DATASCIENCE

Day 1

Learning is of 3 types- supervised learning,unsupervised and reinforcement learning.

# SUPERVISED LEARNING

* Supervised learning involves training the data ,labelling it and then detetcting the labelled data in the future
* Supervised learning is of 2 types – classification and regression.
* Classification deals with categorical outcomes while regression deals with numeric outcomes.

# UNSUPERVISED LEARNING

* Deals with unlabbeled data
* Building music recommender systems

# REINFORCEMENT LEARNING

* Works on action-reward principles
* Self driving cars ,game playing agents

# DEEP LEARNING

* Can be used for all types-supervised,unsupervised and reinforcement.
* It deals with more on improving accuracy than focusing on how the predictions are being taken and why.
* Some of the barriers are the availability of abundant precise data,extreme computing capacity.

# LINEAR REGRESSION

* Involves choosing the best fit line
* If line is y= ax + b an the external point is (p,q) and learning rate is c then the best fit line is y=(a+pc)x + b+c or y=(a-pc)x +b-c
* When there are many points with some over the line and some under the line we use the square method with a point (p,q) over the line and (p,q’) under the line,then line equation is

Y=(a+pc(q-q’))x + b +c(q-q’) or y = (a- pc(q-q’))x + b – c(q-q’)

# GRADIENT DESCENT

* Used to minimize the errors.
* Weight = weight – learning\_rate\*(derivative of error wrt weight)

# MEAN ABSOLUTE ERROR

* A point P(x,y) is given and a line and the point’s projection on line is Q(x,y’)
* Then mean absolute error = )/m , where m is the number of points in the dataset

# MEAN SQUARED ERROR

* A point P(x,y) is given and a line and the point’s projection on line is Q(x,y’)
* Then mean squared error = (1/(2\*m) )\*

# MINIMIZING ERROR FUNCTIONS

* Derivative of error function for a single point wrt slope is –(y-y’)x and wrt y intercept is –(y-y’)
* Total squared error = size\_of\_dataset \* mean squared error

# MINI BATCH GRADIENT DESCENT

* Batch Gradient Descent – Values are calculated for all points,then they are added and weights are updated with the sum of those values
* Stochastic Gradient Descent – Values are calculated point by point ,they are added and weights of the next point are updated with the values calculated.
* But both are slow when dealing with a huge dataset ,so we use mini batch gradient descent where data is divided into small batches and then batch gradient descent is applied.Each batch is updated with the weights calculated for the p revious batch.

# MULTIPLE LINEAR REGRESSION

* We find a line for the linear relationship between two variables,but when it is more than 2 variables we find a plane
* Equation for n dimensions is y’ = w1\*x1 + w2\*x2… +w(n-1)\*x(n-1) + w(n)
* Linear regression can be solved using a system of equations also, but it is feasible when value of n is small.

# LINEAR REGRESSION WARNINGS

* Outliers have a large effect on best fit line
* The relationship between the variables should be linear

# REGULARIZATION

* We find that polynomial regressions are more accurate than linear regressions ,but they also are more complex than the linear regressions.
* They have a higher complexity error tho.
* There are two forms of regularization – L1 and L2.
* L1 complexity = sum of the absolute coefficients of the polynomial
* L2 complexity = sum of squares of coefficients of the polynomial.
* Applications in rocket science and medicine doesn’t tolerate errors so polynomial regression is suitable in those areas while linear regression can be used for song recommender system and youtube recommendations.
* Lamda is a factor used to minimize the complexity error,when multiplied by a smaller lambda polynomial regressions are preferred but when lambda is large linear regressions are preferred.

|  |  |
| --- | --- |
| L1 | L2 |
| * Computationally inefficient | * Computationally efficient |
| * Faster for sparse outputs | * Faster for Non-Sparse Outputs |
| * Feature Selection | * No Feature Selection |

# FEATURE SCALING

* Feature scaling is transforming data to a k n own set of common values .There are two methods – Standardising and Normalizing
* Standardizing :- feature(standard) = (feature-feature.mean())/feature.std()
* Normalizing :- feature(normalized) = (feature – feature.min())/(feature.max()-feature.min())
* We should use feature scaling when we are dealing with distance based metrics and when we are incorporating regularization.