Machine Learning

3 kinds of machine learning algorithms are present.

1. Regression
2. Classification
3. Clustering

**Regression:**

One example of Regression is, predicting marks of the student by taking the previous data of the student. In the regression we are predicting one **continuous variable.**

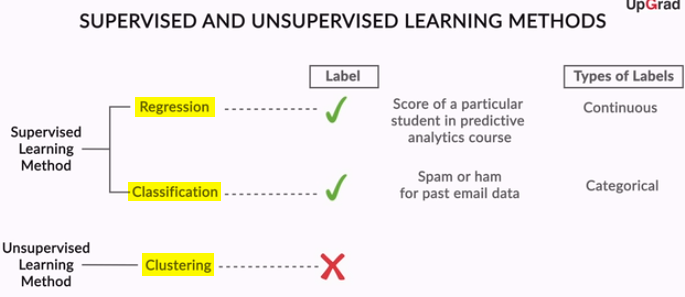
**Classification:**

One example of classification is, predicting if one email is spam or not. Here the prediction is a **categorical variable.**

**Clustering:**

In this category, customer segment is split in to multiple categories. For different categories different kind of offerings can be given.

**Note:** The main difference between classification and clustering is, in classification we are already given the levels as part of the dataset. In the clustering, based on the data property, different categorical clusters are created.

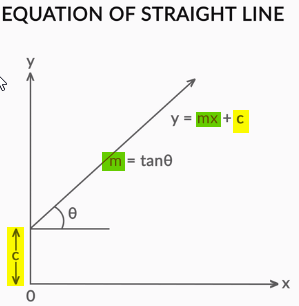


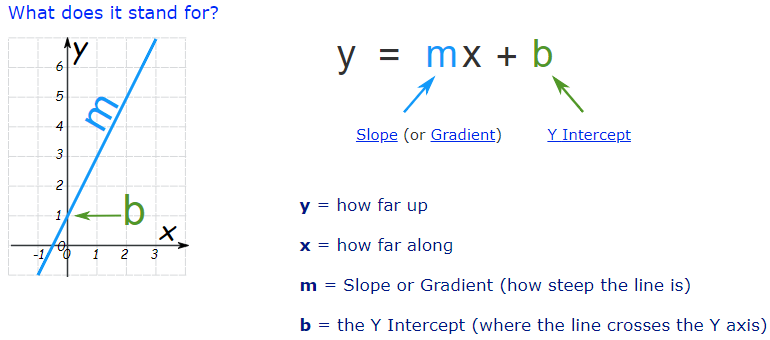
An **independent variable** is the **variable** that is changed or controlled in a scientific experiment to test the effects on the **dependent variable**. A **dependent variable** is the **variable** being tested and measured in a scientific experiment. ... The **independent** and **dependent variables** may be viewed in terms of cause and effect

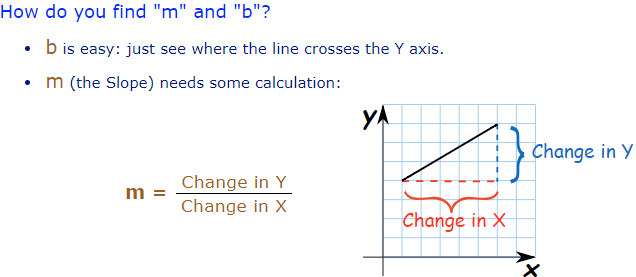
# Simple Linear Regression

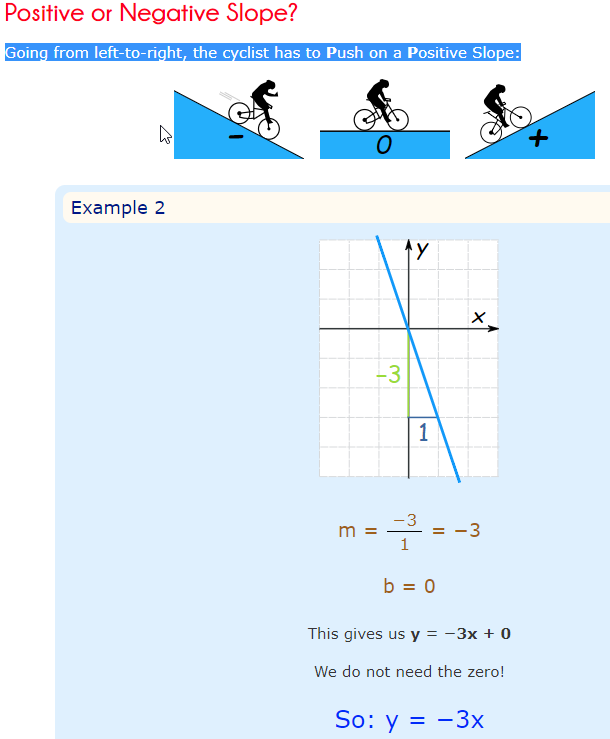
This is the model where one independent variable is used to create the model.

## Equation of a Straight Line

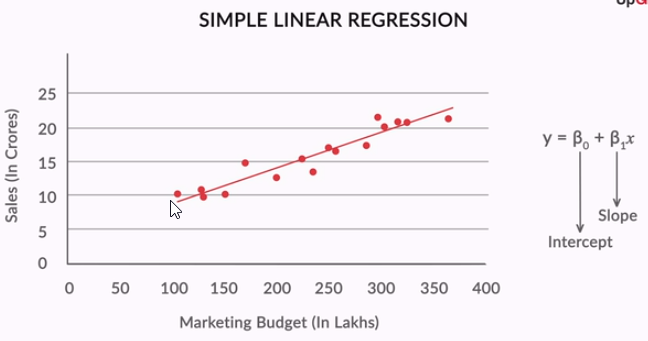




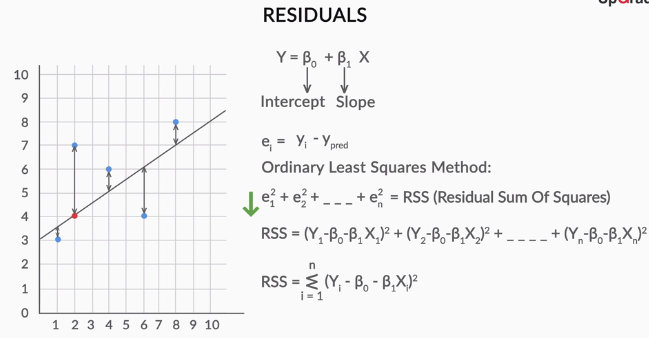




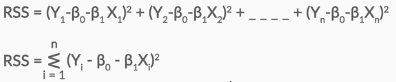
## Simple Linear Regression Example



## Best Fit Line



To make best fit line, we need to find out the residual of each point i.e. e (**actual – predicted**). The target is to minimize the RSS (Residual Sum of Squares)



As we need to minimize RSS, e calculations ie done above by replacing Y\_Pred with intercept slop formula.

**Here Cost function is RSS.**



**Differentiation**: In the differentiation method, make the quadratic equation for Beta0 and Beta1 and solve the equation to get the value of Beta0 and Beta1.

**Gradient Descent**: In the Gradient Descent method, iteratively move for better Beta0 and Beta1 from one initial value to minimize the cost function.

## Strength the Linear Regression

To strengthen linear regression, TSS (Total Sum of Square) can be calculated and from value of RSS and TSS R-Square can be derived.

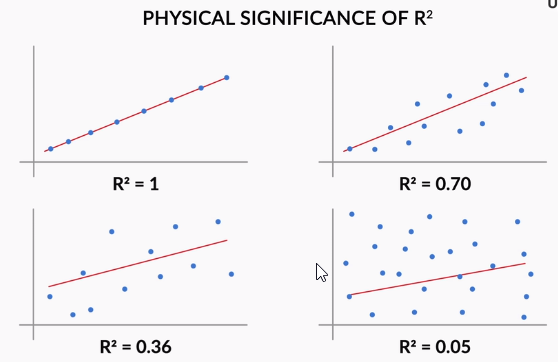
Using only RSS, if the unit of the “independent variable” i.e: X is changed then RSS value is going to change. Hence the total calculation will be impacted. Hence R-Square method is better.

**How to calculate TSS:**



TSS value is basically difference square between Y Actual and Y Mean.

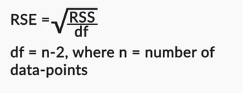


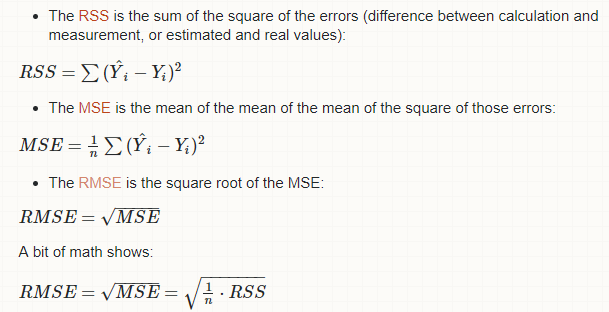


**Note:**

**From above graph it can be concluded that if R2 is close to 1 then it is the best fit.**

Like RSS there is another parameter to do the prediction called RSE. (Residual Square Error). RSE is having same kind of disadvantage like RSS.

****

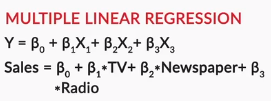
****

## Assumptions of Linear Regression

|  |
| --- |
|  |

# Multiple Linear Regression

In multiple linear regressions, multiple features are taken in to consideration for prediction.



* 1. Understanding of the Python code

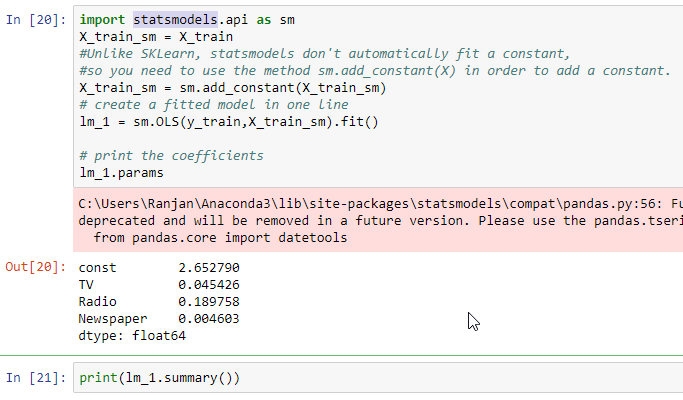
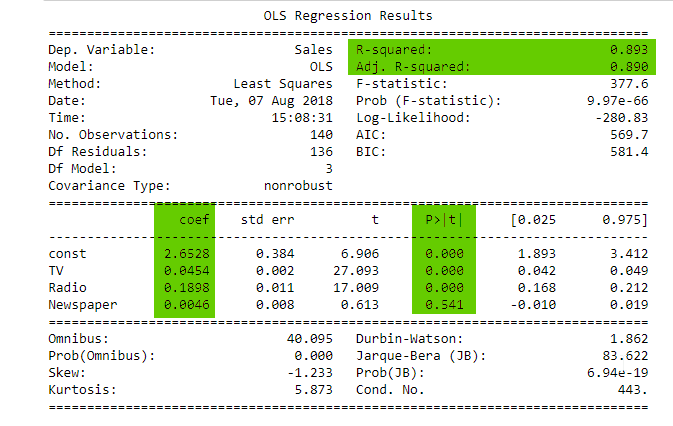
1. Using pair plot, all fields to all field relation can be plotted, as shown below



1. Pair Plot for feature vs action.



1. Basic train test split, fit to the model and prediction part is same as Simple Linear Regression.
2. Check P-Value Using Stats Model

**** ****

|  |
| --- |
| * In the above result, the major things to observed are  1. R-Squared 2. Adj. R-Squared 3. P-Value for each feature 4. Coefficient for each feature 5. From R-Squared, it can be derived that what is the accuracy of the model 6. For P-Value, the NULL Hypothesis is, co-efficient is 0. 7. If P-Value is more, that means Null Hypothesis is “True”. That means Coefficient of that feature is 0. If Beta value is 0, that feature can be removed from the equation. 8. In the above example “NewsPaper” P-Value is big, hence can be discarded from the prediction features. But there can be 2 possibilities  * “NewsPaper” variable is not having any relation with Sales based on P-Value. * “NewsPaper” variable is having some impact on Sales but as “NewsPaper” is correlated to some other features, significance of  “NewsPaper” is minimal.  1. Simple Linear Regression can be performed between “NewsPaper” and “Sale” same as it is done in last section for “TV” and “Sales”. After doing this simple regression it is found that, individual testing Coefficient is .04 for “NewsPaper” but in Multiple Linear Regression scenario it is “.004”. Hence  “NewsPaper” is having lesser impact when multiple variables are involved. As “NewsPaper” is related to “TV” which is showing some +ve Correlation. 2. “Adj. R-Square” is useful when many random features are added. If Random features are not useful, R-Square gives the variance where “Adj R-Square” penalized the unused features. |

# Housing Case Study

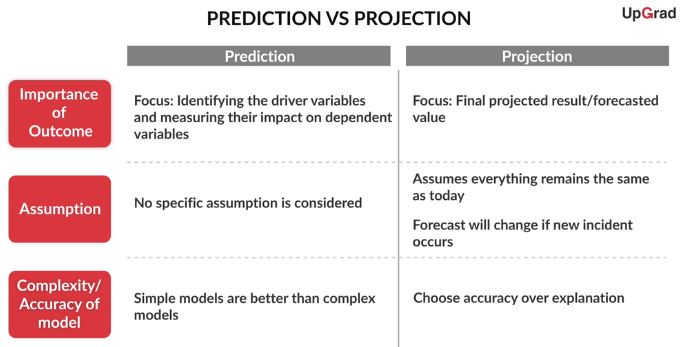
|  |  |
| --- | --- |
| **Steps** | **Detail Explanation** |
| 1 | Cleaning of the dataset |
| 2 | Make all categorical variables to make it numerical. (2 Values in Categorical) |
| 3 | Using “Dummy Variable” Concept, multiple categorical based columns can be splitted in to multiple columns. If there are n categories then n-1 dummy columns will gets created.  <https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faqwhat-is-dummy-coding/> |
| 4 | Create some derived columns based on the domain experience or from the understanding of the column data. |
| 5 | Scale the entire variables in the same unit. 2 Methods majorly used for normalization.  <https://en.wikipedia.org/wiki/Feature_scaling>  {\displaystyle x'={\frac {x-{\bar {x}}}{\sigma }}} |
| 6 |  |

**Primary Process for Linear Regression:**

|  |
| --- |
| Let us recall all the steps we used throughout the linear regression model building process:   1. Once you understood the business objective, you prepared the data, followed by EDA and the division of data into training and test datasets. 2. The next step was the selection of variables for the creation of the model. Variable selection is critical because you cannot just include all the variables in the model; otherwise, you run the risk of including insignificant variables too. 3. This is **RFE (Recursive Feature Elimination)** can be used to quickly shortlist some variables which are significant to save time. 4. However, these significant independent variables might be related to each other. This is where you need to check for multicollinearity amongst variables using **variance inflation factor (VIF)** and remove variables with high VIF and low significance (p>0.05). 5. The variables with a high VIF or multicollinearity may be statistically significant or p<0.05, in which case you will first have to check for other insignificant variables (p>0.05) before removing the variables with a higher VIF and lower p-values. 6. Continue removing the variables until all variables are significant or p<0.05, and have low VIFs. 7. Finally you arrive at a model where all variables are significant and there is no threat of multicollinearity. 8. The final step is to check the model accuracy on the testing data. 9. During one by one feature elimination, if any feature is having 0 p-Value and High VIF, then we should not drop that feature. 1st target for the features with p-value > 0.01 and High VIF. |

**How to Decide How many features to take:**

|  |
| --- |
| 1. Calculate adjusted R-Square using following formula.     N 🡪 Number of data set  K 🡪 Number of features.   1. If we need to take the features from a range of 4 to 20, then plot Adjusted-R-Square for features from 4 to 20 using RFE for both train and test dataset. 2. Observe the point in the plot (Mostly in test data set) where there is no change in Adjusted-R-Square value and becoming fix. |

****

**Question: 1 (Major Assumption in Linear Regression)**

List down at least three main assumptions of linear regression and explain them in your own words. To explain an assumption, take an example or a specific use case to show why the assumption makes sense.

Linearity:

The 1st important assumption for Linear Regression is there should be a linear relation between dependant variable and independent variables. Using scatter plot, we can check if any independent variable is related to dependant variable. If not then it can be checked using some of the derived variables from the independent variables with dependant variables. In the house prediction example, with increase in house area there is increase in house price. If there is no house area parameter as part of independent variable but there is a parameter call house address. If the dataset is provided with multiple house information belongs to same area, then it is not possible to tell from this independent variable if the house price is going to increase or decrease based on that exact lane number of the house. If there is no exact trend of increasing price not found then it is not possible to have the Linear Regression for this scenario.

Multicollinearity:

If there is correlation existing between independent variable then it is considered to be multicollinear. In the linear regression we try to find out change in dependent variable when there is 1-unit change in independent variable with making another independent variable constant. In this way we find multiple independent variable impact to dependent variable.

If the independent variables are collinear, then while changing one independent variable, we can’t make other independent variable constant if it is corelated. Hence, we can’t find the exact relation until the multicollinearity is removed from regression.

In the same house price prediction example, if we have 2 independent variable, house area and number of bedrooms, then we can see both the features are related and both are impacting to the housing price. When we check the regression, if we consider both the independent variables then we can’t make the model proper. Hence, we need to drop one of the features because of the multi collinearity.

**Question: 2 Difference between different model evaluation metrics such as r-squared, adjusted r-squared, RMSE and Residual Plot**

By now you have seen multiple **model evaluation metrics** used for regression models, such as r-squared, adjusted r-squared, RMSE, the residual plot etc.

In this question, you are required to**explain at least three regression model evaluation metrics**in your own words.

1. For the final model that you have built, explain each evaluation metric with its intuition (i.e. what and how it measures) and relate the intuition to its mathematical formula. You may use figures or examples to explain if needed. Limit your answer to 1000 words for this part.
2. Compare the advantages and disadvantages of any three evaluation metrics. If you do not think there's any advantage or disadvantage of a certain metric, mention that. Limit your answer to 1000 words for this part.

# Linear regression Gradient Descent Advance Minimisation

|  |
| --- |
| **Unconstrained Minimization:** Minimization is without any limit. Can go the any minimum value.  **Constrained Minimization:** In this method, there is a limit to the minimum value. Lets see it with some example.  ***1D Example:***    Here we can see, if there is no constrain then the minimum value is 3.  If we can make the constrain that Theta should be <= 2 then we are getting the value as 2.  ***2D Example:*** |

## Unconstrained Minimization

|  |
| --- |
| **Unconstrained Minimization: (1D)**     * In the closed form, direct derivative was taken * IN the Gradient Descent, with the learning rate, we are go to minimum theta value step by step. * As we can see in Closed form, the value is coming 2. Where in Gradient descend, we are going toward 2 in each step. 1.24 🡪 1.42 (Increasing Order) |

|  |
| --- |
| **Unconstrained Minimization: (2D)** |

|  |
| --- |
| **Gradient Descend for Cost Function:**     * Same unconstrained minimization equation is applied for the Gradient Descend with 2 variables. * Here we have x and y actual as per the data set. * Using y = mx + c formula with initial value of m and c as 1, the above gradient descend formula is applied with minimum learning rate to get the Y predicted (Y with hat) * Y predicted and actual Y difference will be calculated as minimum as shown above.   <https://learn.upgrad.com/v/course/162/session/17414/segment/88869> |

## Constrained Minimization

|  |
| --- |
| In the multivariate Linear Regression model (Shows in last curve), there is a chance of **overfitting**. We will see how overfitting happens and how we can avoid it using constrained minimization. |

|  |  |
| --- | --- |
| **Method to Use Excel for getting R^2 Value:**    **Best Fit:**    Over Fitting: (With Polynomial Degree 4) |  |

|  |
| --- |
| * Here M defines to Degree of Polynomial * If Degree is Very High, then the equation will try to fit all the points and leads to overfitting as shown above with value M = 9 * When M value is high, it is observed that the coefficient of each parameter is also very high. Refer the following snapshot for the same.      * To limit the over fitting, we need to put some restriction to the degree of the polynomial. In indirect way, we can put the restriction to the coefficient with higher value. Hence the formula can be something like below.      * Using **Ridge/Lasso** as constrain we can try to minimize the cost function.   **Visualizing Constraints:**    With the constrain of Ridge and Lasso we can see the cost function prediction is not going to local minima. As Local minima can lead to overfitting as weight (Coefficient) will be very high. |

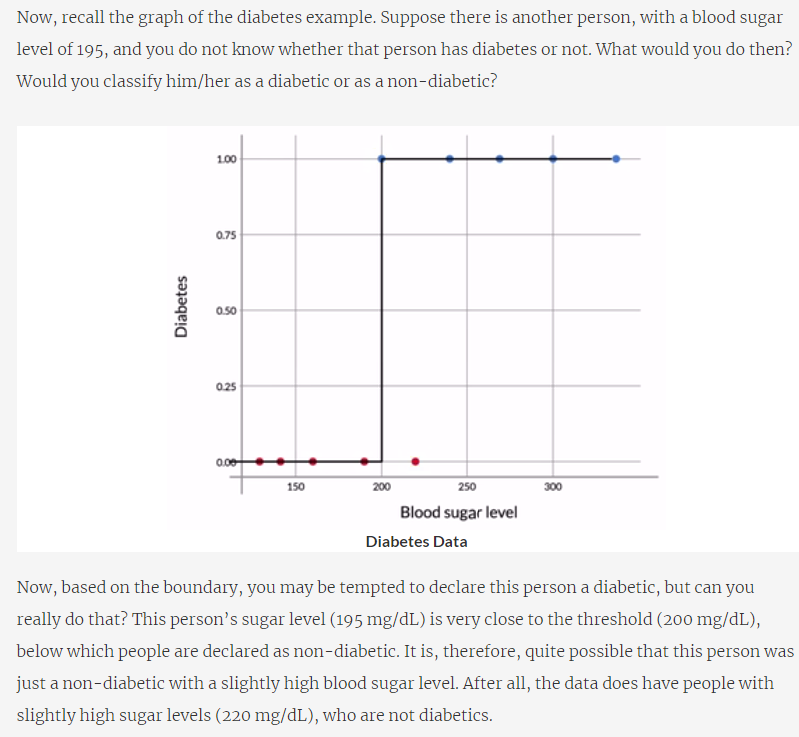
# Basic Algebra for Logistic Regression

|  |
| --- |
|  |
|  |
|  |
|  |

# Logistic Regression

## Univariate Logistic regression

### Binary Classification



### Sigmoid Curve

In the above example, it is difficult to put a line which can clearly distinguish between diabatic or not. Hence, we need to take another approach. In the new approach, we need to find a proper boundary between diabatic and non-diabatic using probability to cover the boundary scenario. Same can be drawn using Sigmoid curve.

|  |
| --- |
|  |

|  |
| --- |
|  |
| .65 and .33 in the sigmoid graph is the probability of diabatic. For 210 (Guess value as per Graph marked in Red) probability is 0.33 where for 240 (Guess Value as per Graph marked in Blue) probability is 0.65 |

### Finding Best Fit Sigmoid Curve

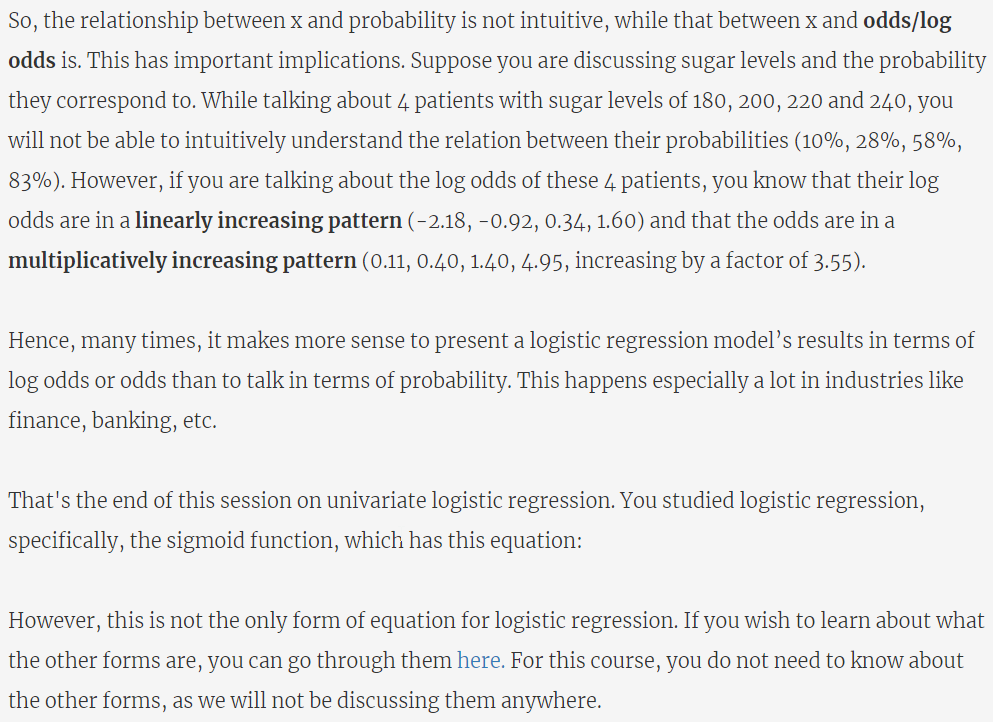
|  |
| --- |
| Beta0: Min and Beta: 1 Max    Beta0: Mid and Beta: 1 Max |
| Note:  For the base fit, we need to minimize P1, P2, P3, P4 in the shown picture and need to maximize P7 to P10. To make it uniform we can maximize product of (1-P1) till (1-P4) and P5 till P10    All “No” which should have minimized is normalize by subtracting by 1 to maximize it. All “Yes” by default maximize  **Here is P is probability which is derived from Beta-0 and Beta-1 values using the cost function of Sigmoid.** |

### **Odds and Log Odds**

|  |
| --- |
|  |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  | When X Increases, Odds increases reciprocally. |

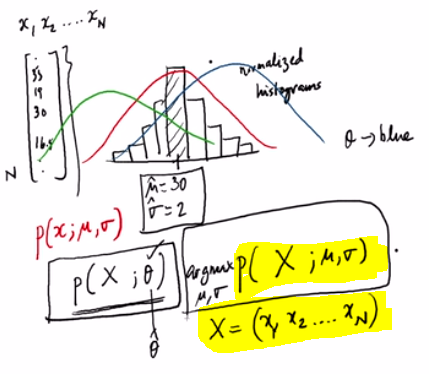
****

|  |
| --- |
| **Optional Logistic Regression Formula:** |
| links = sm.families.linkslogm = sm.GLM(y\_train,(sm.add\_constant(X\_train)), family=sm.families.Binomial(link=links.logit))logm.fit().summary()  links = sm.families.linkslogm = sm.GLM(y\_train,(sm.add\_constant(X\_train)), family=sm.families.Binomial(link=links.probit))logm.fit().summary()  links = sm.families.linkslogm = sm.GLM(y\_train,(sm.add\_constant(X\_train)), family=sm.families.Binomial(link=links.cloglog))logm.fit().summary() |

|  |
| --- |
| **Summary Univariate Logistic Regression:** |

Now the question is how to get the optimal values of Beta-0 and Beta-1 such that, cost function can be maximized. The method used for this is called **“Maximum Likelihood Estimation (MLE)”**

### **Maximum Likelihood Cost Function**

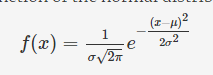


The cost function is to get the best Mue (Mean) and Sigma(Std Deviation) for X Random dataset.

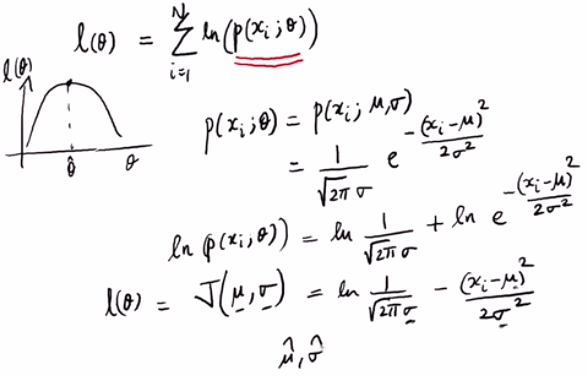
|  |
| --- |
| **Cost Function:**     * + J(Theta) is the cost Function   + Theta is the parameter we wanted to estimate (Mue and Sigma)   + We need to try different value of theta for Value X (x1,x2.. xn) to check which theta is giving maximum probability for X (x1, x2, x3 .. xn)   As per the last figure red gaussian distribution theta is more accurate than others.  **Assumption:**   * + All the value of x in X is independent to estimate theta. Hence it is the multiplication of all the independent estimates. Here Pie is the multiplier (Similar kind to Sigma)   + Log(L(theta)) 🡪 This will give the summation rule. As Logarithm rules tell log(xy) = logx + logy     Suppose we have n samples from independent and identically distributed observations, coming from an unknown probability density function f(x|theta), where theta is unknown.  So, how to arrive at the log-likelihood function? Here are the steps to sum it up:   1. Making a Joint Density Function,  **https://lh5.googleusercontent.com/CfRPCCgsevVZjPAFpKx3KlWmfmIdFYB9KUMszwRlGa9QqmiW4az4cnacgeCr21erww1YcgDPytL1Ovvl5401RpPBzDGtjH48iuwuAF4Kfhs9zqugGZAIqROf4pu9f5lGJriRQn_i**   2. Finding the Likelihood function, x samples are fixed “parameters” and theta will be the function’s variable  **https://lh3.googleusercontent.com/xC3qxjAzUXGbq0uJmSQyZxTisSH8zhxtKE4vyJ6KELqkoWDhC0HmCvq9Ywn6Bw21BJyZZzEiXb3hPxPcLai1h4bySXvuvg1rq1i0cb864dms2AgfhfXAUL8P3Ni9ScOOO7vtBPZ2**   3. For further simplification, we make it Log-Likelihood function, as it's easier to deal with log  **https://lh4.googleusercontent.com/rZxOP2EPZqcbxCAz-gasOjI61WqMVt_WoYP9WrMnbDHQe1LtDGVVZ0zYFYAH_Qpeyh32hl_Kgv90IHxXalILN2vUkjT0b9ZKUu6oPUdT6-j2RYBnCELtCEfWA2ebmWHi33rIkrL_**   On practising machine learning we often use a model to describe the process, that results in the data that are observed. Each model contains its own set of parameters that ultimately defines what the model looks like.   Maximum likelihood estimation is the process of finding parameters for a given statistic which makes the known likelihood distribution a maximum. |

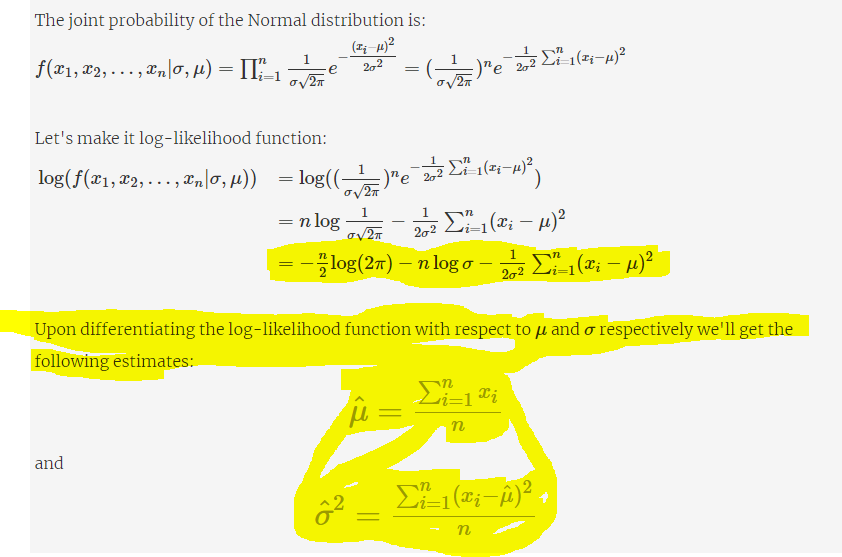
### **Maximum Likelihood Estimation for Continuous Distributions**

Probability distribution function for point x is as below:



Now if we put this function to get the cost function, the formula will be as below.



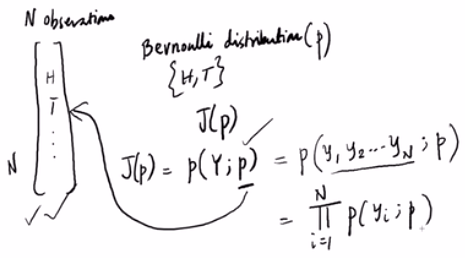


**Note:** Once final cost function is got, we can get the parameter in 2 ways same as linear regression.

* 1. Through Differentiation (which is done above) for both Mue and Sigma (2 Parameter 2 Equation)
  2. Another approach is Gradient Descent where step by step reduce the value of X to get to best Mue and Sigma

### **Maximum Likelihood Estimation for Discrete Distributions**

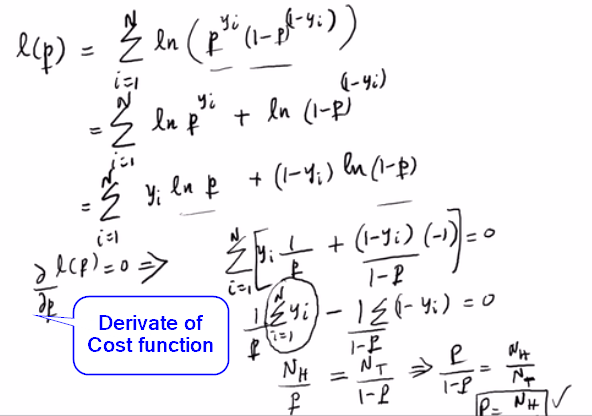
In the discrete distribution technique, we have random observations. (e.g: Toss result Head/Tail for different attempt). We need to find out the maximum probability of getting head/Tail in each attempt.



* + Total number of observations as shown above is N.
  + The cost function is to get the probability of each attempt outcome %.
  + Each attempt will be considered as individual attempt; hence the cost function will be multiplications of each attempt cost function as shown above.

|  |
| --- |
|  |

Following is the detail method how the derivate of the cost function is used to derive the final formula for the probability of each attempt.



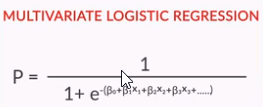
### **Maximum Likelihood Estimation for Logistic Function**

|  |
| --- |
| Lets’ understand cost function of Logistic function with one example.  Derive if a person is Male or Female from one of the features “Length of the Hair”  More number of features will be added later.  “yi”  Male or Female  “xi” Length of Hair     * Here for feature xi different bins are created for different set of parameters. For some set, Male probability is High and for some set female probability is high. To make a better fit, we need to draw the sigmoid lines with Beta-0 and Beta-1 |

|  |  |
| --- | --- |
| **Deriving Cost Function with Combination of Sigmoid and Maximum Likelihood:** | Expression is nothing but Sigmoid Function |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

## Multivariate Logistic regression



|  |
| --- |
| 1. For Logistic Regression Model Building, data set can be prepared by dropping NULL value columns, removing outliers.  2. All the categorical variables can be created with Dummy variables and using RFE max feature count to do the analysis can be set.  3. In the Model execution, feature with high p-Value and VIF can be eliminated 1st. Feature with high VIF but 0 p-value can be given lower priority.  **Note:**  Detail Coding can be referred through “Logistic+Regression+-Telecom+Churn.ipynb” file. |
|  |

### Model Evaluation (Logistic Regression)

|  |
| --- |
| Here 0.50 is the manual assumption. In next part we will check how to take actual assumption. |

|  |  |  |
| --- | --- | --- |
|  | **Predict (Yes)** | **Predict (No)** |
| **Actual (Yes)** | **True-Positive (TP)** | **False-Negative (FN)** |
| **Actual (No)** | **False Positive (FP)** | **True Negative (TN)** |

* FP and FN Based on where “Predict is there”
* Predict and Actual Changes from above example, so FP and FN changed. “F” is fixed but “P”, ”N” based on “Predict” column.

|  |  |  |
| --- | --- | --- |
|  | **Actual (Yes)** | **Actual (No)** |
| **Predict (Yes)** | **True-Positive (TP)** | **False- Positive (FP)** |
| **Predict (No)** | **False Negative (FN)** | **True Negative (TN)** |

|  |  |  |
| --- | --- | --- |
|  | **Predict (No)** | **Predict (Yes)** |
| **Actual (No)** | **True-Negative (TN)** | **False-Positive (FP)** |
| **Actual (Yes)** | **False Negative (FN)** | **True Positive (TP)** |
| **Accuracy:**    Accuracy = (TP + TN) / (TP + TN + FP + FN)  **Sensitivity and Specificity:** | | | |

|  |
| --- |
|  |

### Model Evaluation Comprehension

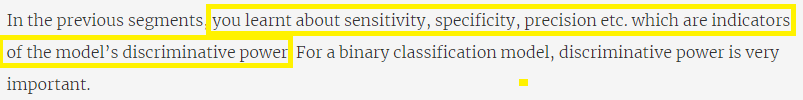
|  |
| --- |
| **Precision, Recall and F1-Score**      # Accuracy, precision, recall and f1 score  metrics.accuracy\_score(y\_true, y\_pred)  metrics.precision\_score(y\_true, y\_pred)  metrics.recall\_score(y\_true, y\_pred)  metrics.f1\_score(y\_true, y\_pred)  If number of classifier is more than 2 then following link can be refer to do the classification. As ROC works on binary classification, we need to make each classifier as one vs all.  ROC curve is calculated for a binary classification problem. For a multiclass classification problem, you can use one-vs-all logistic regression from sklearn and then get ROC score as shown  <http://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html> |

### Gain, Lift, KS Statistics and ROC for Model Evaluation

 🡪 This attached document can be seen to understand the calculation for Gain/Lift and KS Statistics Chart

**Important Note:**

**All these 3 evaluations are done by sorting the probability in the descending order.**



|  |
| --- |
| **Gain Chart:**    **Lift:**      KS Statistics:    ROC (Receiver Operating Characteristic) |

### Summary

|  |
| --- |
|  |

## ROC Curve (Model Evaluation Method)

* + ROC Curve is the plot between TPR (True Positive Rate/ Sensitivity) and FPR (False Positive Rate/ 1 - Specificity) to give a balance value between these 2 for the better prediction.
  + Sensitivity = True Positive Ratio (TPR), Specificity = True Negative Ratio (TNR)
  + FPR 🡪 1 – Specificity

|  |
| --- |
| * In the above example, if cut-off is kept after , then above confusion matrix will be output and FPR and TPR calculation is done accordingly. Here the noting point is “Green” is “True” and “Red” is False. * For ROC Curve, we need to take each cut-off point and get the confusion matrix for that point with FPR and TPR. Check the following snapshot for reference.      * ROC Curve for the above points will be as follows.     Above pink colour marked points are FPR to TPR plotting point   * **AUC (Area Under Curve)** tells the area under the ROC Curve which is drawn above.      * **More the AUC is better the model is** * Refer the Python code provided to draw the ROC curve and get the AUC to validate the model. |

## Logistic Regression-Industry Applications

### Nuances of Logistic Regression (Sample Selection)

|  |
| --- |
| 1. Sample selection is an important task. E.g. From bank data if we need to find the loan defaulter, then using last 10 years data looks relevant. If the close look of the data says that, 1st 7 years the loans were given to Salaried Employee and from last 3 years the loan has been introduced to students, then the data set will have the dominance of Salaried employee data (7 Year Data) over student data. Such kind of scenario, better to split the data in to 2 categories and run 2 models independently to check the result. |

### Nuances of Logistic Regression (Segmentation)

|  |
| --- |
|  |

### Nuances of Logistic Regression (Variable Transformation-I)

|  |
| --- |
|  |

### Nuances of Logistic Regression (Variable Transformation-II)

|  |
| --- |
|  |
| **Pros & Cons of WOE:** |
| **IV (Information Value):**           * Important things to Note from the previous example is NA is having some meaning in some scenario. So, replacing the NA with mean or median Is not a good idea. We can think of replacing the NA with similar OWE value. In bellow example, NULL value can be replaced with value in 87% to 92% range. As OWE value is nearly similar in both NULL and this scenario(87% to 92%). |

# Naive Bayes

This algorithm is a classification algorithm for text-based content. E.g. if one email is spam or ham.

## Bayes’ Theorem and Building Blocks

### **Conditional Probability and Its Intuition**

|  |  |
| --- | --- |
|  | Probability of SPAM will increase if there is any unusual word present in the email. Here presence of “VIAGRA” increases the probability of the mail as Spam. |
|  | Without any condition the probability mention is called “Prior” probability.  With addition of condition, the probability changes. It is called “Posterior” probability. |
|  | |

### **Bayes’ Theorem**

|  |
| --- |
| P(A) 🡪 Probability of India Win  P(B) 🡪 Probability of Sachin’s Century  **Joint Probability:**  Probability of India Win AND Sachin Century (P (A, B))  P(A∩B) = 10/100 = P(B∩A)  **Conditional Probability:**  Probability of India Win Given Condition Sachin Makes Century  P(A|B) = (10/100)/ (12/100) = 10/12 = P(A∩B)/P(B)   * The important note in the conditional probability is we need to take the all favourable outcome based on the condition B. Based on that favourable outcome, we need to calculate probability of A.   **Interpreting Joint Probability in Term of Conditional Probability:**  Probability of India WIN And Sachin Century  P(A|B)\*P(B) 🡺 Probability of A given B \* Probability of B |

|  |
| --- |
|  |

|  |  |
| --- | --- |
|  | |
|  | Explanation provided not correct.  Actual Explanation:  160/200 = 4/5 |
|  | Explanation provided is not correct. It should be  240/800 = 3/10 |
|  | P(Green|Murli Says Green) = **P(A|B)** = [P(B|A) \* P(A)]/P(B) OR  P(A B)/P(B)  Using Formula 1:  [(160/200) \* (200/1000)]/(400/1000)  = (16/20)\*(20/100)\*(100/40)  =  = 16/40 = 2/5 |

## Naïve Bayes For Categorical Data

### **Naïve Bayes- With One Feature**

|  |  |
| --- | --- |
|  | |
| * Here “Type of Mushroom” is the output where we have multiple features as shown above. Lets’ do analysis of single feature and impact on output using Naïve Bayes. | |
|  | * With this condition we will find the probability of the Mushroom is edible or not.     Ci 🡺 Type of Mushroom  Xj 🡪 jth feature |

|  |  |
| --- | --- |
| **Single Variable Analysis:** | |
|  |  |
|  |  |

|  |
| --- |
| * As P(X) is there in both +ve and -ve case, it can be considered as just a scaling factor and can be removed as part of the calculation in further cases.   **Final Result for One Feature:**     * As Probability of Edible = “Yes” is more than “No”, test point is consider as Edible. |

### **Conditional Independence in Naive Bayes**

|  |
| --- |
| * If assumption is taken that the features are conditionally independent then the above probability can be split in to 2 sub-set of probabilities as shown below. * This assumption is called Naïve Assumption. |

### **Deciphering Naive Bayes**

|  |  |
| --- | --- |
|  | |
|  |  |
|  |  |

|  |
| --- |
| * Prior Probability is more then it will impact the Posterior probability. * Likelihood function and Prior Probability are directly proportional to Posterior probability. |

## Application of MLE (Naïve Bayes Classifier)

|  |
| --- |
| **Example:** Predict Male/Female Based on Hair Length     * For each feature X (here Length of Hair) we need to see for each category (Male/Female) if the distribution is gaussian. Hence and of X for male and female should be fetched for further calculation.        * As here the features are continuous variables, Maximum Likelihood for Continuous distribution formula will be applied to calculate the value of and |

## Naïve Bayes for Text Classification

|  |  |
| --- | --- |
| * We have 5 document which is classified to 2 categories. * Lets’ split all the documents with distinct words. * Delete all the commonly used words or “stop words”      * Count number of times each word occurred in each document. It is Bag of Words; hence the Sequence of the sentence doesn’t matter. As the complete set of words are taken as a “Bag of words”.        * If we put the above category in matrix form and separate based on class (Education/Cinema), following matrix we can see. * 13 represents, total number of words occurs in Class **Education**. * 8 represents, total number of words occurs in the Class **Cinema**.     **Problem Needs to Solve:**   * Given a New Document whether it belongs to “Education” or “Cinema”        * Using above table, when the new document comes, probability calculation of “Document” or “Cinema” can be done. * Lets’ Check “Great Story” document belongs which category. | |
|  |  |
| * Lets’ consider a scenario where we wanted to check document “Very Good educational institution”, whether belongs to “Education” or “Cinema” * When we will take each word from this document and check the probability in both “Education” and “Cinema” class, it can be found that word “educational” is present in “Bag of Words” but not present under “education” category. Hence for education category, probability will be becoming 0 for word “educational” * In such scenario “Laplace Smoothing” is applied. | |

### **Laplace Smoothing**

|  |
| --- |
| * As per “Laplace Smoothing” if one word is giving the probability as 0, then increase the value by 1 and also increase the count for other words by 1 as well. * Her in below example, word “Good” was having the value 0 before. Now 1 is added to make it as a proper value of probability calculation. The result of which, other variables are increased by 1 each. Hence total count of words in “Education” changes from 13 to 25. Also, in Cinema it is 20 from 8. |
| * **As overall count changed, overall probability is changed for both the class.** |

## Bernoulli Naive Bayes

|  |
| --- |
| * This theorem counts one word only one time if the word occurs multiple time. * Other calculations will be same as above.      * Detail calculation and example of Bernoulli is provided inikh below attached document. Same Laplace Smoothing applied for 0 probability scenarios. |

# Unsupervised Learning

## Clustering

In the clustering task, the categories are unknown. Based on random dataset property, the categories can be derived. E.g. Grouping of customers of one online store in to different clusters and target separate targeted market strategy.

|  |
| --- |
| * Clustering is the technique to split the data set based on certain similarity. * Segmentation 🡪 This basically tells., how to split the data to have a better business.   To do segmentation we use clustering technique. |

## K Means Clustering

|  |
| --- |
| K-Means 🡺 K Clusters       1. For a given dataset, take 2 random points called “Cluster-Center”, if number of clusters decided as 2. 2. Find the distance of each dataset from these 2 random data set and categorize the data set to cluster. In above example, some data set belong to red cluster and some dataset belongs to yellow cluster. Categorization happens based on which cluster is near to the data set. If data point P1 is near to cluster C1 than C2, then P1 will be categorized as C1. This step is called **“Assignment Step”** 3. After “Assignment Step” is done for all the dataset, we will have the cluster wise categorical data set. For cluster C1, we have some set of data points and for cluster C2 we will have some set of data points. Now we will calculate the “Mean” of data set belongs to C1 category and C2 category which will be the “cluster-center”. This step is called **“Optimization Step”** 4. Now follow the Step: 2 and Step: 3 iteratively, until we get the best “Cluster-Center” from which the distance values are not changing for whole dataset. 5. Check the step by step calculation of cluster in the attached excel sheet. When the Cluster value is becoming unchanged, no need to proceed further.        1. If Data set is represent in 2-dimentional as (x1,x2) then cost function will be as follow.      1. Here we wanted to minimize the Euclidian distance between each data point from the cluster. 2. is consider as k clusters where represent I th cluster from K cluster.     **Assignment Step:**   1. In above example, we can see (X1, X2) data point need to categorize in to or 🡺 as per the explanation.     **Optimization Step:**     1. In above example, we need to recalculate and by taking mean of each category     **Real Time Example:**  **http://thespread.us/clustering.html** |

### **K Means as Coordinate Descent**

|  |
| --- |
| 1. Here we have 2 steps 2. Assignment steps where we Compute Z given Random 3. In Optimization step, we compute from given Z. |
| 1. In the Linear Regression, Global Optima is same as Local Optima. 2. In K-Means, the Local Optima (LM) can vary based on the initial value of cluster value. Which may land to different local minima than Global Minima. In the above example, we can see Global Minima (GM) is at one point, but LM can vary based on different initial value of cluster. Hence it is called **“Non-Convex Optimization”** |

### **K Means++ Algorithm**

|  |
| --- |
| * One cluster center can be picked randomly in the dataset and all the data points distance squared ()can be calculated. * Then one of the data pointshaving maximum probability of bigger will be picked for the 2nd cluster center. * From 2nd cluster center, again calculate for all other points. Same as previous step, if any distance is probabilistically more then pick that point as 3rd cluster. * **These 3 cluster can be considered as initial random cluster for the K Means Algorithm.**   <https://www.naftaliharris.com/blog/visualizing-k-means-clustering/> |

### **Practical Consideration in K Means Algorithm**

|  |
| --- |
| **Major practical consideration in K Means Algorithm are** |

|  |
| --- |
| **How to Choose K**     * If we consider M(k) is any random matrix which is taken “k” as number of clusters need to consider, then like above graph, we need to find some point where M(k) is giving larger value. Note here that we are not sure what is matrix we are using for M(k). * It is also not guaranteed that number with M(k) with higher value can give better count of cluster. It is possible to have different count of cluster to get the **better insight** about the data set.      * In above example, we can see, if k =2 then all the elements are not closed to each other (Logically can have 3 cluster). Hence the data points are not tightly coupled or **not cohesive**. * In the scenario of K = 4, we have created cluster A and B which is not very dissimilar.   So, to choose K, we should use the Metric (M(k)) which should be “Cohesive” and “Dissimilar”     * From above understanding we concluded that for every data set “i”, we need to have a(i) 🡺 Avg Distance from Own Cluster is as small as possible where as b(i) 🡪 Avg distance from nearest neighbour cluster is as large as possible. * S(i) is the formula to accommodate both a(i) and b(i) to make a(i) smaller to b(i) * For every value of k, if we take average of S(i) for all data set, then we can find the max M(k) from which we can derive what should be ideal value of K. * **Silhouette Analysis is used to decide how many cluster is best for the dataset provided.**       **How to Decide If K Mean can be Applicable or not:**     * **Hopkins Statistics** will give the result if the data set is ok for K mean or not. In above plot, for 3rd plot Hopkin Statistics will return 1 where other 2 cases it is returning 0 and 0.5. Hence, we can decide 3rd scenario K mean can be applied. |

|  |
| --- |
|  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Python Execution and Interpretation of K-Means:**   * Check the Python Implementation of k-Means from Python Note Book  1. **Data Preparation:**   Based on business need overall data can be segmented in RMF mode. (Recency, Frequency and Money Value)    For different kind of data set, different kind of rule can be applied to aggregate the data set which meet the business scenario.    **Steps Involved in Data Preparation:**     1. **Making Clusters**      1. **How to Decide How many clusters Needed**        1. **Analysis of the Clusters and Conclusion**     **Other Behavioural Segmentation Types** | | | |
|  | * Min 1 Year data needed to do the RFM segmentation. | | |
|  | This is mostly in the kind of online shopping.   * Type of Person, we can check some property like whether he buying for himself or for other. If delivery address and staying address is not same then he is buying for other. If gender of the person and type of product (Gender Basis) is different then also some derivation can be done. These are just some example; other examples can be derived in this category. * Understand the Intent of the person. If person buy some special kind of product always, giving suggestion for other kind of product is not very useful. | | |
|  |  |  |  |
| * In CDJ, 3 different categories of experience are shown. Based on these 3 categories customer segmentation can be done. | | | |
| **Summary** | | | |
|  | | | |

## Hierarchical Clustering

|  |
| --- |
|  |
| * In above picture, random 10 elements are taken and considered all individual as individual cluster. * Now pick 2 closest element and form a cluster (5,7). Now join 8 in to (5,7) to form the cluster and so on. * In each step, number of clusters are getting reduced and tree like formation happen as shown in right side dendrogram. Hence this method of clustering is called Hierarchical clustering. * Height of the dendrogram represents the dissimilarity major. More the height more the dissimilar.        * If the height of the dendrogram is cut in half, then we can see we can get 5 clusters. * Basically, we need to find the cut-off point to cut the dendrogram. * E.g. If We cut it as 1, then we will get 2 clusters. |
|  |

### **Type of Linkages**

|  |
| --- |
| * If n number of points are there, then total number of iterations to form the clusters are n -1 |

## K-Mode/K-Prototype Clustering

|  |
| --- |
|  |

## DB Scan Clustering

|  |
| --- |
|  |
|  |
| <https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/>  <https://medium.com/netflix-techblog/tracking-down-the-villains-outlier-detection-at-netflix-40360b31732>  <https://www.oreilly.com/ideas/clustering-geolocated-data-using-spark-and-dbscan> |

## Why Gaussian Mixture Model

|  |
| --- |
|  |

## Gaussian Mixture Model

|  |
| --- |
| * In GMM, we check the probabilistic approach of data point into a cluster. A given data point, we can derive what is the probability of lying under a particular cluster or other cluster.   **Limitation of K-Means**     * In the above graph, we can see 2 clusters are getting overlapped. In this scenario, the overlapped section either will be consider in cluster: 1 or cluster: 2. So basically in this kind of situation, for the common data points between 2 clusters, we can’t take the proper decision. GMM will help to overcome such problem.      * In above example, C1 and C2 will be given by K-Means. If we see C1, it is more spread hence both edge points of C1 clusters will not very similar. Hence model shape is also important which K-Means not able to identify. Using GMM we can able to solve this problem as well.        * In above example, x1 and x2 are GPS co-ordinate and we have some delivery boy to deliver the item. In above picture, we can see there are 4 roads where the delivery needs to be done. In K-means, we can’t have the cluster like an oval shape. In GMM we can have the cluster with an oval shape to satisfy the business scenario.     **Limitation of 1-D Gaussian Model 🡺 Insight to Gaussian Mixture Model**  (Refer 6.1.5 Maximum Likelihood Cost Function)     * In above distribution, for single feature X1, if need to see the single cluster distribution (Green distribution shown in picture), then it is not so effective and accurate. * If we make the above distribution to 2 cluster or 2 gaussian distribution (Shown in Red Colour) then the distribution looks better. * For the above formula, refer to the previous discussed chapter in 6.1.5.        * In above model, after getting 2 Gaussian distribution, we can use Naïve Bayes classification to judge the probability.      * In above example, though we have a clear cluster for American and Mexican people height with compare to Indian people height. We are not able to cluster well in 1-D Gaussian as the gaussians are overlapping. * In above scenario, Height of “Native America” and “Mexican” both belong to American height category where “Indian” coming in between. Hence when we do the gaussian distribution, there is an overlapping of gaussian.      * In above example we have 2 speaker voice information which is distributed. “Black” points related to one speaker and “Green” points related to another speaker. Now if we go for 1-D gaussian distribution, then we get one single cluster which is located at centre. Based on this Gaussian distribution, we can’t have a cluster for 2 speaker voice. Hence, we need to go for the 2-D gaussian distribution.   **2-D Gaussian:** |
| Expectation Step: (“E”)       * In the assignment step of K-Means, we make the points belongs to cluster 1 or cluster 2. * The same step in GMM is called “expectation” step. In this step, we are taking the probability of each point belongs to each cluster. If we have 2 cluster, then what is the probability of one point to cluster1 and cluster2.     Maximisation Step: (“M”)     * Here is 1 in case of normal K-Means but in case o f GMM this value will be probability of each point in cluster.   <https://brilliant.org/wiki/gaussian-mixture-model/> |
| Working Example With E-M for GMMs  Expectation Step:      Maximization Step:     * Step: 1 was expectation step where the initial point was taken to find the probability of each data point in which cluster. * In step: 2 mue-2 is changed its position as data point for cluster-2 are more. * In the next iteration of expectation step (Step: 3) now the cluster 2 distribution will be stiffer. |

# PCA

## Maths Behind PCA

|  |  |
| --- | --- |
| **Basic Linear Transformation:**  <https://www.youtube.com/watch?v=kYB8IZa5AuE&index=8&list=PLZB6ZNYJM4y98DBySEpOFpdVOkT8b2flP&t=0s> **(Linear Transformation)**  <https://www.youtube.com/watch?v=P2LTAUO1TdA&t=617s> (Change of Basis) | |
|  | **“Alternate Basis”** |
| **In Our Co-ordinate System:** | In Another Co-Ordinate System: |
|  | |
| (-1,2) co-ordinate present in other co-ordinate system with basis vector (b1 and b2) will be (-4, 1) in our co-ordinate system.    Basis vector b1 and b2 of another vector space becoming (2,1) and (-1,1) in our co-ordinate system.   1. Co-ordinate of Another Vector Space (A)\* Basis Vector of Other Vector Space in Our Co-Ordinate System(B) = Converted Co-Coordinate in our Vector Space(C). (Considering our Vector Space Basis is 1,0 and 0,1)   A \* B = C   1. In other word, if we need to get Co-Ordinate of another vector space where the basis vector of another vector space is given and co-ordinate of current vector space is given then   B = A(Inverse) \* C  ================================================================================= | |
| * When we do a linear transformation with 90-degree rotation, the basis vector change from x => (1,0), y => (0,1) to x => (0, 1) y => (-1,0) * How 90-Degree rotation will do the transformation in another vector space?     🡺 Converted to our language by multiplying the basis matrix of another co-ordinate system.  🡪 Apply 90-degree transformation in our vector space by multiplying transformation matrix in our vector space (0,1) and (-1,0)  🡪 Inverse of Basis Vector can be multiplying to get the rotated co-ordinate value in another co-ordinate space.       * Here M is the transformation of a co-ordinate in current vector space. * In another vector space, any co-ordinate does the same transformation by using above formula where A represent the basis vector of other vector space in our language.   **Eigen Vector and Value:**  [**https://www.youtube.com/watch?v=PFDu9oVAE-g&t=55s**](https://www.youtube.com/watch?v=PFDu9oVAE-g&t=55s)  **🡺**A is the transformation matrix which multiply with vector V to get a scaled version of vector V.  🡪 Now right-hand side is multiplying with one identity matrix to make the multiplication with vector.  🡺  <https://www.youtube.com/watch?v=Ip3X9LOh2dk> (Determinant of the matrix) | |

## Why and What PCA

|  |
| --- |
| * During logistic regression, we check the multicollinearity and start dropping the variables. But dropping variables also leads to loss in information. PCA gives a better method to choose variables which are not correlated. |

## Building Blocks of PCA

|  |
| --- |
| <https://www.mathsisfun.com/polar-cartesian-coordinates.html>  (Cartesian Co-Ordinate to Polar Co-Ordinates) |

## Illustration - Finding Principal Components

|  |
| --- |
| * Find Z1 (PCA 1) which is addition of linear combination of all the features. We need to find Z1 which covers maximum variance (coverage) of all the features. * Like we found Z1, we need to find Z2 which should be completely independent to Z1.      * In above example, in 3rd figure PC1 is the line which is closest to all the data points. * PC2 should be completely independent to PC1, hence it should be perpendicular to PC1. So PC2 can be shown as same way in above figure. * Each row is datapoint and each column is basis of the vector space. |

## Comprehension - Calculating the Principal Components

|  |
| --- |
| * Here the original point transformed their new basis to (0.8, 0.6). When the original point is transformed to new basis, we don’t want to change the direction of the vector. Hence it will be result to a eigen vector with some eigen value. |

## Additional Reading on SVD and PCA

|  |
| --- |
| **SVD:**            **Deriving U and V Of SVD:** |

|  |
| --- |
| **PCA:**      **Covariance of all vectors in Matrix Form:**          Diagonalization of above matrix is basically telling the off-diagonal values are 0 and all the variance are present in diagonal.  **As per the Eigen Value Decomposition:**            **SVD to PCA Mapping:** |

## Singular Value Decomposition

|  |  |
| --- | --- |
| Basic Math on Matrix:  ***Diagonalization of a Matrix:***   * We say that two square matrices A and B are similar provided there exists an invertible matrix P so that   A =𝑃𝐵𝑃−1  If A is similar to B, then A and B have the same eigenvalues.   * We also say that a matrix A is diagonalizable if it is similar to a diagonal matrix. * In other words an nxn matrix A is called diagonalizable if A =𝑃𝐷𝑃−1 for some diagonal nxn matrix D and for some nxn invertible matrix P * If an n × n matrix A has n linearly independent eigenvectors the A can be reproduced as a diagonal matrix D such that * A = 𝑃𝐷𝑃−1 , * Where the columns of P are n linearly independent eigenvectors of A and, the diagonal entries of D are eigenvalues of A that correspond, respectively, to the eigenvectors in P. * Steps for diagonalizing an nxn square matrix A   1. Find n eigenvalues (repeated or not) for A and form a diagonal matrix D with eigenvalues on the diagonal;  2. Find n linearly independent eigenvectors corresponding to these eigenvalues, if possible, and form an invertible P ∈ 𝑅𝑛𝑥𝑛 ;  3. A = 𝑃𝐷𝑃−1   * Let us see an example to illustrate these steps. Let us diagonalize | |
|  |  |
| * Here A is the original Matrix * V is the orthogonal matrix which defines the basis of the new transformation. In the previous case it is (0.8,0.6) for one data point. Orthogonal means, all the element in this matrix are perpendicular to each other. Another property of the orthogonal vector is V(inverse) = V(Transpose) * AV = US 🡺 Original Matrix Transformation to new Vector Space Where U is Eigen Vector and S is eigen Value. * 🡪 Adding in both the side to cancel v from left side. * 🡪 As are same for orthogonal matrix, Final SVD formula is this.      * In above example, n features are reduced to k features. S represents a diagonal matrix which will tell which features are having higher contribution. Higher the value of Diagonal value higher the variation captured by feature. * Following “SCREE Plot” can be used to decide the value of k. In below picture we can see the mapping between number of PC (Principle Component) to variance coverage %. * If Business demand can be achieved with covering 80% of the variance, then accordingly K can be choose. PC7 means K value is 7.     **Relation Between SVD and PCA:** | |

## Practical Considerations and Alternatives to PCA

|  |
| --- |
|  |

# K-NN

<https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>

# Model Selection



## Principles of Model Selection

|  |
| --- |
| * Data source for which we need to do the prediction. * System is the manual system which check the data source to decide whether the mail is spam or ham. * Now the System is replaced with a Model which will do the work automatically. * To do that, we have put a Learning Algorithm which will take the data which is taken by system. So Learning Algorithm will do “What” to do with the data. * After Learning Algorithm is ready, we can prepare a final Model. Final Model will have the process “How” the mails are getting classified as Spam or Ham.      * Hypothesis class is the list of classes (Algorithm) considered in the “Learning Algorithm” step. * Among the class of algorithm, the best suit algorithm needs to consider for the model.  Simplicity, Complexity and Overfitting   Based on above kind of data, Following different class of Models can be used.       * **Evaluation of the model should be done on test data.** * Let’s say we have decided to use “Regression” model. Which “Regression” model we should use?              * Red and Blue lines are trying to fit all the training data which is leading to overfitting and the model is complex model. * In the simpler model (Green Line), model try not to reach all the training data. But provide a best fit.    Bias-Variance Trade-off      * In the **complex model “Variance” is high and “Bias” is low**. E.g: For the 1st Guy in the example, if there is change in question paper (input data), he will perform badly. Hence his Variance is High. But his Bias is low as if the input data is similar as training data, then he can crack the exam. * In the **Simpler Model “Variance” is low but “Bias” is high**.     **Regularization:** |

## Model Evaluation

|  |  |
| --- | --- |
| * Process of choosing “Hyperparameter” should follow, following steps.  1. Do the splitting of the data in to train and test. 2. Take some “Fix Hyper Parameter” and prepare the model. 3. Now test the model in the test data 4. Change the Hyperparameter to finetune the model and do it in loop.   In the above process, we are cheating the system by including test data during model creation. As per the rule, test data should be isolated and should be used only when the model is ready. Hence following method can be used, where data set can be split in to 3 part. **Training, Validation and Test**. Above steps can be done not on Test data but in Validation data. Final model can be tested on Test data. | |
| * When data set is small splitting of data to train and test can lead to missing of information. As test data can have some crucial information which part of is not “training data”, which will impact the model. Also this method applicable for above Hyperparameter scenario where we need to split the dataset into 3 categories. * To avoid such scenario, following “Cross-Validation” algorithm is used. * Complete data set is split into multiple segments. Here in below example it is 6. * Now make 1st segment as test data and other segment as train data to make a model. * In the 2nd iteration, make 2nd segment as test data and all other segment as train data and build another model. * Take average accuracy of all the model.   This above process is called “Cross-Validation”. All the algorithm used this method to provide better prediction result. | |
|  |  |
|  | |

## Cross Validation and Hyperparameter Tuning using Python

|  |
| --- |
| * Usually choose Hyperparameter values range from 1 to number of features.      * With above figure, when number of features increase, the complexity of the model increases, and we can see train score will keep on increasing but test score will get stagnant in some point and after that it will keep reducing. * Most right portion of the graph is more complex area where most left section is the simple model area. * As we need to take a model which is neither too complex nor too simple, we need to see the scenario where train score and test score gap is less. In that section we can see what is the number of feature and same can be taken for final prediction. * In above figure, we can go ahead with 10 features. * In another example, for which same plot shown below, we can keep the number of features around 10-12. As after that, we can see there is no change in test score. But as the number of features increase which increasing the complexity, train score is keep on increasing.      * There are some more kind of cross validation which is present from which K-Fold is most commonly used. * For classification problem we use “Stratified K-Fold” if classification data available is not evenly distributed. If we need to classify good or bad customer but the data available with us is with 98% good and 2% bad, then using K-Fold, the random split of train-test will not maintain the same ratio of good and bad. Hence in this kind of scenario, we will use the special “Stratified K-Fold” technique.   **Additional reading:**  <http://scikit-learn.org/stable/modules/cross_validation.html> |

# Support Vector Machine (SVM)

## SVM - Maximal Margin Classifier

|  |
| --- |
|  |

|  |
| --- |
| * In this example (SPAM/HAM Detection), we have dictionary of words where each feature represents the count of words. Feature X, represents word count of some word from bags of words, in one email. |

|  |
| --- |
| **Concept of a Hyperplane in 2D**       * (1) The line distinguish spam and ham is having the equation as shown in picture. This line is called “**Hyperplane**” * (2) For Spam equation should > 0 * (3) For Ham equation should be < 0     **Concept of a Hyperplane in Multi Dimension**         * For a Given data point, we need to find in which hyperplane this belongs to. * Using the co-efficient value, we can check if the formula values are coming > 0 or < 0. Based on that we can decide whether the data point is plane 1 or 2.     **Maximal Margin Classifier:**     * The closest point from the line should be symmetric distance from both the side. In the above example, the nearest blue point to the line and nearest red point to line should be symmetric.          * 2D line equation and nD equations are provided in above snapshot * Add a dummy column with value 1 to normalize the w0 coefficient. * Now in simpler form we can write the nD line equation as the dot product of w vector and y vector. Y vector is nothing but all data point x along with dummy column 1. * Now we have one extra level column which tells in which side the data point is present. If the value is positive 1, then it is one side of plane. If the value is -1, then it is another side of the plane. * By multiplying level l, we can write the formula as shown above in highlighted way.   **Distance of the point from Hyperplane: (Hyperplane in 2D is a line)**     * Distance of a point from the Hyperplane is shown as above. * The square root of sum of square of coefficient can be normalize to 1 to make the distance formula more concise. * Same method is used in below formula by professor as the precondition to calculate maximum margin in the Hyperplane.                     **Limitation of Maximal Margin Classifier:** |

## Soft Margin Classifier

|  |  |
| --- | --- |
| * The 4 points marked in above picture which is near to hyperplane are the points matter to adjust the SVM separator.      * Such points are called “**Support Vectors**” which is used iteratively to find the proper hyperplane.      * Partial Intermingled points can be handled using Hyperplane formula.      * Feature shown below can’t be used using Hyperplane or “Soft Margin Classifier”. We need to use “Kernels” which we will read in next session. | |
|  |  |
|  | |
| **Slack Variable:**       * For each data point one variance is added which can hold the variance to move the hyperplane. This is called Slack Variable. * If value of is less than 1, then the variance is small for the data point. If the value is bigger than 1, then the variance for that point is big, as it is changing the sign to negative. * To have a stable model, we need to have a limit of for all data points. Summation of all datapoint value should be less than C. Where C is the cost of the variance. * If C is high, then the model is very flexible and can accommodate lots of error. Hence the model prediction is not so good. * If the value of C is low, then model is strict. | |
| * In the above picture, if the line is draw as above, then we are actually trying to fit all the data point correctly in the training data. In this way the separation of the model is not evenly split. This will lead to overfitting as any new data point near to red area also will treat as blue as red area is not having any breathing area. * If we make C value little big then the separator will be as shown in below picture. In that way, may be one point which is actually in blue area, we are predicting wrongly as red area but we are making the model more stable for the unknown data points, as the plane is evenly split. | |

## Python Implementation

|  |
| --- |
|  |

## Kernels

|  |  |  |
| --- | --- | --- |
|  | | |
|  | | |
|  | | |
|  | | |
|  |  | |
|  | | |
|  | |  |
|  | | |
|  | | |
|  | | |
| **Kernel Trick:**         * From the above snapshot we can see by doing feature transfermation we are making the non-linear relation to linear relation between dataset. Here “theta” is the function which is doing the linear transformation of the the vector. Finding value of “theta” is difficult. Hence Kernel trick came into picture. * Kernel trick derive a formula called “Kernel function” which can be used directly to do the transformation, rather deriving the value of “Theta” * In the above example, 1 and 2 gives the same result. For 2 we need to derive “thata”. In 1, using the direct formula we can do the feature transformation.   (Question: How forumla for 1 is derived ? Asked in discussion forum) | | |

### **Usage of Kernels in Python**

|  |
| --- |
| * From above execution it is found that, the gama value is very less to make a better fit. As gama controls the non-linearity of the model, we can conclude the given dataset is already in linear form, hence gama plays very less role to make the model better. So, if the gama value is less, then we can conclude the given dataset is already in the linear form which doesn’t need kernel based SVN. Simple linear SVM is enough for classification. Same way if the data is already linear, then value of C is also small. These 2 factors (C and Gama) will make the difference when the data is non-linear and we adjust these 2 factors to make the non-linear data to linear. Already linear data takes a very small value of these 2 factors. |
|  |

### **Choosing a Kernel Function**

|  |
| --- |
| * Do the plotting of the data to see the kind of scatter of data and how they are interrelated. * If the data can be linearly separated, then polynomial kernel is ok to use. * If the data can’t be linearly separated then, RBF is best choice. |

# Trees

## Introduction to Decision Trees

|  |  |
| --- | --- |
| A Basic Example:  Asking a series of questions in nested if-then-else structure. | |
|  | |
| **Step to Install Graphviz:**    <https://graphviz.gitlab.io/_pages/Download/Download_windows.html> | |
| **Linear Regression with Decision Trees:**   * In the following scenario if we wanted to predict the weight with given age parameter then we can’t apply the linear regression directly as for each group of age, the linearity of the data is different. As we can see in the 1st picture, the linearity of the data is different in different age group. * For such kind of the linear regression, it is not possible to have a single linear regression model. For each group one linear regression model can be performed which we can see done in right side decision tree. | |
|  |  |
|  | |
|  | |

## Algorithms for Decision Tree Construction (Gini Index/Entropy-Information Gain)

|  |  |  |
| --- | --- | --- |
| **Concept of Homogeneity:**       * To do the classification based on attribute, 1st find the attribute which is more homogeneous to classify the label. Like in above example let say BP is one attribute which classify the label very significantly with one threshold. May be not 100% but 90% of the dataset we are able to categorized correctly based on the rule of one attribute. Data set under one label is called homogenous. This 100%, 90% is called **homogeneity measure**. | | |
| **Generic Algorithm:** | | |
| **Gini Index:**       * **P 🡺 Play N🡪 Not Play** | | |
| **Entropy and Information Gain:**   * **Information gain** in 2nd case is more than case 1 as shown in figure below. In case: 1 when we split the data set, in both the data set the labels are distributed in in 50-50 ratio where in case: 2 labels are distributed in 90-10 ratio. | | |
|  | |  |
| * Information gain is defined in terms of entropy of data set which is defined as below. | | |
| Pi 🡺 Gini Index | | |
|  | | |
|  | | |
|  | | |
|  | | |
| * The attribute which is having higher information gain, that attribute can be consider for the splitting the data set. | | |
|  |  | |
|  | | |
| * For classification model, we used gini, entropy, information gain measure to classify the data set. * In the regression model, as the output is a continuous variable, we split the data in different combination of attribute and check the R^2 value for each model. If R^2 is not proper, then split further until get proper R^2. | | |
|  | | |
|  | | |

## Truncation and Pruning

|  |
| --- |
| **Advantage and Disadvantage:**                 * Based on Homogeneity measure, truncation can be controlled. No need to grow the tree till the end. * In the Pruning scenario, the complete decision tree is made and we cut the tree to certain level. |

### **Tree Truncation**

|  |  |
| --- | --- |
| **Different Truncation Criteria:** | |
|  |  |
|  | |
|  | |

### **Tree Pruning**

|  |
| --- |
| * One non-leaf node can be selected to cut the tree and same node can be allocated as leaf node. In the classification problem, each leaf defines a decision. Here as each leaf is having some decision and we are cutting all these leaves to single leaf; the decision value of the leaf will be **Majority of Merged Subset.** That means all the sub-set leaf which were cut, gather all the levels for them and take the Majority level.        * Take the tree and try pruning in each node in the tree. * Validation test accuracy before pruning and after pruning can be checked. * If after pruning the accuracy not worse than previous one (without pruning) * Keep doing the step until the validation performance dropping significantly. |
|  |

|  |
| --- |
| **Graded Questions: (Refer Python Notebook to Get the information about the data)** |

## Random Forest

|  |
| --- |
|  |

### **Creating Random Forest**

|  |  |  |
| --- | --- | --- |
|  | | |
|  | |  |
|  | |  |
| * Out of Bag Error is nothing but the test performance of the model. * As in random forest, we don’t do the train and test split, different model can take different sample of the data. Some data point which is not taken for one model becomes the test data for that model. Detail example of this given below. | | |
|  | | |
|  | | |
|  |  | |
|  | | |

# Model Selection Best-Practices

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | | |  |
|  | | |  |
| 1. **Logistic Regression:**      1. **Decision Trees**      1. **SVM** | | | |
|  | | | |
|  | |  | |
|  | | | |
|  | | | |
|  |  | | |
|  | | | |
|  | | | |
| * If categorical variables are multinomial then use CHAID. If categorical variables are binomial then use CART. | | | |
| **Choosing between Trees and Random Forests - I** | | | |
|  |  | | |
|  | | | |
|  | | | |
|  | | | |
|  | | | |

# Advanced Regression

## Generalized Linear Regression

|  |  |
| --- | --- |
| * In the sales figures we can see there are 2 kind of observation. Overall sales increases month on month. Also, there is a wavy pattern of sales. Cosine is tracking the wavy part where steady increase captured by other parameters.        * In the simple linear regression, we do the simple straight line. * In advance linear regression, using the parameter derive a quadratic function which can best fit to the kind of parameter distribution present. In above picture, we can see 1st one is linear mode. 2nd and 3rd one, we have used complex model. * We need to do the exploratory data analysis by plotting the parameter relation to understand what kind of equation can fit the line. | |
|  |  |
| * If raw attribute is taken is taken for prediction, then it may not give a proper result. * We can have derived column which is non-linear and can be used as feature for prediction. * Here in above example, only height or only weight can’t tell the blood sugar level. As Height is depends on weight and vice-versa. Hence derive column will give the better outcome. * Raw attributes are attributes which are not derived. * Raw attribute and derived attribute both can be used as feature. * From date parameter, we can derive the age of the customer. | |
| <https://www.youtube.com/watch?v=uQhTuRlWMxw&t=1s>  <https://www.youtube.com/watch?v=MC7l96tW8V8&list=PL39469144F25ACECE&index=3>  <https://www.khanacademy.org/math/linear-algebra/matrix-transformations/matrix-transpose/v/lin-alg-showing-that-a-transpose-x-a-is-invertible> | |
|  |  |
|  |  |
|  | |

## Regularized Regression

|  |  |
| --- | --- |
| **Ridge and Lasso Regression:** | |
|  |  |
| * Ridge regression brings a 2 major benefit * With regularization term and “Lambda” hyper parameter, we are able to control the overfitting of the model. * When we calculate “Alpha” (Co-Efficient) using above formula, we need to do inverse of then we need to find if is invertible or not. To make one term invertible, we need to have determinant of that term as 0. * As we get random dataset, making the dataset invertible is not possible directly. By adding “Lambda” error term which is an identity matrix, we are able to make inverse of the data point. This is another benefit in the Ridge regression after regularization.        * Lasso Regression helps to reduce the features by making the coefficient 0.      * In above picture, red contour is error term. “**Ridge**” regularization contour dotted circle around “alpha1” and “alpha2”. “Lasso” regularization contour is the square boxes whose edges are touches to axis of “alpha1” and “Alpha2” * As we need to find best “error term” and “regularization term”. For this we need to find the point where “error term” and “regularization term” touch each other “tangentially”.   **Tangentially means:**     * In Ridge and Lasso case, tangential touches shown as star in picture. For Lasso the tangent points are one of the axes of “Alpha”. This implies other “Alpha” value is 0. In this way lasso makes coefficient of some of the features 0. | |
| * More the value of R^2, better the model is. * If d goes up (Number of Predictors), n-d-1 is go down => RSS/n-d-1 goes up🡺 Adjusted R^2 goes down. * Basically, more features, penalized the Adjusted R^2.   **Feature Selection:** | |

# Boosting