UNIT 3 → Unsupervised Learning

**Linear Regression ::**

* Methods for finding the Linear regression parameters (w0, w1, w2..wn) ; where–
  + n = no.of features
  + m is no. of examples.
* **Normal Equations method** - We do matrix operations like multiplication, inverse to find the w vector i.e., w0, w1,w2…
* To find w’s where Loss is minimum ie, find the derivative of Loss function and equate to 0.
  + **Properties**
  + Suitable for Small datasets.
  + Time taken is O() as it involves the inverse of the matrix.
  + But not all matrices are invertible.

Solving a Problem using Linear Regression

[Untitled0.ipynb](https://colab.research.google.com/drive/1yMTCytTBqU-44taRLfg4eAz8gG8ZFG6D?usp=sharing)

**Derive Normal Equations Formula :**

* Given the feature matrix x and target vector y, we will find the regression function,
* X values are present in the dataset, We have to find ’s ie, w1, w2,w3,....wn.
* To get the optimal values of W’s, the difference between to be close to.
  + - where m = no.of observations/rows/examples in a dataset.
    - Our objective is to minimize the loss.
    - Find those W’s where the derivative of loss function is zero.
  + On simplification;
  + Apply partial derivative ;
  + On further solving;
  + On equating to zero.

**Gradient Descent Algorithm :**

Steps :

1. Choose initial values of w0, w1,w2.
2. Repeat

* until Convergence.

1. Criteria :
   1. Derivatives are 0, hence w’s are not getting updated.
   2. Repeat only specified no. of times.

[Polynomial Regression Proble](https://colab.research.google.com/drive/1v5o6hSgWNWO_cbbvlrZB4KLKXUvKa1VI?usp=sharing)m - Collab link

**Polynomial Regression ::**

* Dataset has one feature X
* Other features are generated as functions of X.

* Linear regression line :
* Polynomial regression line : .
* **Train Set Accuracy** : Linear Regression << Polynomial Regression(1 feature) << Polynomial Regression(n features).
* In the model building i.e, In finding the W’s of the regression function, Dataset is divided into 2 subsets.
  + **Train dataset** : training/fitting the model dataset.
  + **Test dataset** : These data points are not used in the training process. It means they are new to the model.
* If the model is used for predicting the y values, on the test data and performing well then the model is a good model.
* Any model should perform well on the training data and test data, then that particular model is said to be generalized well.
* If the model is performing well on train data and not on test data, such a model cannot predict the values correctly. So it is a useless model. Such a scenario is known as **Overfitting.**
* **Underfitting :** The model is not performing well on the train data as well as test data.
* **Bias** : The difference between train and test accuracy is large.
* **Variance** : Model performance is varying on different subsets of the dataset
* An ideal model should have less Bias and less variance..

**Solution to Overfitting :**

Overfitting issues can be resolved using **Regularization**.

**Regularization** : By forcing some of the coefficients of higher order terms closer to zero, the complexity of the model is reduced. Thus, solving the overfitting issue.

There are 2 methods of regularization ,

1. |W|

**Ridge and Lasso Regression** : Takes care of overfitting problems by making some of the features as insignificant.

* L&S regression is a feature selection method i.e., we are going to choose some of the features.
* As we increase the Alpha value, some of the coefficients/W’s become 0.
* With high dimensional dataset, the problems are,
  + More training time is required.
  + You may have an overfitting problem.
* To overcome this, use feature selection or dimensionality reduction methods.

**Ridge Regression :**

+ → Regularization term

**Lasso Regression :**

+ |W| → Regularization term

**The Loss function for R&L :**

L = + Alpha\*Regularization term { Alpha → regularization parameter}

Regularization term :

1. —> Ridge regression
2. |W| —> Lasso regression

**Implementing R&L using a Dataset:**

[Lasso- Ridge Regression Problem](https://colab.research.google.com/drive/1mQa5voZZxD97KKQ6HOWIV7Eu43Uvfuty?usp=sharing)