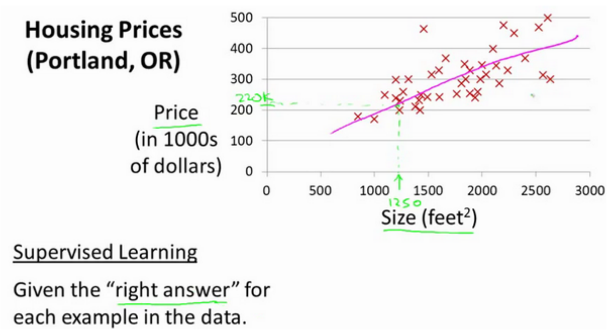
Andrew Ng Course Note 2

Shu Wang

May 12 2023

1 Linear Regression

Example: Predict price of house based on size of house.



Training a model by giving a data that has the right answers. Data with price and square feet given.

Regression model predicts numbers.

Any supervised model that predicts a number is a regression model.

In contrast with regression, another type of supervised machine learning is classification. Classification model predicts categories or discrete categories.

Classification: only a small number of possible outputs.Discrete,finite set of possible outputs.

A dataset that is used to train the model is called a training set.

Notation:

x = 'input' variable, also called 'feature'

y = 'output' variable, also called 'target'

In this housing example, x = house square feet, y = house price.

m = number of training samples

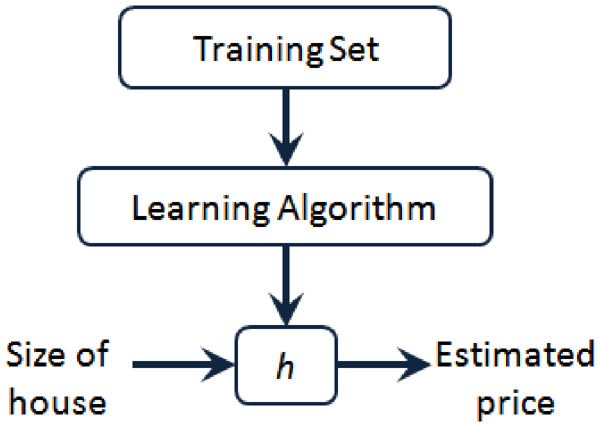
(x, y) = a single training example

$(x^{(i)}, y^{(i)})$ is the i^{th} training example

i is the index of the sample.

$\hat{y}$ is the estimate or prediction of y.

Function f is called the model, x is called the input/input feature.



y refers to the target,the actual true value in the training set.

In contrast, $\hat{y}$ is an estimate,it may or may not be the true value.

How to represent f?

$f_{w,b}(x) = wx + b$

Value of w, b will determine value of $\hat{y}$.

Simply,we write $f(x)$.

The algorithm generates a linear line.

Sometimes you want to fit more complex functions(e.g, second degree polynomial). But since linear function is easier to work with, let's use a line as a fundation that will eventually get to more complex models.

This model is called linear regression. Specifically, linear regression with one variable. The term 'one variable' refers to a single input variable of feature x. It is also called univariate linear regression.

2 Cost Function

Define a cost function. Cost function will tell us how well the model is doing.

Model: $f_{w,b}(x) = wx + b$

w, b are called parameters of the model.

In Machine Learning, w, b are variables that you can adjust during training to approve the model. It is also called coefficients or weights.

value of w gives you slope, b gives you intercept.

Choose the line that fits the data well, roughly passing through the training examples.

For a given data point $(x^{(i)}, y^{(i)})$, the model also gives a prediction $\hat{y}^{(i)}$.

Equation: $\hat{y}^{(i)} = f_{w,b}(x^{(i)})$

$f_{w,b}(x^{(i)}) = wx^{(i)} + b$

Task:

Find w, b

s.t $\hat{y}^{(i)}$ is close to $y^{(i)}$ for all $(x^{(i)}, y^{(i)})$

We construct a cost function.

Error for a single term: $(\hat{y}^{(i)} - y^{(i)})^2$

Sum of Squared error:

$$\sum_{i=1}^{m} (\hat{y}^{(i)} - y^{(i)})^2$$

where m = number of training examples.

Average squared error: $\frac{1}{m} \sum_{i=1}^{m} (\hat{y}^{(i)} - y^{(i)})^2$

We normally divided by $2m$, as this would make it easier for differentiation.

$J(w, b) = \frac{1}{m} \sum_{i=1}^{m} (\hat{y}^{(i)} - y^{(i)})^2$

The squared error cost function is the most commonly used one for linear regression. Give good results for all regression.

$J(w, b) = \frac{1}{2m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)})^2$

Model: $f_{w,b}(x) = wx + b$

Parameters: w, b

Cost function:

$J(w, b) = \frac{1}{2m} \sum_{i=1}^{m} (f_{w,b}(x^{(i)}) - y^{(i)})^2$

Goal:

$\underset{w,b}{\text{minimize}}\, J(w, b)$

Now let's assume we only have one parameter, w.

Simplified:

$f_w(x) = wx$

Cost function:

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

Goal:

$\underset{w}{\text{minimize}}\, J(w)$

Comparison between $f_w(x) and J(w)$:

for fixed w, function of x.

Take $w = 1$.Take three points $(1, 1), (2, 2), (3, 3)$.

$J(w) = \frac{1}{2m} \sum_{i=1}^{m} (f_w(x^{(i)}) - y^{(i)})^2 = \frac{1}{2m} \sum_{i=1}^{m} (w * (x^{(i)}) - y^{(i)})^2 = \frac{1}{2m}(0^2 + 0^2 + 0^2)$.

Now you can plot $J(w)$, which is a function of w.

And we have $J(1) = 0$.

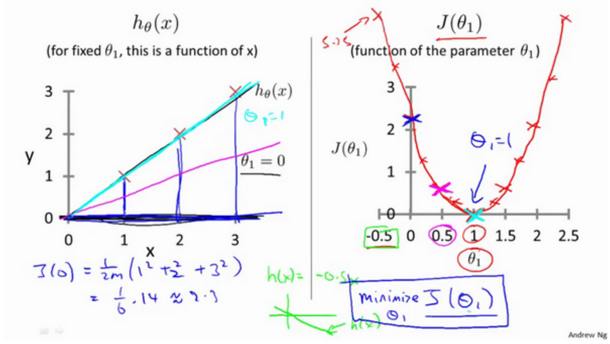
Similarly,we can compute $J(w)$ when $w = 0.5$.

$J(0.5) = \frac{1}{2m} * (0.5^2 + 1^2 + 1.5^2) = \frac{7}{12}$

$J(0) = \frac{1}{2m} * (1^2 + 2^2 + 3^2) = \frac{7}{3}$

For each value of w, you can calculate the value $J(w)$.

Choosing a value of w that causes $J(w)$ to be as small as possible will give us a good model. In this case, picking $w = 1$ will result in $J(w) = 0$.



Goal of linear regression:

$\underset{w}{\text{minimize}}\, J(w)$

General case:

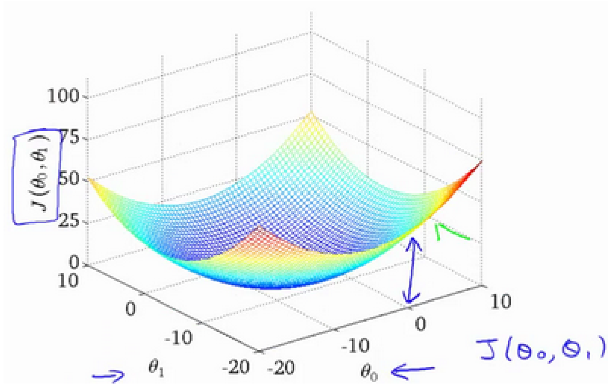
$\underset{w,b}{\text{minimize}}\, J(w,b)$

The original model with two parameters:

$f_{w,b}(x) = 0.06x + 50$

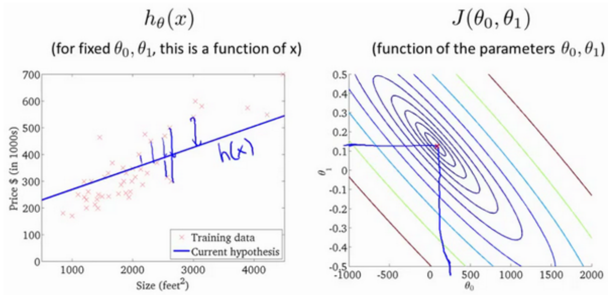
U-shaped curve when only one parameter involved.

If we have two parameters, the cost function would have a shape of a soup bowl.



Any single points in this surface represents some particular choice of w and b.

We could also use a contour plot.



Take horizontal slices of the 3D plot.

Take a particular combination $(w, b) = (-0.15, 800)$. This combination is not fit the training set well.

Another choice: $(w, b) = (0, 360)$. This is a horizontal line.

Another choice: $(w, b) = (0.13, 71)$. Sum of squared error is pretty close to the minimum sum of squared errors.

Automatically finding the values of w, b that gives you the best fit line that minimizes the cost function.

The algorithm is called: gradient descent and variation on gradient descent algorithm.