Machine Learning for Engineering and Science Applications

Linear Regression

Least Squares

Gradient Descent

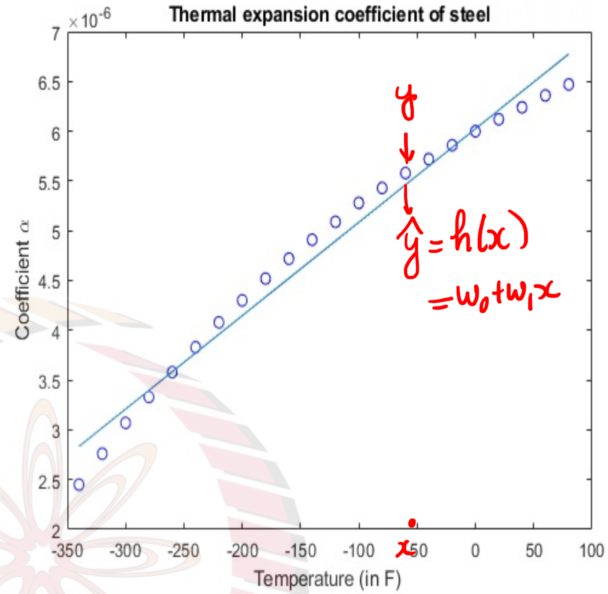
Regression example

Our example was as follows

Table 1 Coefficient of thermal expansion versus temperature for steel.

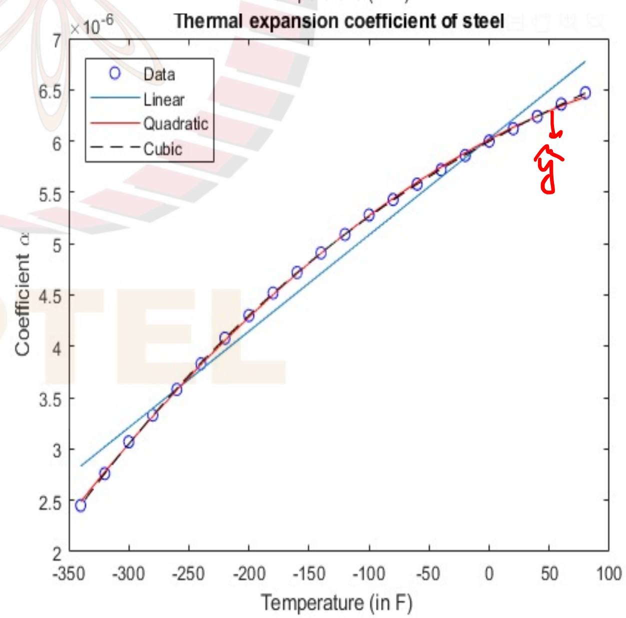
Temperature, T	Coefficient of thermal expansion, α
°F	in/in°F
80	6.470×10^{-6}
60	6.360×10^{-6}
40	6.240×10^{-6}
20	6.120×10^{-6}
0	6.000×10^{-6}
−20	5.860×10^{-6}
−40	5.720×10^{-6}
−60	5.580×10^{-6}
−80	5.430×10^{-6}
−100	5.280×10^{-6}
−120	5.090×10^{-6}
−140	4.910×10^{-6}
−160	4.720×10^{-6}
−180	4.520×10^{-6}
−200	4.300×10^{-6}
−220	4.080×10^{-6}
−240	3.830×10^{-6}
−260	3.580×10^{-6}
−280	3.330×10^{-6}
−300	3.070×10^{-6}
−320	2.760×10^{-6}
−340	2.450×10^{-6}

x $\quad$ y (Ground Truth)



$\hat{y} = h(x)$

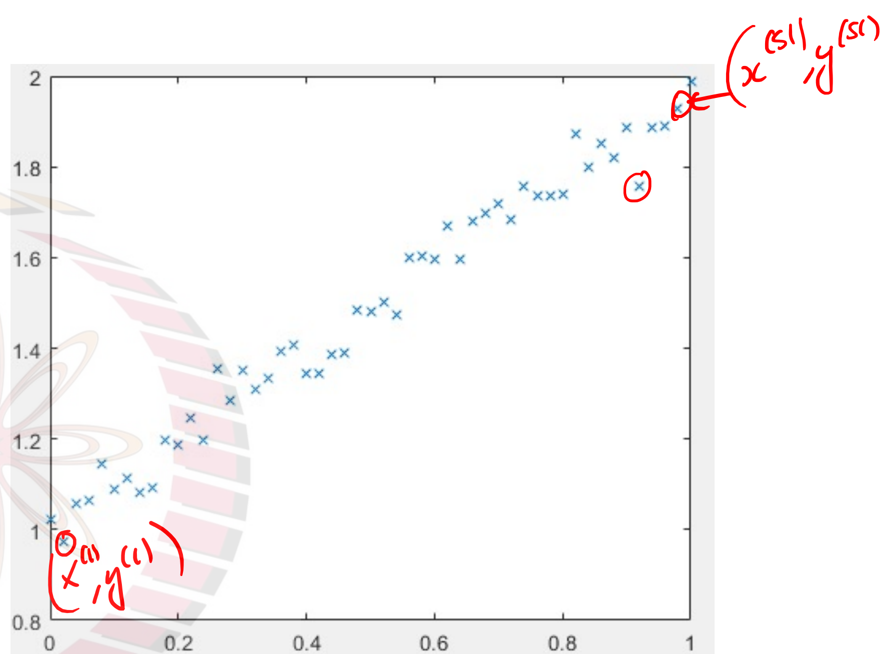
$w_0 + w_1 x + w_2 x^2$



We will discuss how to come up with these in the coming video(s)

The general univariate linear regression problem

Input (x)	Output (y)
$x^{(1)}$	$y^{(1)}$
$x^{(2)}$	$y^{(2)}$
…	…
$x^{(m)}$	$y^{(m)}$



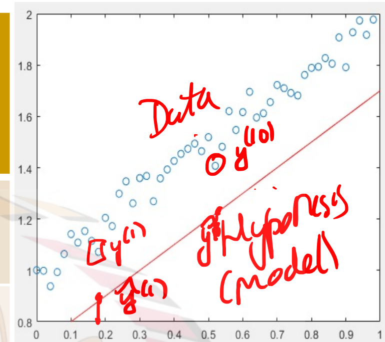
We start with the case where there is a single input and a single output

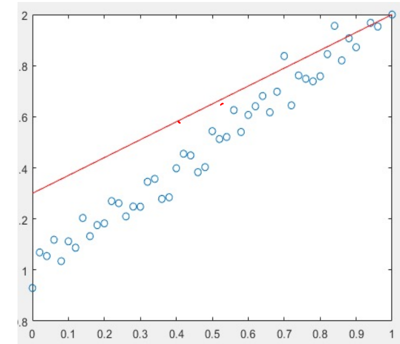
$(x^{(i)}, y^{(i)})$: i^{th} "example" of (input,output) set

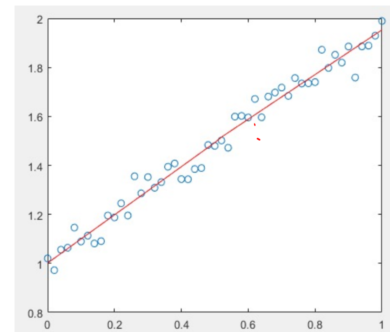
$\boxed{m}$: Number of examples or data points

The general univariate linear regression problem

Input (x)	Output (y) Given Data	Model ($\hat{y}$)
$x^{(1)}$	$y^{(1)}$	$\boxed{\hat{y}^{(1)}}$
$x^{(2)}$	$y^{(2)}$	$\hat{y}^{(2)}$
…	…	…
$x^{(m)}$	$y^{(m)}$	$\hat{y}^{(m)}$







Quantify

We now introduce our first model hypothesis – Linear Model

$$\boxed{\hat{y} = h(x) = w_0 + w_1 x}$$ ← Form

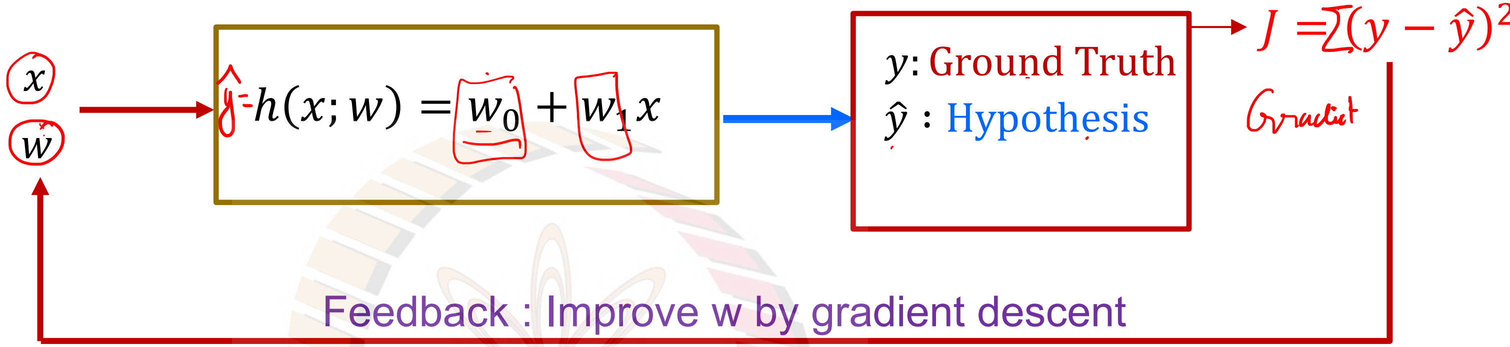
There are infinite w_0, w_1 possibilities. Which do we choose?

For this, we define a cost function $\boxed{J = \frac{1}{2m} \sum_{i}^{m} \left(y^{(i)} - \hat{y}^{(i)}\right)^2}$ Least Squares

Optimal $\boldsymbol{w}$ is the one that minimizes the above cost function

Called the least squares fit. The cost function is called least mean square (LMS)

Finding the linear regression coefficients Using gradient descent



We have m data points $\left(x^{(i)}, y^{(i)}\right)$ $\quad i = 1,2, \dots, m$

Guess $\boldsymbol{w} = [w_0, w_1]$

For any given guess of $\boldsymbol{w}$, we have the corresponding output

$$\hat{y}^{(i)} = w_0 + w_1 x^{(i)}$$

Calculate $J = \frac{1}{2m} \sum_{i=1}^{m} \left(y^{(i)} - \underbrace{w_0 - w_1 x^{(i)}}_{\hat{y}^{(i)}}\right)^2$

Improve $\boldsymbol{w}$ by using

$$\boldsymbol{w} = \boldsymbol{w} - \alpha \nabla_w J$$

Stop when the stopping criterion is met

The final set of $\boldsymbol{w} = [w_0, w_1]$ obtained are the regression coeffs.

Input (x)	Output (y)	Model ($\hat{y}$)
$x^{(1)}$	$y^{(1)}$	$\hat{y}^{(1)}$
$x^{(2)}$	$y^{(2)}$	$\hat{y}^{(2)}$
…	…	…
$x^{(m)}$	$y^{(m)}$	$\hat{y}^{(m)}$

But how do we calculate $\nabla_w J$?

Calculating the least squares gradient

$[w_0, w_1]^T$

$\left[\nabla_{w_0} J, \frac{\partial J}{\partial w_1}\right]$

$\frac{\partial J}{\partial w_0}$

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

$\hat{y} = w_0 + w_1 x$

$$= \frac{1}{2m}\sum_{i=1}^{m}\left(y^{(i)} - w_0 - w_1 x^{(i)}\right)^2$$

$\hat{y}^{(i)} = w_0 + w_1 x^{(i)}$

compact notation

$\frac{\partial \hat{y}^{(i)}}{\partial w_1} = x^{(i)}$

$$\boldsymbol{w} = \boldsymbol{w} - \alpha \nabla_w J$$

$$\Rightarrow \; w_0 = w_0 - \alpha \frac{\partial J}{\partial w_0}$$

$$w_1 = w_1 - \alpha \frac{\partial J}{\partial w_1}$$

$x_0 \equiv x^0 = 1$

$x_1 \equiv x^1 = x$

$\hat{y} = w_0 + w_1 x_{(1)}$

$= w_0 x_j^0 + w_1 x^{(1)}$

x_0

Claim: $$\boxed{\frac{\partial J}{\partial w_j} = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)x_j^{(i)}}$$

$m = 2$

Proof $\frac{\partial J}{\partial w_0} = \frac{1}{2m}\frac{\partial}{\partial w_0}\left[\left(y^{(1)} - w_0 - w_1 x^{(1)}\right)^2 + \left(y^{(2)} - w_0 - w_1 x^{(2)}\right)^2\right] = \frac{1}{2m}\left[2\left(y^{(1)} - w_0 - w_1 x^{(1)}\right)(-1) + 2\left(y^{(2)} - w_0 - w_1 x^{(2)}\right)(-1)\right]$

$= -\frac{1}{m}\left[\left(y^{(1)} - \hat{y}^{(1)}\right) + \left(y^{(2)} - \hat{y}^{(2)}\right)\right]$

$\Rightarrow \frac{\partial J}{\partial w_0} = -\frac{1}{m}\sum_{i=1}^{m}\underbrace{\left(y^{(i)} - \hat{y}^{(i)}\right)}_{\text{Error}}$

$$\frac{\partial J}{\partial w_1} = \frac{1}{2m}\frac{\partial}{\partial w_1}\left\{\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2\right\}$$

$$= \frac{1}{2m}\sum_{i=1}^{m} 2\left(y^{(i)} - \hat{y}^{(i)}\right)\left[-\frac{\partial \hat{y}^{(i)}}{\partial w_1}\right]$$

$$= -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)\frac{\partial \hat{y}^{(i)}}{\partial w_1} = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)x^{(i)}$$

Steps of the linear regression procedure

1. Decide on α, ϵ and stopping criterion

2. Make an initial guess for the weight vector $\boldsymbol{w} = \boldsymbol{w}^{(0)}$

3. Calculate $\boldsymbol{w}^{(k+1)} = \boldsymbol{w}^{(k)} - \frac{\alpha}{2m}\sum_i \left(y^{(i)} - \hat{y}^{(i)}\right)\boldsymbol{x}^{(i)}$

4. Calculate stopping criterion
    1. If condition satisfied, stop
    2. If not satisfied, go to Step 3

Let us now see a code to implement this