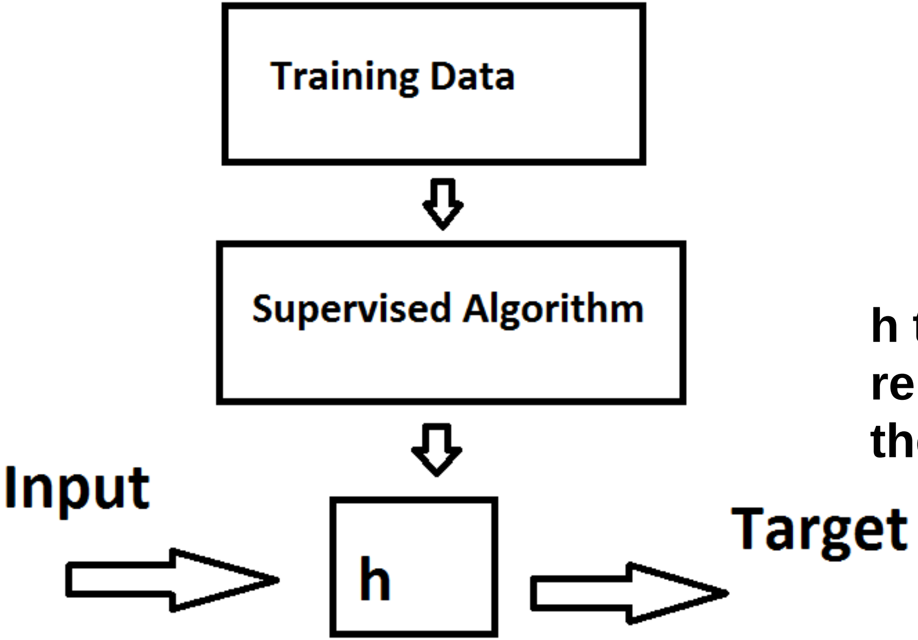
“gold standard”, contains labels.

Some Terms in Supervised Learning

Test Data

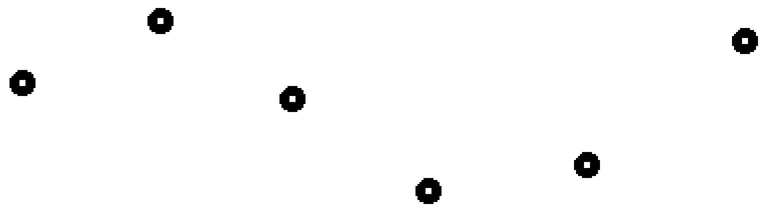
Estimate the accuracy using the unseen data (validate using some performance metric P)

Hypothesis Function



h tries to map the relationship between the input and target

Regression



Regression

How do we choose the “right function”?

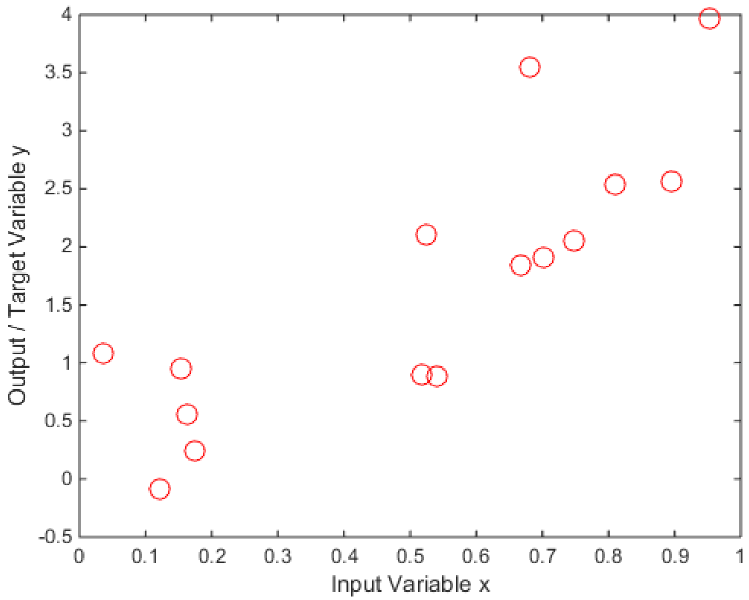
How do we measure the “rightness”?

How do we trade off between the degree of fit and the complexity of solution?

Linear Regression:drawing

Notation:

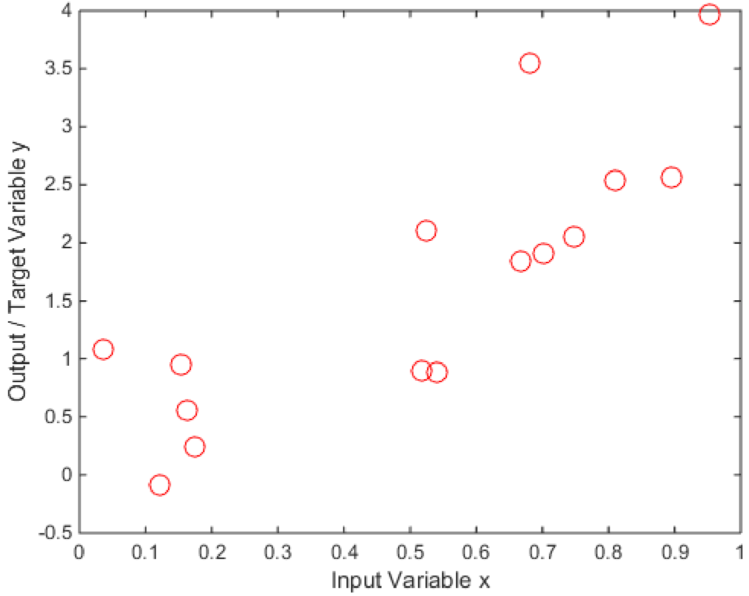
m: Number of Training Samples
x: input variables
y: output variables



Linear Regression:drawing

m: Number of Training Samples
x: input variables
y: output variables

Training data points: (x,y)
"ith" data point: $(x^{(i)},y^{(i)})$

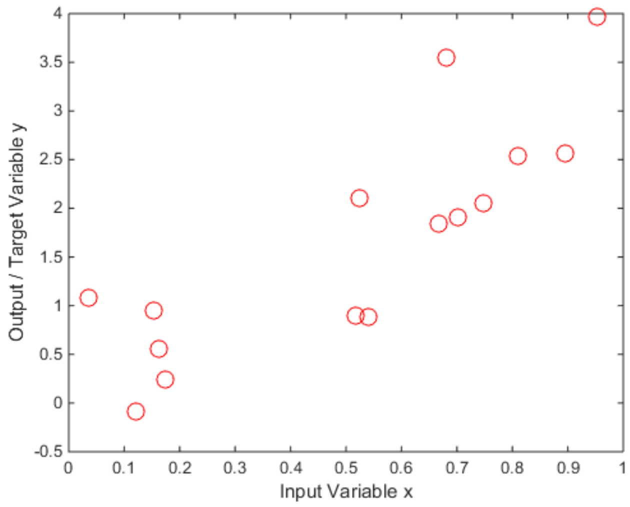


Linear Regression: single variable

Consider the single independent variable case:

y = hθ(x) s. t.

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



Cost Function

Noise is present -- problem of real data
What are the unknowns here?

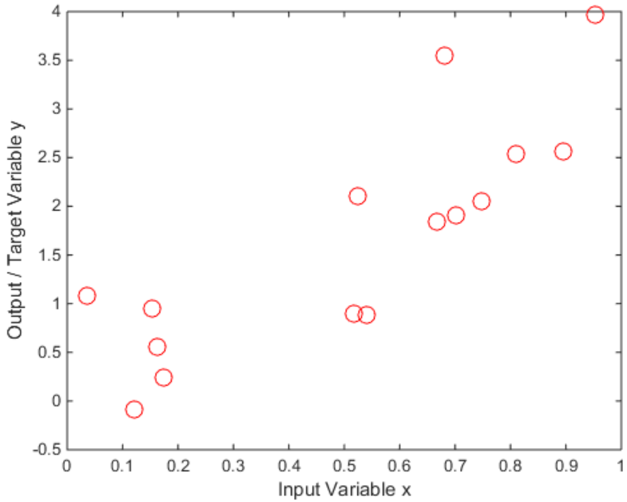
$$h_\theta(x) = \theta_0 + \theta_1 x$$

Cost Function

Noise is present -- problem of real data

What are the unknowns here?

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



Cost Function

What is the goal here?

Choose: θ_0 & θ_1 s.t. $h_\theta(x)$ is close to y for the given training data

How do we write this up??

Cost Function

What are we trying to minimize?
What are the knowns? What are the unknowns?

Cost Function

Find the values of Θ_0 & Θ_1 to minimize this expression:

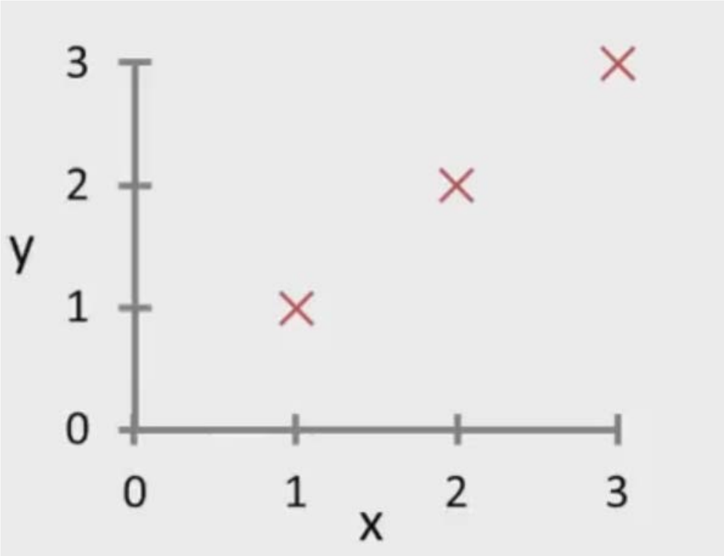
$$\text{Cost Function:} \quad J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

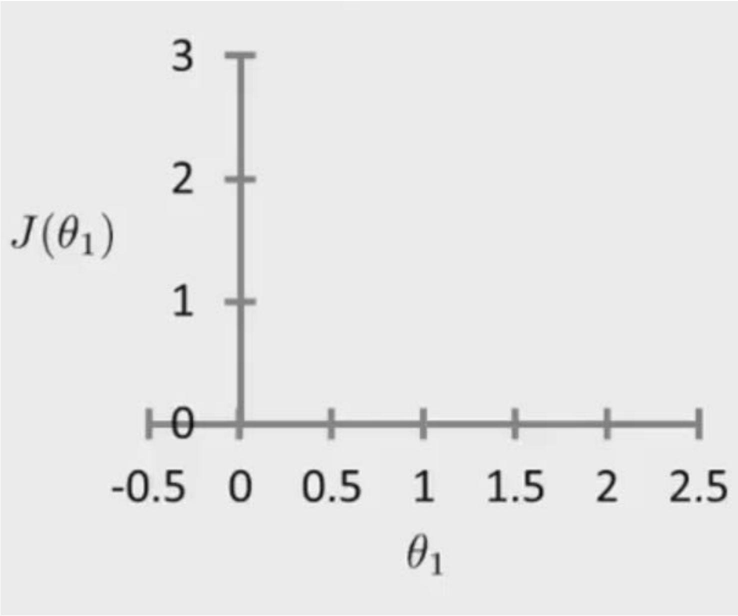
Cost Function/ Squared Error

Find the values of Θ_0 & Θ_1 to minimize this expression:

Cost Function: $$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

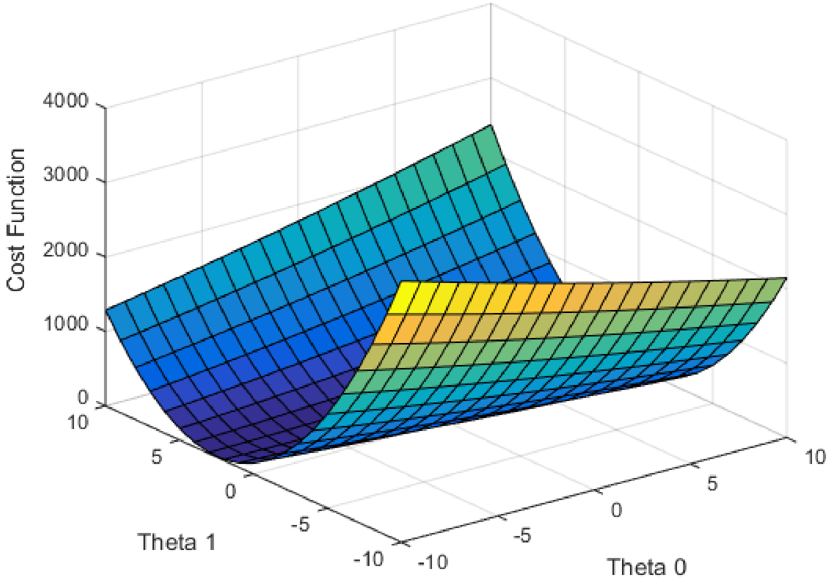
Cost Function: Understanding J(ϴ)





Cost Function: $J(\theta_0, \theta_1)$?

Cost Function



Optimization

- We have created a cost function that we want to minimize over the training data samples
- We want to experiment with different values of θ_0 & θ_1 so that the cost function $J(\theta_0,\theta_1)$ keeps reducing so we can end up in the minimum (hopefully)

Gradient Descent (drawing)

Start with initial values. Keep changing theta values till we reach the minima.

Gradient Descent

α is called the Learning Rate (some books also use ***η***), ∊ [0,1]

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

Gradient Descent

α is called the Learning Rate, ∊ [0,1]

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

What is the global minimum?

What happens to the 2nd term once we reach there?

Gradient Descent

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

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

Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$

$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

Effect of α

- Too small
- Too large

Gradient Descent

Repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

}

Some Pointers On Implementation:

- Using temp variables (2nd parameter is a function of Θ_0 and Θ_1)
- Updating the parameters at the end simultaneously, after computing the partial derivatives for each parameter
- Values of α : varying from 0 to 1

Multivariate Gradient Descent

Hypothesis: $h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

Parameters: $\theta_0, \theta_1, \ldots, \theta_n$

Cost function:

$$J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Note: Both x and theta are now (n+1) dimensional

Gradient Descent

For m training samples, then we get:

$$
\begin{aligned}
&\text{Repeat until convergence } \{ \\
&\qquad \theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \qquad \text{(for every } j\text{).} \\
&\}
\end{aligned}
$$

Gradient Descent

Batch vs Online

Batch: considers all the training samples in the data (or batches at a time)

Online: considers each training sample one at a time

Q: Is our method online or batch?

Feature Scaling

Are there any challenges here that could affect our optimization function (cost function)?

Feature 1	Feature 2	Target
0	4100	255
1	6544	422
2	7711	122
1	100	661

Feature Scaling

Replace feature x with (x - mean) / (max - min)

Feature 1	Feature 2	Target
0	4100	255
1	6544	422
2	7711	122
1	100	661

How to verify that the Algorithm is working as it should?

What should the cost function look like over time? (What is time here??)

Stopping Criteria

- This is an iterative algorithm
- How do we know when to stop?
  - choose a small threshold ε: if the change in cost function is below ε, stop the iterations
  - Hard code the number of iterations
  - A combination of both (look at the graph first)

Polynomial Features

Suppose I have this function:

$h_\theta(x) = \Theta_0 + \Theta_1 x + \Theta_2 x^2$

Does this change the algorithm we have learned so far?