ML Crash Course

Supervised Learning

Use data which is collected historically, and help to predict new data in future

label – what do we want to predict (y)
Features – input variables describing the data (x)
Example – one piece of data (x, y)
unlabeled example – no label present in the data
model – Used to predict

→ Use labeled examples to train the model.

EX:-
Email spam detectors

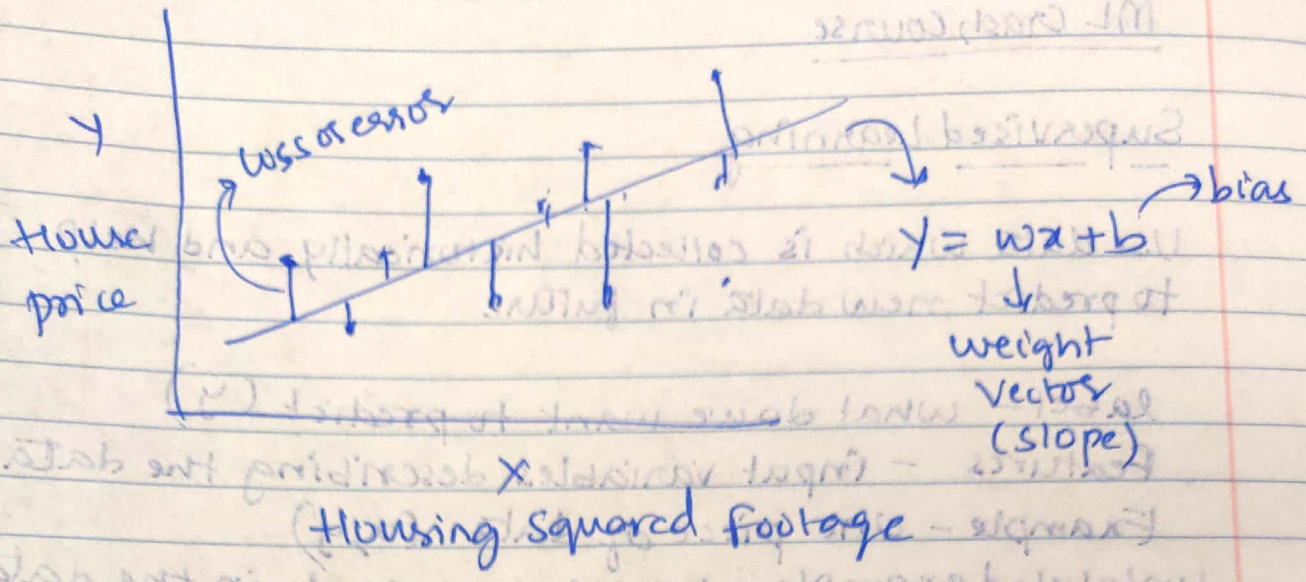
Here labeled example is which are already mentioned as spam or not spam

Once we train our model using labeled examples, we predit the unlabeled examples using the model.

Supervised Learning

Regression
→ predict real valued output
EX:- value of house in California?

classification
→ predict discrete valued output
EX:- Is email spam or not?



Loss → Squared Error [Square of difference between actual value and predicted value)

for entire dataset

$$\text{Loss} = \sum_{(x,y)} (y - \text{prediction}(x))^2$$

Training a model :-

Learning good values for all the weights and the bias from labeled examples. and attempts to find the model which minimizes the loss.

"emperical risk minimization"

Loss :- Penalty for making wrong decisions

Goal of training model is to minimize loss across all the training examples

Squared Loss :- ~~(Observed -~~

$$(\text{actual value} - \text{predicted})^2$$

$$= (y - y')^2$$

Mean Squared error :- Average Squared loss

$$MSE = \frac{1}{N} \sum_{(x,y)} (y - \text{prediction}(x))^2$$

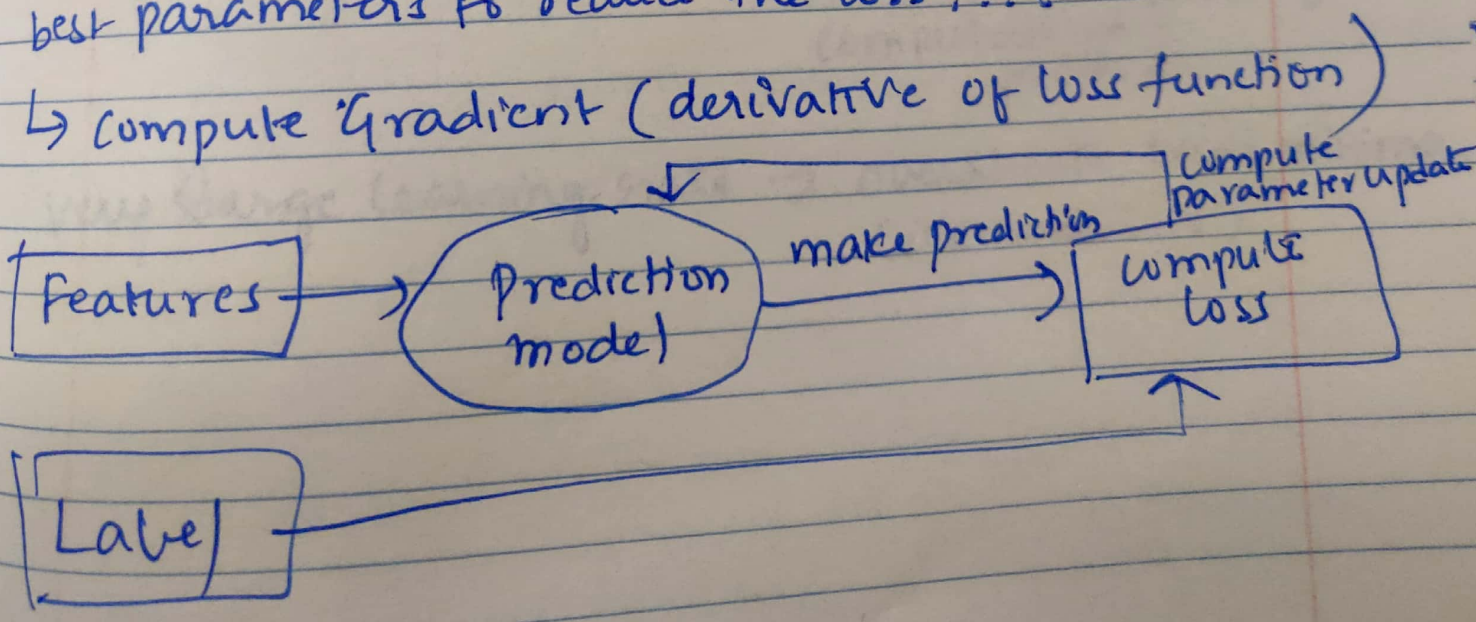
N - no of examples

y - actual value

x - feature (input)

Now, who is the guide to find out which are the best parameters to reduce the loss ???

↳ Compute Gradient (derivative of loss function)



Stochastic gradient Descent-

Batch → Total number of examples you use to calculate gradient in a single iteration.

what happen when a batch is entire huge dataset?

* Extremely computationally expensive *

→ We can draw samples randomly from dataset, but this might lead to redundancy

→ As the batch grows, redundancy might increase

→ How to get right gradient on average for much less computation?

choosing random samples from dataset and calculate big average estimate

SGD (stochastic gradient descent) only uses single example (batch size 1) per iteration

Stochastic means one example comprising each batch chosen at random.

Mini SGD → Compromise between full batch iteration and SGD. (10, 100 examples at random)

It reduces SGD noise but works better than Full batch