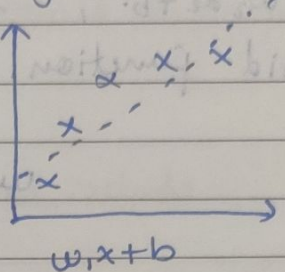
11. The Problem of Overfitting.

a. Overfitting and Underfitting.

Regression example:

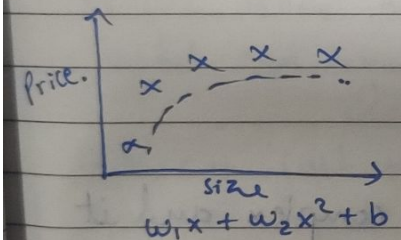


→ linear fit.

Does not fit the training set well.

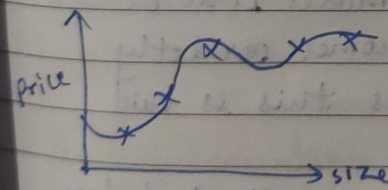
The model is underfit.
The model has high bias.

Underfit - Not able to fit the data.
high bias - Pre conception that the data is linear.



- fits training set pretty well.
Just right.

generalization - We want our model to be generalized well.
It should perform well on examples that it has never seen before.



fits the training set extremely well.

Overfit the data - fits training data too well.
Will not generalize and perform well on unseen examples.

It has high variance. (same as overfit).
Trying very hard to fit every training example.

A well generalized model (just right) is a good model.

· Classification Problem.

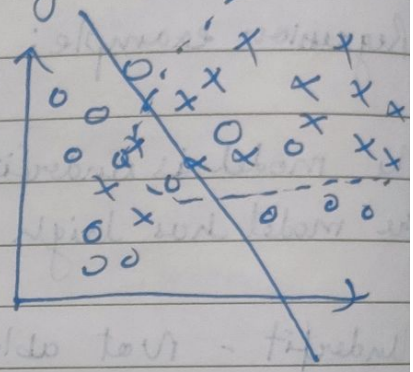
four example, if we have our mode as $z = w_1 x_1 + w_2 x_2 + b$. , $f_{\vec{w},b}(\vec{x}) = g(z)$.

g is sigmoid function.

for.

Then. it is underfit.

underfit example.



if, $z = w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2 + b$.

Then, it is just right.

if $z = w_1 x_1 + w_2 x_2 + w_3 x_1^2 x_2 + \dots + b$.

Then, the decision boundary will be complex and it will overfit.

c. Question. Our goal when creating a model is to be able to use the model to predict outcomes correctly for two examples. A model which does this is said to generalize well.

When a model fits the training data well but does not work well with new examples that are not in the training set, this is an example of:

a) Underfitting (high bias) b) Overfitting (high variance)
c) A model that generalizes well.

Answer- b.

II. Addressing Overfitting.

a. Collect more traing examples.

If we get more data, the model will fit a f^n that is less wiggly.

b. Select features to include / exclude.

eg. size, bedrooms, floors, age, avg. income, ..., distance to coffee shop, price.(y).

We all → features.

If we have all features and insufficient data, our model can again overfit.

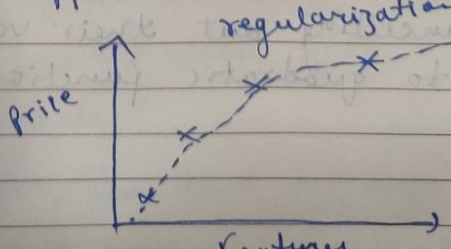
We can just select few features, like size, bedrooms, and age to get it just right.
This is called feature selection.

Dis advantage:
We are throwing away information.

c. Regularization.

Reduce the size of parameters w_j.

We had large values of w_j, we can eleminate a parameter to reduce it's effect.



If let's us keep features.

small values of w_j.

$f(x) = 13x - 0.23x^2 + 0.0000041x^3 - 0.00001x^4 + 10.$

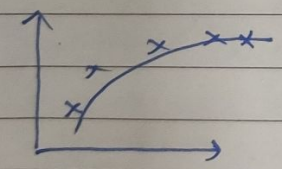
d. Summary: Addressing Overfitting.

Options- Collect more data
- Select features
- Reduce size of parameters using ~~Regularization~~ Regularization.

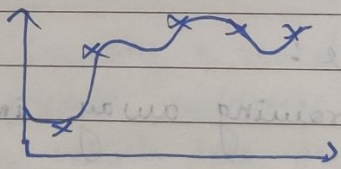
e. Question: Applying regularization, increasing the number of training examples, or selecting a subset of the most relevant features are methods for:
a) Addressing underfitting. b) —"— Overfitting.
Answer- b.

III. Cost function with Regularization.

a. Intuition.
we use the previous lecture.



$w_1x + w_2x^2 + b$



$w_1x + w_2x^2 + w_3x^3 + w_4x^4 + b$

make w_3, w_4 really small (≈ 0)

~~Choose~~ $\min_{\vec{w},b} \frac{1}{2m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + 1000\,w_3^2 + 1000\,w_4^2$

if we make them small

we minimize their cost. We are cancelling out their values. we make them closer to quadratic function.

b. Regularization.

Smaller values of $w_1, w_2 \dots w_n, b$, we have a simpler model which is less likely to ~~very~~ overfit.

More generally, if we have a lot of features, we ~~not~~ penalize all of our features.

say n features, we penalize $w_1, w_2, \dots, w_n$.

Our new cost function should look like this.

$$J(\vec{w}, b) = \frac{1}{2m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m} \sum_{j=1}^{n} w_j^2$$

λ = lamda = is called regularization parameter
$\lambda > 0$.

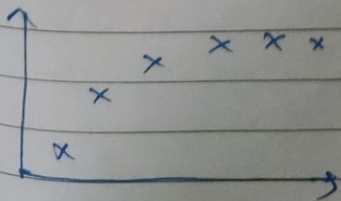
By scaling both terms by 2M, we have better way to calculate value of λ
We don't penalize b. We can include or exclude b.

c. Regularized Cost function.

$$\min_{w,b} J(\vec{w}, b) = \min_{w,b} \left[\frac{1}{2m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)^2 + \frac{\lambda}{2m} \sum_{j=1}^{n} w_j^2 \right]$$

fit data. mean squared error. regularized term.

We balance these two goals.



if $\lambda = 0$, we don't use regularization.

$f_{\vec{w},b}(\vec{x}) = w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4 + b$.

This will overfit.

if $\lambda = 10^{10}$, we choose $w \approx 0$, and $f(x) = b$.
This will underfit.

We need to choose just right λ.

d. Quiz: For a model that includes the regularization parameter λ (lambda), increasing λ will tend to.
a) Decrease $w_1, w_2 \dots w_n$ d) ~~Increase~~ decrease b.
b) Increase $w_1, w_2 \dots w_n$ c) Increase b
Answer: a).

IV. Regularized Linear Regression.

a. Regularized Linear Regression.

$$J(\vec{w}, b) = \frac{1}{2m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)^2 + \frac{\lambda}{2m} \sum_{j=1}^{n} w_j^2 \Big).$$

Gradient Descent: repeat { $w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w}, b)$
$j = 1, \dots, n.$
$b = b - \alpha \frac{\partial}{\partial b} J(\vec{w}, b)$ } simultaneous update.

Now, $\frac{\partial}{\partial w_j}$ was $\frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right) x_j^{(i)}$

and $\frac{\partial}{\partial b}$ was $\frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)$

With added regularized term, we will have $\frac{\lambda}{m} w_j$ for w_j. b remains the same.

b. Implementing gradient descent:

repeat { $w_j = w_j - \alpha \left[\frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(x^{(i)}) - y^{(i)} \right) x_j^{(i)} \right) + \frac{\lambda}{m} w_j \Big]$

$b = b - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right)$. } simultaneous update.

Now, $w_j = 1 \cdot w_j - \alpha \frac{\lambda}{m} w_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right) x_j^{(i)}$

$w_j \left(1 - \alpha \cdot \frac{\lambda}{m} \right)$. usual update.

Now, α is a small number, λ is a small number, (1 or 10). $\alpha (0.1)$, and m is 50.

Now, $\frac{\alpha \times \lambda}{m} = \frac{0.01 \times 1}{50} = 0.0002$.

Now $(1 - 0.0002) = 0.9998$.

we are multiplying it by a small number. We are restricting the value of w in every iteration.

c. How we get the derivative term($\frac{\partial}{\partial w_j} J(\vec{w}, b) = \frac{\partial}{\partial w_j} \left[\frac{1}{2m} \sum_{i=1}^{m} \left(f(\vec{x}^{(i)}) - y^{(i)} \right)^2 + \frac{\lambda}{2m} \sum_{j=1}^{n} w_j^2 \right]$.

$= \frac{1}{2m} \sum_{i=1}^{m} \left[\left(\vec{w} \cdot \vec{x}^{(i)} + b - y^{(i)} \right) \times 2 x_j^{(i)} \right] + \frac{\lambda}{2m} \times 2 w_j$

$= \frac{1}{m} \sum_{i=1}^{m} \left[\underbrace{\left(\vec{w} \cdot \vec{x}^{(i)} + b - y^{(i)} \right)}_{f(x)} \cdot x_j^{(i)} \right] + \frac{\lambda}{m} w_j$

$= \frac{1}{m} \sum_{i=1}^{m} \left[\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)} \right) \cdot x_j^{(i)} \right] + \frac{\lambda}{m} w_j$

d. Question: Assuming α, the learning rate, is a small number like 0.001, j is 1, and $m = 50$, what is the effect of the 'new part' on updating w_j?
a) decreases value of w_j b) increases value of w_j
Answer - a.

V. Regularized Logistic Regression.

Now, we have $z = w_1x_1 + w_2x_2 + w_3x_1^2x_3 + \dots + b$.

$$f_{\vec{w},b}(\vec{x}) = \frac{1}{1+e^{-z}}$$

a. New cost function:

$$J(\vec{w},b) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\left(f_{\vec{w},b}(\vec{x}^{(i)})\right) + (1-y^{(i)}) \log\left(1-f_{\vec{w},b}(\vec{x}^{(i)})\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2 \Bigg\} \min_{\vec{w},b} J(\vec{w},b).$$

Now, using Gradient descent, repeat {

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(\vec{w},b) \quad \&$$

$$j = 1 \dots n, \qquad b = b - \frac{\partial}{\partial b} J(\vec{w},b) \Bigg\}$$

Now, $\frac{\partial}{\partial w_j} = \frac{1}{m}\sum_{i=1}^{m}\left[\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)\cdot x_j^{(i)}\right] + \frac{\lambda}{m} w_j$

$$\frac{\partial}{\partial b} = \frac{1}{m}\sum_{i=1}^{m}\left(f_{\vec{w},b}(\vec{x}^{(i)}) - y^{(i)}\right)$$

$f_{w,b}(\vec{x}^{(i)})$ is the logistic function.