INTERVIEW QUESTIONS WITH ANSWERS

You can contribute to the notebook (through Interview Q&As Unanswered - If you have any suggestions for this notebook: write your suggestions in the above link). This is the first version I will improve this notebook further in subsequent stages.

Host: Karthik Kumar Billa LinkedIn | GitHub (Research Intern)

Link for this book: OneStop4ML_Interview

LinkedIn Group: https://lnkd.in/gsS2B4g (Contributors to the notebook can be found in this group)

Course notebook: Available at AAIC's google drive | Comments

Main Content for the Notebook			**Relative coverage in Actual Interviews for the job role**				
	Topic	**Skill**	**Python Developer**	**Data Scientist**	**ML Engr**	**DL Engr**	**Rank of importance**
1	Communication , Presentation	Explaining things well	10	10	10	10	**3**
2	Programming Exercises	Python, Programming aptitude	30 15	15 10	10 10	10 10	**1**
3	Data Science	Data collection: SQL Data Engineering Data Visualization	20 5 10	15 15 15	10 10 10	5 10 10	**2**
4	Machine Learning	Math behind End to End workflow	 5	5 5	10 10	5 5	**4**
5	Deep Learning	Math behind End to End workflow	 5	5 5	5 5	10 10	**5**
6	Interview Experiences						
7	External Resources (Other Desirable skills: AWS, GCP, Tableau, Pytorch, Tensorflow 2.0)						

Quotes and Notes

- **"Your LinkedIn and Github activity brings you connections and jobs".** - Karthik
- **"Showcase your portfolio and prove that you are the solution for the problem".** - Karthik
- **"Don't leave jobs, AIML takes a lot of time".** - Karthik
- **"You need to own your work. I want to see your work, prove that through previous results."** - Interviewers
- **"Nobody will be given a job for model.fit and model.predict**." - Bardovv

From Investors:

- **"If I buy you a very fast and reliable racing car do you possess or develop the mindset of a car racer to be successful on the track. And if I buy you a business do you possess or develop the mindset of an entrepreneur to face business challenges. Business challenges are different from technical challenges".**
- **"Why do interviewers reject candidates? And why are there multiple rounds for a company for a job role? Every manager who is in direct contact with the job role is interested in knowing the skills of the candidate and assessing whether that candidate is good enough for his company. Say the departments are ML, DL, DS and Software developers, managers from each department assess whether you understand how these departments work and (importantly) to see what value you bring to the table or to the company. During this assessment people get rejected. (Insider insights)".**
- **"You need to have goals set. Goals should be specific, attainable, should have challenges and be meaningful. If a goal is complex, divide it into parts".**

Note: This notebook is an effort towards helping all of us get jobs in the field of Machine Learning, Data Science and Deep Learning. The purpose is to reduce deviations that we face when doing Google search. We need to have a set of answers ready for basic interview questions. (For AAIC students only) If you find any difficulty in understanding a topic, go through my notebook provided through AAIC google drive.

Importance of AAIC (Applied Course): The case studies and the hands on Machine Learning approach they have built are important. The things that we are covering here is just theoretical. We in no way touch or document the effort made by AAIC in case studies and assignments. It is their own intellectual property.

Note: In the actual industry: People work more on Data collection and preprocessing upto 80% of the time. SQL and Data Science skills are mostly desirable. Programming is the next desired skill. Machine Learning and Deep Learning are just a few lines of code (good to have an understanding, important to have for making analysis, but does not give any major advantage in Industry or in Interviews). There is absolute criteria in terms of number of years experience needed: Minimum 1. For freshers it is not that easy to get a job. If you have >=1 year experience, it is alright even if you are an average candidate, but if you are a fresher you need to improve your skills until you get a job.

“By concentrating on everything you will miss out on the most important things, by highlighting the whole page you have highlighted nothing. If someone asks to give a review of their work, give them a good response. And if someone points at your shortcomings, don’t feel bad it is just that your time is behind his time, but both have almost travelled the same journey. If someone is showing off, let them do it. **For you, you are the most important.** And remember corona made mankind think again. Will they learn?**”** After two weeks of panic mode I came back to normal by just taking a break from all the rat race. - Karthik

“It’s very easy to be tricked into believing that self-worth is tied to external labels of success. Nonsense. You do you, enjoy the moment, take stock of whom and what you have in your life, how you’re unique, and focus on what you can control. Everything takes time, and overnight success takes decades to happen. Never-mind that social media tries persuading you otherwise. Someone out there loves you and thinks **you’re awesome** (Think of your mom and your network). I do too.” - Simon Pouliot (LinkedIn Post)

Want to be a Data Scientist or Machine Learning Engineer or a Deep Learning Engineer?

Skills Required: Detailed Skill Requirements

- Programming: Programming Thinking and delivering code, comfortable with whiteboard,

  Tools: **Python, SQL, AWS, Tensorflow**, Tableau, NoSQL, MongoDB,
- Mathematics: Linear Algebra, **Calculus, Probability, Statistics**
- Data Analysis: **Explorations, visualizations, Data cleaning and preprocessing**
- Communication: **Explaining work and its results, Presenting visualizations, Proper Documentation, Neat Coding**
- Machine Learning, Deep Learning: **Algorithm knowledge, Model training and evaluation**
- Productionization: **Deploy**ing best model into real world for user interaction
- Self case studies (Portfolio): Showcase **end to end solutions** to real world problems (from data collection/web scraping to deployed maintenance)

A FRESHER TAKES A YEAR TO COMPLETE 25% OF THE ABOVE PROCESS!

Check where you are:

Programming + Communication (Data Exploration) + Math: **Data Analyst**

Programming + Data Engineering + Math + Machine Learning: **Machine Learning Engineer**

Programming + Data Engineering + Communication + Math + Productionization: **Data Scientist**

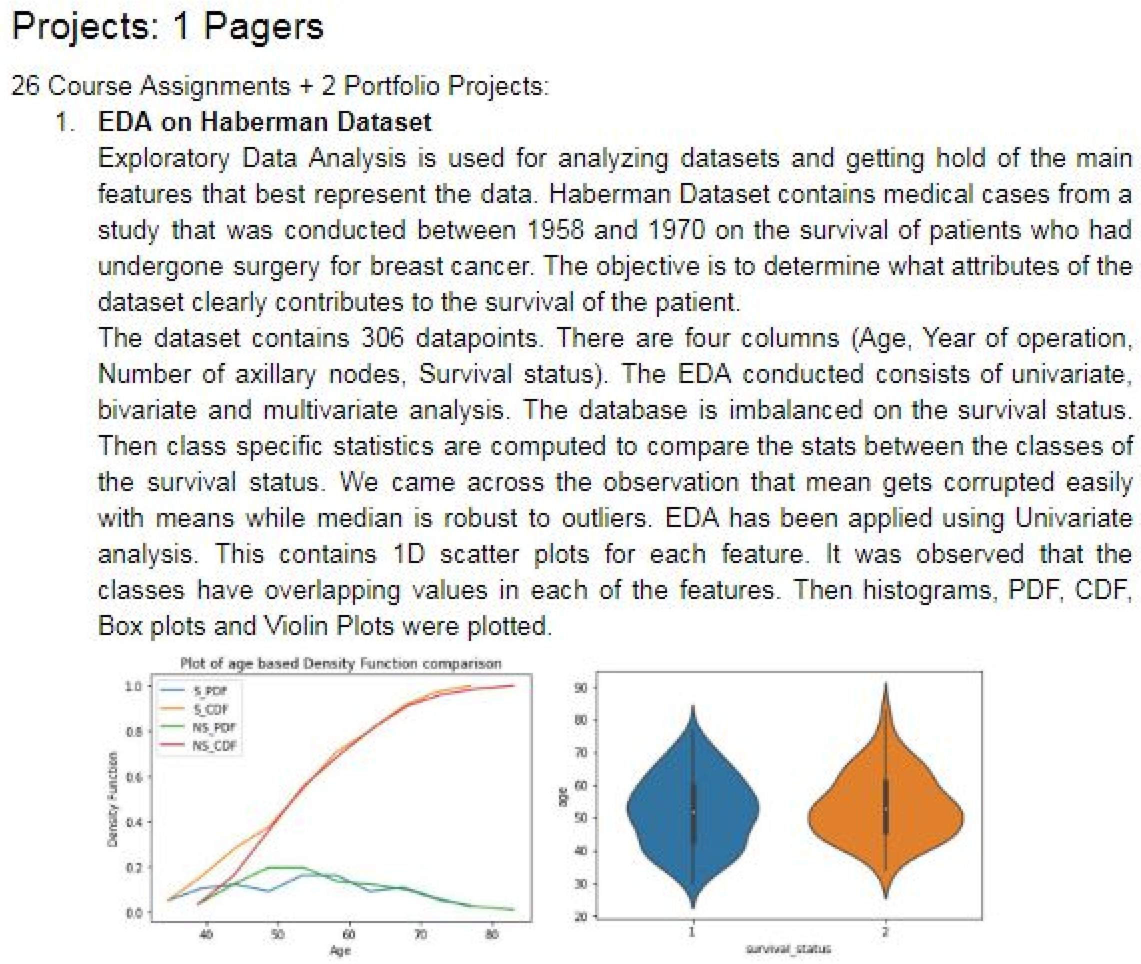
Main Advantage of taking Data Science or Machine Learning is to **Transform Data into Achievements** (it helps you take guided decisions in every situation).

Mistakes to be avoided

1. Too much theory (have a balance in practical and theory).
2. Jumping into ML without learning Math.
3. Ignoring the working of models (you need to start by building simpler base models).
4. Implementing from scratch.
5. Domain knowledge needs to be applied to **define the business problem** and apply **Exploratory Data Analysis** to narrow down on suitable ML algorithms.
6. Learning Advanced algorithms without knowing classical algorithms.
7. Not practicing EDA (75% of the work in industrial projects lies not in Machine Learning algorithms rather it lies in Problem definition, Data collection, Data visualization and data preparation, Model deployment and maintenance (re training the model with varying trends in the datasets is another important topic).
8. Forgetting about **data leakage**.
9. Non sequential and unstructured problem solving or code development.
10. Inconsistent for a long period of time.
11. Neglecting **story-telling** is a mistake.
12. Having a favorite algorithm is a mistake.
13. Neglecting hyperparameter tuning is a mistake.
14. Changing tools frequently is also a mistake.

Write your own One Pagers:

It is very much beneficial to write a one page summary on any of the topics that you have covered just after reading it. You can compile it and also blog. A one pager summary of assignment will aid in your interview preparation. I suggest you add one pager summaries to your portfolio. For example:



Projects: 1 Pagers

26 Course Assignments + 2 Portfolio Projects:

1. **EDA on Haberman Dataset**
   Exploratory Data Analysis is used for analyzing datasets and getting hold of the main features that best represent the data. Haberman Dataset contains medical cases from a study that was conducted between 1958 and 1970 on the survival of patients who had undergone surgery for breast cancer. The objective is to determine what attributes of the dataset clearly contributes to the survival of the patient.
   The dataset contains 306 datapoints. There are four columns (Age, Year of operation, Number of axillary nodes, Survival status). The EDA conducted consists of univariate, bivariate and multivariate analysis. The database is imbalanced on the survival status. Then class specific statistics are computed to compare the stats between the classes of the survival status. We came across the observation that mean gets corrupted easily with means while median is robust to outliers. EDA has been applied using Univariate analysis. This contains 1D scatter plots for each feature. It was observed that the classes have overlapping values in each of the features. Then histograms, PDF, CDF, Box plots and Violin Plots were plotted.

This will help you showcase your work efficiently. You can build a powerpoint presentation. Remember to showcase results and let the recruiters crave for those results.

General Interview Rounds

Interview Structure

Rounds: 2 to 6

Round 1: Take home assignments or MCQs or Phone Screens

Round 2: Take home Assignment based Video call interview and/or Phone Screening

Round 3: ML and DL Technical round with one of the company engineers

Round 4: Programming round with one of the company engineers

Round 5: Hiring Manager Round

For people who want to read ML in detail: start from this book: Hands on ML by Aurelien Geron

The things you need to be ready with on the day of your interview:

Checklist:

1. Knowledge of the company (and its challenges) you are going to interview with.
2. What do you think the company will help you grow in (the company does not know about how it can help you develop, you need to tell them how)?
3. Be ready to have a 30-60-90 days schedule. Also have a 3 to 5 year plan ready.
4. Be ready to showcase your strengths, previous achievements and project walk throughs.
5. Be ready to explain how you became victorious from a failure.
6. Be ready to tell a story about you in less than 3 minutes. Draft a **one-pager** (a template will be developed for an example company).
7. Be ready with some of the questions from this notebook.

Template for Resume (2 pages) - After writing the following details you can reformat your resume as in Link:

Name

Contact

Mail

Location

Github

Linkedin

Blog

Professional Summary: Adjectives + objective

Skills:

Programming languages:

Tools:

Area of Interest:

Experience:

Company, Location

Designation, Responsibilities: Accomplished [X] as measured by [Y] by doing [Z]

Projects:

Objective, responsibility, process, outcome

Education:

College, location, year, branch

Achievements

HR Questions

1. Tell me about yourself.
2. Why are you interested in this job?
3. Why do you want to apply to our company?
4. What do you know about the company?
5. How did you hear about the job?
6. Why did you leave your previous job?
7. Why are you currently unemployed?
8. What did you do this year?
9. What are your strengths?
10. What is your weakness?
11. What is your greatest achievement?
12. Do you have a work style?
13. How would your colleagues describe you?
14. Do you prefer working in a team or independently?
15. How do you deal with work stress?
16. What is your expected Salary?
17. What other roles are you interested in?
18. Where do you see yourself in five years?
19. What do you do in your free time?
20. What is your ideal work environment?
21. How will you help this company grow?
22. What else should we know about you? Why are you suitable?
23. If we give a rating 6 out of 10 on your skills, how will you prove to us that you are 10 for this job?
24. Can you walk me through your portfolio projects?
25. If you have the power to change anything, in 10 seconds tell me what you want to change in the next 24 hours?
26. What machine learning algorithm do you think you are?

Basic Mathematics

Linear Algebra

1. Define Point/Vector (2-D, 3-D, and n-D)?

Point: An element of a space or a location in space. It is denoted by n coordinates in an n-dimensional space.

A point or a vector is defined by a set of scalars which indicate the location under consideration in a space.

Point/Vector: $[x_1, x_2, \ldots, x_n]$

Distance between two points:

$$d = [(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + (x_{13} - x_{23})^2 + (x_{14} - x_{24})^2 + \ldots + (x_{1n} - x_{2n})^2]^{1/2}$$

2. How to calculate Dot product and angle between 2 vectors?

Let a and b be two vectors where:

$a = [a_1, a_2, \ldots, a_n]$

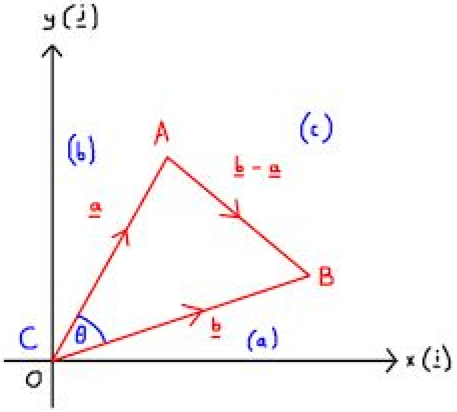
$b = [b_1, b_2, \ldots, b_n]$

a and b be should be of same dimensionality;

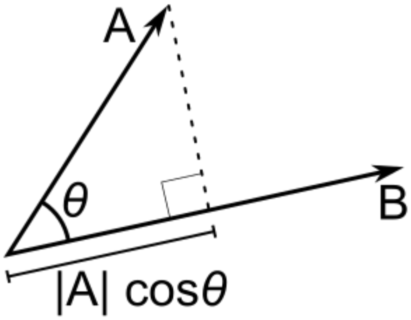
Dot_product(a, b) = $a.b = a_1b_1 + a_2b_2 + \ldots a_nb_n$

Angle $\theta_{ab} = \cos^{-1}[a.b/||a||\ ||b||]$

$$||b|| = [b_1^2 + b_2^2 + \ldots b_n^2]^{1/2}$$



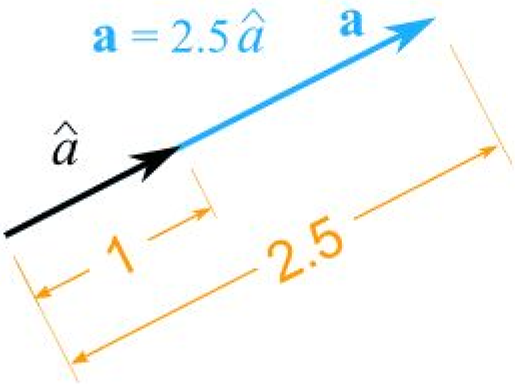
3. Define Projection, unit vector?



A projection is a mapping of a set into a subset. With projection we transform a vector onto another vector. The transformation allows us to compare the two vectors.

projection(a on b) = a $\cos(\theta_{ab})$ = a.b/||b||

A unit vector is a vector which has a length of 1. ||â|| = 1



4. Equation of

a. Hyperplane (n-D): $a_0 + a_1x_1 + a_2x_2 + \dots a_nx_n = 0$;

  i. distance of a point($[p_1, p_2, p_3, \dots, p_n]$) from a HyperPlane = $a_0 + a_1\ p_1, a_2\ p_2, a_3\ p_3, \dots, a_n\ p_n,$ = abs(a^T.p)/||a||

b. Hypersphere: $[x_1^2 + x_2^2 + \dots x_n^2]^{1/2} = r^2$

c. Hyper ellipsoid (n-D): $[(x_1/a_1)^2 + (x_2/a_2)^2 + \dots (x_n/a_n)^2]^{1/2} = 1$

d. Hyper-cuboid: $x_1\ \varepsilon\ [a_1, b_1], x_2\ \varepsilon\ [a_2, b_2], x_1\ \varepsilon\ [a_3, b_3], \dots$

Probability and Statistics

1. What are Random variables: discrete and continuous?

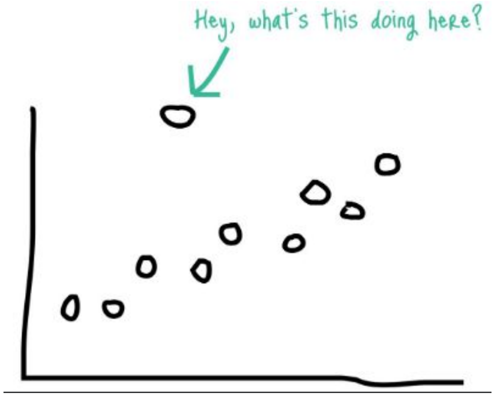
A random variable is a mapping that assigns a real number X(w) to each outcome w. Random variables are used to quantify outcomes of a random occurrence, and therefore, can take on many values. Random variables are required to be measurable and are typically real numbers. Random variables generate outcomes different every time when the experiment is repeated even if the conditions are identical.

A ***discrete random variable*** is one which may take on only a countable number of distinct values such as 0,1,2,3,4, ... Discrete random variables are usually (but not necessarily) counts. If a random variable can take only a finite number of distinct values, then it must be discrete. Examples of discrete random variables include the number of children in a family, the Friday night attendance at a cinema.

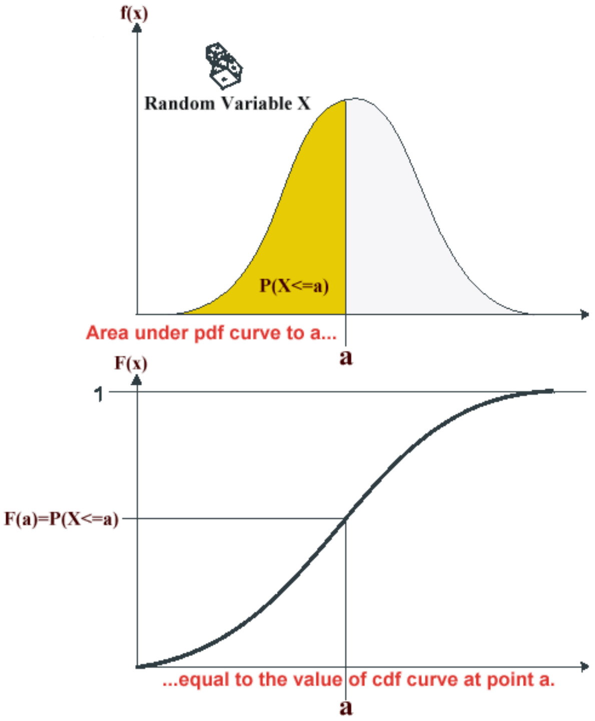
A ***continuous random variable*** is one which takes an infinite number of possible values. Continuous random variables are usually measurements. Examples include height, weight, the amount of sugar in an orange.

2. Define Outliers (or) extreme points.

An **outlier** is a data point that differs significantly or that appears to be inconsistent from other observations in the datasets. An outlier may be due to variability in the measurement, it may indicate experimental error or may be a genuine observation. An outlier can cause serious problems in statistical analyses.



3. What is PDF?


f(x)
Random Variable X
P(X<=a)
Area under pdf curve to a...
a
F(x)
1
F(a)=P(X<=a)
...equal to the value of cdf curve at point a.
a

Probability density function (PDF) is a statistical expression that defines a probability distribution (the likelihood of an outcome) for a continuous random variable. PDF for an interval indicates the probability of the random variable falling within the interval.

4. What is CDF?

The cumulative distribution function at any value x gives the probability of the random variable X, taking a value from negative infinity up to x and is defined by the following notation:

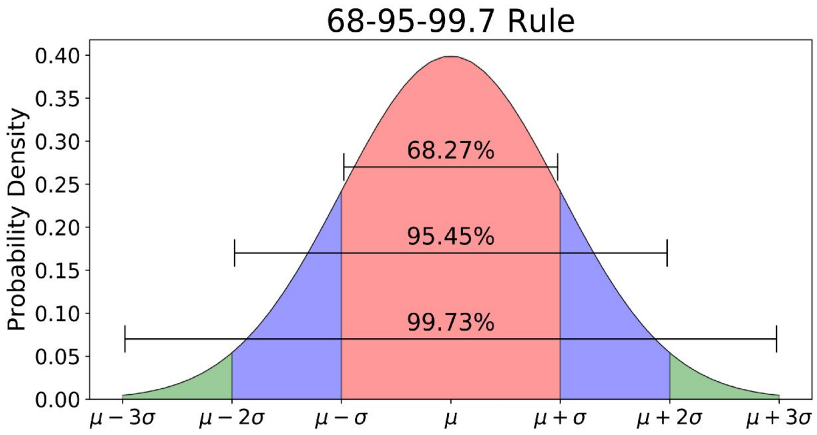
$$F(x) = P(X \leq x)$$

CDF completely describes the distribution of a random variable.

5. Explain about 1-std-dev, 2-std-dev, 3-std-dev range?

The normal distribution is commonly associated with the 68-95-99.7 rule which you can see in the image below. ~68% of the data is within 1 standard deviation (σ) of the mean (μ), ~95% of the data is within 2 standard deviations (σ) of the mean (μ), and ~99.7% of the data is within 3 standard deviations (σ) of the mean (μ).

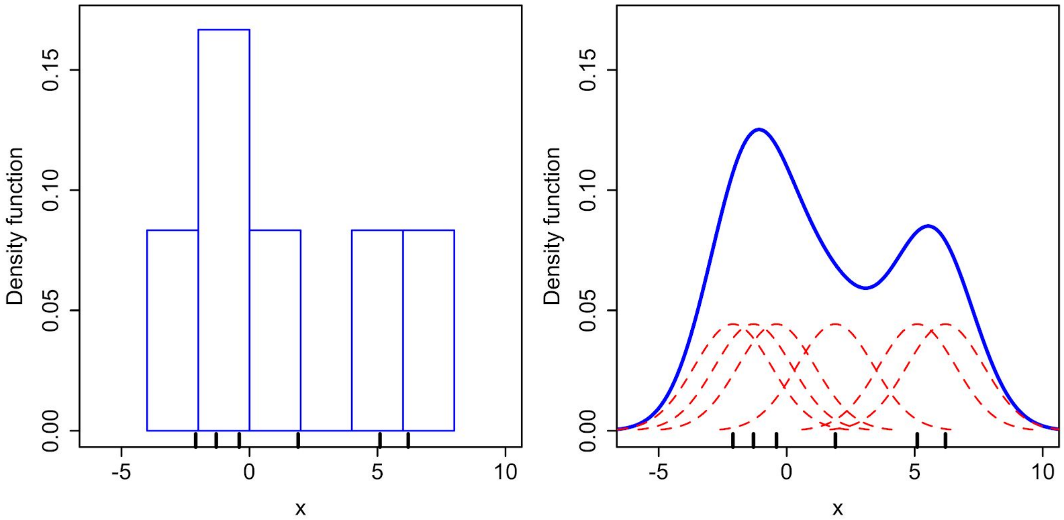
Standard deviation: Measure of spread in statistics: $\sigma = (\Sigma(x_i – \mu_x)^2)^{1/2}$



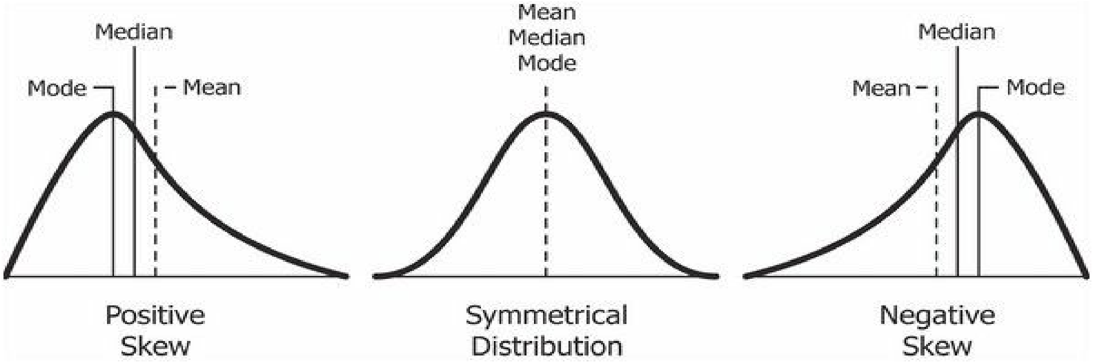
6. What is Kernel density estimation?

It is a non-parametric technique to estimate the unknown probability distribution of a random variable, based on a sample of points available from that distribution. At every data point in the sample we place a normal kernel. The kernels are then summed up to provide the pdf of the population.

Kernel density estimation converges faster to the true underlying density function for the continuous random variable. This is a better approximation of the density function of the population in comparison to a histogram.



7. What is Symmetric distribution, Skewness and Kurtosis?



Symmetric distribution: if the PDF of the distribution has a mirror image about the mean then the distribution is symmetric over the mean. $F(x_0 + h) = F(x_0 - h)$

Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean.It measures the lack of symmetry in data distribution. A symmetrical distribution will have a skewness of 0.

$$\tilde{\mu}_3 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{(\mathrm{E}[(X-\mu)^2])^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}}$$

Kurtosis a measure of the "tailedness" of the probability distribution of a real-valued random variable. Like skewness, kurtosis describes the shape of a probability distribution. It is not a measure of peakedness of the curve. This provides an idea of outliers in the distribution.

$$\mathrm{Kurt}[X] = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2} = \frac{\mu_4}{\sigma^4}$$

Smaller the kurtosis the better the distribution is, that is we will have less outliers. Kurtosis of a Gaussian random variable is 3. To compare Kurtosis of a distribution with the Gaussian kurtosis we use **excess kurtosis** which equals kurtosis - 3.

For a sample of *n* values the **sample excess kurtosis** is

$$g_2 = \frac{m_4}{m_2^2} - 3 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^4}{\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right]^2} - 3$$

Additional: Standardized moment:

$$\mu_k = \mathrm{E}\left[(X-\mu)^k\right] = \int_{-\infty}^{\infty}(x-\mu)^k P(x)\,dx$$

k	Expression	Comment
1	$\tilde{\mu}_1 = \frac{\mu_1}{\sigma^1} = \frac{\mathrm{E}\left[(X-\mu)^1\right]}{(\mathrm{E}[(X-\mu)^2])^{1/2}} = \frac{\mu-\mu}{\sqrt{\mathrm{E}[(X-\mu)^2]}} = 0$	First moment about mean is zero
2	$\tilde{\mu}_2 = \frac{\mu_2}{\sigma^2} = \frac{\mathrm{E}\left[(X-\mu)^2\right]}{(\mathrm{E}[(X-\mu)^2])^{2/2}} = 1$	Second moment about mean is variance itself
3	$\tilde{\mu}_3 = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{(\mathrm{E}[(X-\mu)^2])^{3/2}}$	Third moment gives skewness of the distribution
4	$\tilde{\mu}_4 = \frac{\mu_4}{\sigma^4} = \frac{\mathrm{E}\left[(X-\mu)^4\right]}{(\mathrm{E}[(X-\mu)^2])^{4/2}}$	Fourth moment gives kurtosis of the distribution

8. How to do Standard normal variate (z) and standardization?

A standard normal variate is a Gaussian Normal distribution with mean 0 and variance 1. It is a random variable with expected value 0 and variance 1.

$$Z \sim N(0,1)$$

We can convert a normal distribution into a standard normal variate by using standardization. In standardization we perform mean centering and scaling.

$$X = (X - \mu) / \sigma$$

We convert a random variable into a standard normal variate to understand the characteristic of the distribution of the variable. Characteristics of a standard normal variate are available as standard tables which can be used to make inferences of the distribution characteristics. This will make analysis of the random variable feasible. While two random variables under comparison need to be standardized to make meaningful comparisons. Standardization brings the distribution of the two random variables to the same scale with mean 0 and variance 1.

We can easily estimate the 90th percentile and other statistics of the distribution when we have the random variable in the form of a standard normal variate.

9. Importance of Sampling distribution & Central Limit theorem.

Sampling distribution: It is the probability distribution of a given random-sample-based statistic. If an arbitrarily large number of observations each involving multiple observations (drawn independently and with replacement) were separately used to compute a statistic for each sample.

Sampling Distribution is the probability distribution of the values a statistic of a population takes on when an independent multiple samples are separately used to compute the statistic value (sample mean or sample variance).

Procedure:

- Let X be any distribution of a population.
- Pick ‘m’ random samples independently of any large size, say n each (same or different) $S_1, S_2, \ldots, S_m$
- For all samples, compute mean:$S_{1\mu}$, $S_{2\mu}$, …, $S_{m\mu}$. These means tend to follow a normal distribution as ‘n’ increases irrespective of the type of population distribution. This distribution of sample means is called Sampling distribution of sample means.
- As b increases, mean of sample means reaches population mean and variance of sampling distribution reaches population variance divided by n.
- Sampling distribution: $X \sim N(\mu, \sigma^2/n)$; is used to make inferences about population statistics based on finite samples. Samples are drawn from the population independently with replacement.

Central Limit Theorem: The **central limit theorem** states that if you have a population with mean μ and standard deviation σ and take sufficiently large random samples from the population with replacement , then the distribution of the sample means will be approximately normally distributed.

Let X be a population with mean μ and standard deviation σ. If we generate a sampling distribution of the sample statistic then the sampling distribution will be approximately normally distributed when the sample size is large. In addition, mean or any statistic of sampling distribution reaches population statistics as n increases.

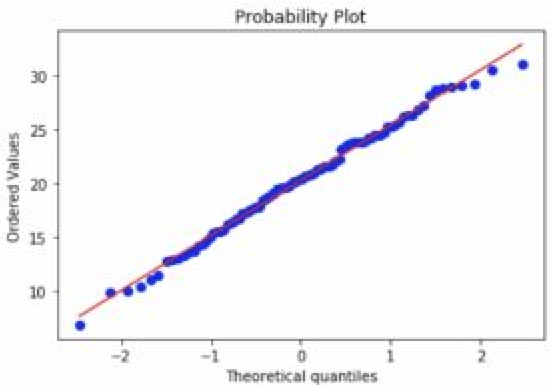
This allows us to estimate population statistics when only finite samples are available. If a single sample is available, we generate multiple samples from this single sample.

10. Importance of Q-Q Plot: Is a given random variable Gaussian distributed?

Quantile-Quantile plot is used to compare the distributions of two numeric variables. The comparison is done by plotting the quantiles of two sets. Generally we compare a variable between two groups(Ex: comparing the distribution of expenditure among Males, Females). So mostly in the Data Science community, this is used to check the distribution of numeric variables that are behaving among two groups. Another major application is to validate whether a variable is following a particular distribution or not. In this case, we take a numeric variable and compare this with the theoretical distribution.

Procedure:

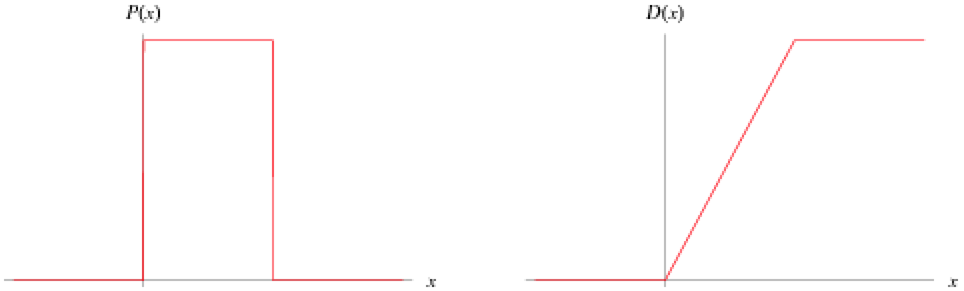
- Let Y be the random distribution as for X we will take Gaussian Normal distribution to compare the Y variable. Generate X as a Gaussian Normal random variable.
- Sort X and Y values and compute their percentile values.
- Plot them to infer for the similarity in the two distributions. A straight line plot indicates that the X and Y have similar distributions and as X is Gaussian, Y can be interpreted to follow a Gaussian Normal distribution.
- The number of observations in both X and Y should be sufficiently large to make a valid comparison.



11. What is Uniform Distribution and random number generators?

Uniform distribution is a symmetric probability distribution. It is defined by a minimum and maximum values of the distribution and all the values of the distribution lie between these max and min values.

A random number generator generates numbers that cannot be reasonably predicted better than by a random chance. It is used to simulate natural generation of random numbers. Random numbers are generally generated from a Uniform distribution as in a Uniform distribution selecting any random value is equally probable.

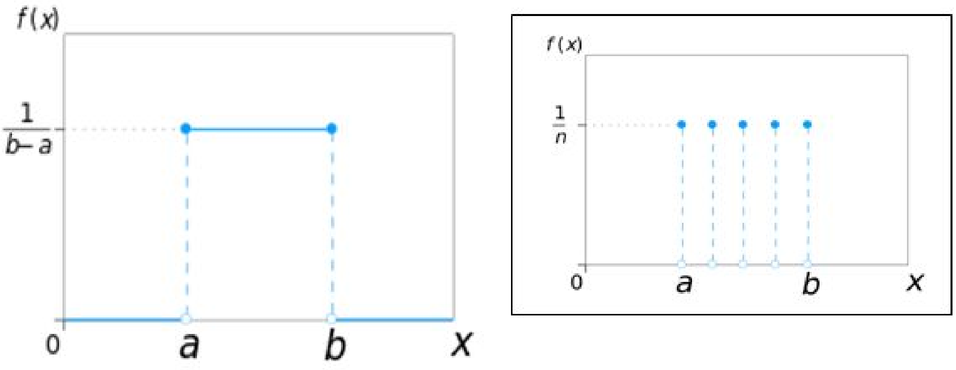


PDF and CDF of a Uniform distribution

12. What are Discrete and Continuous Uniform distributions?

Discrete Uniform distribution has the probability of all outcomes equally likely and with finite values. Example: Rolling of a 6-sided die. Application: Inventory or resource management.

Continuous Uniform distributions have an infinite number of equally probable outcomes. Application: Random Number generator.



(a) Continuous Uniform Distribution

(b) Discrete Uniform Distribution

13. How to randomly sample data points?

Data points can be sampled with or without replacement based on the situation. To have a random sampling we need to have equal probability of selecting any datapoint. The following procedure is applied for a simple random sampling method.

- Choose sample size k.
- Assign each data point a number from 1 to N.
- Generate k random numbers and select data points corresponding to the generated numbers (index).

We can also have a stratified random sampling of data for imbalanced datasets. In addition to simple sampling we break the population into groups and apply simple random sampling on subgroups to preserve the same distribution characteristics from population to sample.

Simple Random sampling for imbalanced datasets is not useful and generates samples that can contain characteristics different from the population. A random sampling is expected to be an unbiased representation of the population.

14. What is Confidence Interval?

A Confidence Interval is used when estimating an unknown parameter of the population from the sample data.The interval gives the range of plausible values for an unknown parameter, and a confidence level that the range covers the truth.

Confidence Interval: An N% confidence interval for parameter p is an interval that includes p with probability N%

For eg - I am 95% confident that the population parameter(sigma) calculated using the sample is between [3,9].

15. Explain about Hypothesis testing?

A statistical hypothesis is an assumption about a population parameter. This assumption may or may not be true. Hypothesis testing refers to the formal procedures used by statisticians to accept or reject statistical hypotheses. Based on significance level we either reject the null hypothesis or fail to reject the null hypothesis in favor of alternative hypothesis.
A null and an alternative hypothesis are defined based on the assumption about the parameter.

16. Define Hypothesis Testing methodology, Null-hypothesis, test-statistic, p-value?

Hypothesis Testing Methodology:

- Choose a test statistic
- Define null and alternative hypothesis
- Compute p-value of significance: p-value indicates the probability of observing test statistic if null hypothesis is true. If p-value is less than threshold level we reject null

hypothesis in favor of alternative hypothesis. If p-value is more than threshold level we fail to reject null hypothesis.

Null Hypothesis - It is the hypothesis of no difference. It corresponds to the idea that the observed difference is due to chance.

Test Statistic - Test Statistic is used to measure the difference between the data and what is expected in the null hypothesis

P-value - It is the chance(probability) of getting a test statistic as extreme or more as the observed one assuming the null hypothesis is true. The smaller P-value(<=0.05) , the stronger the evidence against Null Hypothesis.

17. What does P-value signify about the statistical data?

P- value (observed significance level) is the chance of getting a test statistic as extreme or more as the observed one, assuming the null hypothesis is right. A small P-value (typically 0.05 or less) is the evidence against null hypothesis. We say that the P -value is significant meaning it is small enough that we can safely reject null and a large P-value suggests in the failure of rejecting the null hypothesis.

18. What do you understand by Hypothesis in the content of Machine Learning?

A statistical **hypothesis** is an explanation about the relationship between data populations that is interpreted probabilistically. A machine learning **hypothesis** is a candidate model that approximates a target function for mapping inputs to outputs.

Simple example - Consider Linear regression, the first step is to check if there is any linear relation between X and Y(let Y be sales of and X is advertisement on TV). So we want to check if any positive or negative change in Advertisement on TV can increase or decrease the sales.We can use Hypothesis testing here. A simple linear reg. equation looks like Y = a +bX where a is intercept and b is the slope.

So the Null Hypothesis H0 : there is no linear relationship between X and Y, that is b = 0.

The change in advertisement does not affect the sales of mobile

Alternative Hypothesis H1 : there is linear relationship between X and Y, that is b !=0

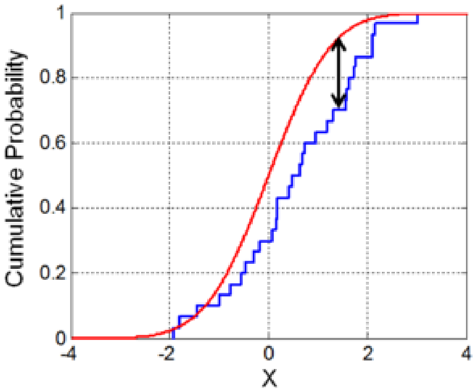
The change in advertisement affects the sales of mobile

Our test statistic here is b=0; if an observed value is taken into consideration, we will have p-value equal to probability of having the observed value if b = 0; if the p-value is greater than threshold value(set by the user 5% generally for Advertisements-Sales problem) then we will fail to reject null hypothesis; if p-value is less than the threshold value then we reject the null hypothesis in favor of alternative hypothesis.

19. How to do K-S Test for similarity of two distributions?

Kolmogorov-Smirnov test: Do two random variables X1 and X2 follow the same distribution?

- We plot CDF for both random variables; We will be using hypothesis testing;
- Null hypothesis: the two random variables come from same distribution;
- Alternative hypothesis: both don't come from same distribution;
- Test statistic; D = CDF(X1) – CDF(X2) throughout the CDF range;
- = supremum (CDF(X1) – CDF(X2))
- = max of (CDF(X1 distribution) – CDF(X2 distribution))
- (at same value on x axis)
- Null hypothesis is rejected at level α, when D > c(α) * sqrt((n+m)/nm)
- c(α) = sqrt(-0.5*ln(α/2))
- D > (1/n0.5)* (sqrt(-0.5 * ln(α) * (1+ (n/m)))



KS Test description

20. What are the conditions for a function to be a probability mass function?

A probability mass function (pmf) is a function over the sample space of a discrete random variable X which gives the probability that X is equal to a certain value.

Let X be a discrete random variable on a sample space S. Then the probability mass function f(x) is defined as

f(x)=P[X=x].

Each probability mass function satisfies the following two conditions: for all x in the sample space, f(x) is never negative and its sum over the entire sample space will always be 1. The random variable should be a continuous random variable.

21. What are the conditions for a function to be a probability density function?

A probability density function (pdf) is a function over the sample space of a continuous random variable X which gives the probability that X is equal to a certain value.

Let X be a continuous random variable on a sample space S. Then the probability density function f(x) is defined as

f(x)= ∫p(x) dx

Each probability density function satisfies the following two conditions: for all x in the sample space, f(x) is never negative and its sum over the entire sample space will always be 1. The random variable should be a continuous random variable.

22. What is conditional probability?

Probability of an event occurring given that another event has already occurred is called a conditional probability.

Given two events A and B, the conditional probability of event A given B has occurred is given as

P(A|B) = P(A ∩ B) / P(B)

And unconditional probability of event B occurring should be greater than 0.

Example: Consider a 6-sided dice roll.

P(A=6) = 1/6

P(B=6) = 1/6

P(A+B = 6) = 5/36

P(A = 3 | A+B = 6) = 1/5 = set([(3,3)])/set([(1,5), (2,4), (3,3), (4,2), (5,1)])

P(A =3 | A+B<=6) = 3/15 len(set([(3,1), (3,2), (3,3)])) / set([A+B<=6])

23. State the Chain rule of conditional probabilities?

If A and B are two events in a sample space S, then the conditional probability of A given B is defined as P(A|B)=P(A∩B)/P(B), when P(B)>0.

24. What are the conditions for independence and conditional independence of two random variables?

Events A and B are independent if knowing that A has happened does not tell about whether B happened.

Events A and B are conditionally independent given a third event C when we already know C has occurred and knowing whether A happened would not convey any further information about whether B happened. Independence also does not imply conditional independence.

Example: Flip two coins. Let A be the event of getting heads on the first toss, B be getting head on the second toss and C be getting the same outcome on both the flips. A and B are clearly independent. But if C happens it means that A and B happened, thus making them conditionally dependent. Given that C has happened, if we know that A happens then it means that B has happened.

25. What are expectation, variance and covariance?

Expectation: Average value of a random variable X. It is calculated as a probability weighted sum of values of the random variable.

$$E[X] = \Sigma(prob(x_i) * x_i)$$

Variance: A measure that describes the amount of variation of the distribution on average with respect to the mean.

$$Var[X] = \Sigma(x_i - E[x])^2$$

Co-variance: Measure of joint probability of two random variables; how do the two random variables change together.

$$Cov(X, Y) = E[(X – E[X]), (Y – E[Y])]$$

$$= \Sigma (x_i - E[x])(y_i - E[Y])$$

$$Cov(X, X) = Var[X]$$

26. Compare covariance and independence?

Cov(X, Y) = 0 does not guarantee that X and Y are independent. But if X and Y are independent then cov(X, Y) = 0

Example:

$X = \{-1, +1\}$, $P(X = -1) = P(X = 1) = 0.5$

$P(Y = 0 \mid X = -1) = 1$

$P(Y = -1 \text{ or } Y = +1 \mid X = 1) = 0.5$

Clearly Y is dependent on X.

$cov(X, Y) = sum(x_i\ y_i\ prob(y_i|x_i))$

$= -1x0xP(X=-1) + 1x1xP(Y=1|X=1) + 1x(-1)xP(Y=-1|X=1) = 0 + 0.5 - 0.5 = 0$

cov(X, Y) = 0 also happens for X and Y dependent

Independence: If knowing X does not give any information on Y, then X and Y are independent

Covariance: amount of change in variables together

27. What is Kullback-Leibler (KL) divergence?

KL divergence is the measure of difference between two probability distributions over the same variable. KL divergence of Q from P is a measure of information lost when Q is used to approximate P.

$D_{KL}(P||Q) = \Sigma(P \log(P/Q))$

And P(x) = 0 when Q(x) = 0 and when P = Q, KL divergence is 0.

KL divergence is non negative.

28. Can KL divergence be used as a distance measure?

KL divergence measures the distance but it cannot be used as a distance measure. It is not a metric measure and is not symmetric.

$D_{KL}(P||Q) != D_{KL}(Q||P)$

29. How to sample from a Normal Distribution with known mean and variance?

Sample from standard normal variate and multiply all values with standard deviation and add mean.

$X \sim N(0,1)$; x is a sample drawn from X; $\sigma x + \mu$ will be a sample drawn from $N(\mu, \sigma)$;

Generate random numbers from uniform distribution and then use Box-Muller transformation for generating independent standard normal distribution. Doing mean shift and scaling we will transform Standard Normal Distribution generated from Box-Muller transformation to Normal distribution with desired mean and variance. (We multiply standard deviation and add mean.)

For Box - Muller transformation the input is two independent uniformly distributed random variables U1, U2. The output of the transformation will be two independent standard normally distributed random variables X and Y.

$X = R \cos\theta$; $Y = R \sin\theta$; $R = \sqrt{(-2 \ln U_1)}$; $\theta = 2\pi U_2$

X is a standard normal variable $\sim N(0,1)$; $\sigma X + \mu$ will be a random variable $\sim N(\mu, \sigma)$

30. Correlation vs Causation?

Correlation is the measure of relationship between two random variables. Causation or Causality is the effect of one random variable over the other. If there's a causal relationship, it would mean A causes B. Number of Eggs laying chickens (A) and No. of Eggs produced(B) by a Chicken farm can be both positively Correlated and Causal. The number of Nobel prizes won (per 10 Mil) A by a Country and Chocolate consumption (kg/yr/ per capita) B are positively correlated but not Causal. **"Correlation doesn't imply Causation."**

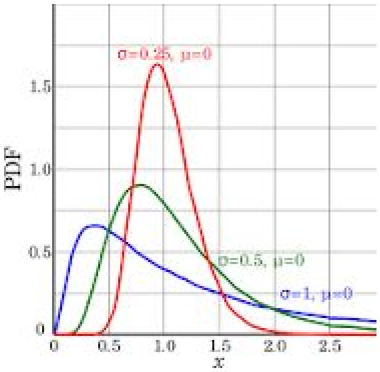
31. Explain about Bernoulli and Binomial distribution?

Bernoulli: discrete probability distribution of a random variable which takes values 1 with probability p and value 0 with probability 1-p: Coin toss:{H, T}, P = {0.5, 0.5};

Binomial: If X is Bernoulli thenY is binomial when Y = n times X; parameters: n, p: An event with n trials with success probability p in each trial

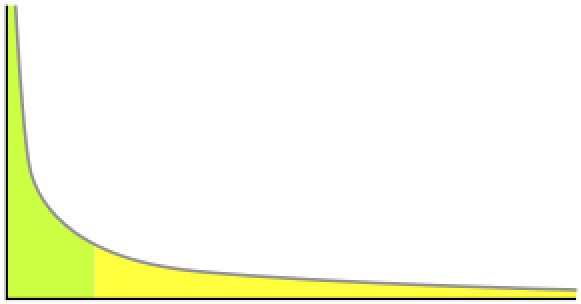
When n=1, Binomial distribution = Bernoulli distribution

32. What is Log-normal and power law distribution?



Log Normal: A continuous probability distribution of a random variable whose logarithm is normally distributed; In user reviews: most of comments are of small length and some of comments have large length of words;

We can employ QQ plot to check similarity of log of distribution to normal distribution;



Power law distribution: a **power law** is a functional relationship between two quantities, where a relative change in one quantity results in a proportional relative change in the other quantity, independent of the initial size of those quantities: one quantity varies as a power of another. For instance, considering the area of a square in terms of the length of its side, if the length is doubled, the area is multiplied by a factor of four.

Green area has 80% values; In bottom 20% of values you can find 80% of mass;

When a distribution follows a power law then the distribution is called Pareto distribution;

Parameters: scale and shape

Applications: allocation of wealth in population, sizes of human settlements;

We can also use a QQ plot against Pareto plot to check whether a given distribution follows a power law;

33. Explain about Box-Cox/Power transform?

A **Box Cox transformation / Power transform** is a way to **transform** non-normal dependent variables into a normal shape. Normality is an important assumption for many statistical techniques; if your data isn't normal, applying a **Box-Cox** means that you are able to run a broader number of tests. Applications: wavelet analysis, statistical data analysis, medical research, modeling of physical processes, geochemical data analysis, epidemiology and many other clinical, environmental and social research areas.

$$y_i^{(\lambda)} = \begin{cases} \dfrac{y_i^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln y_i & \text{if } \lambda = 0 \end{cases}$$

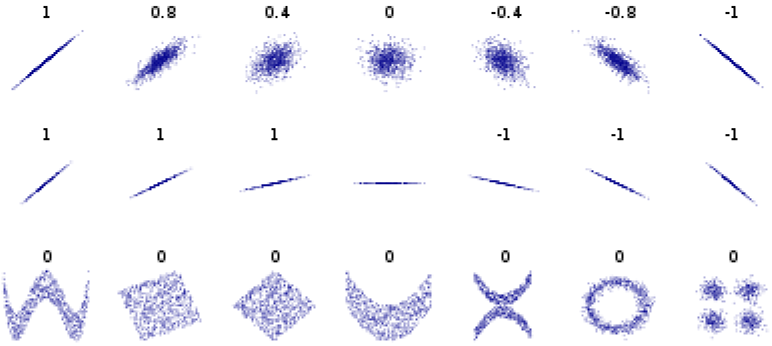
34. Explain about the box cox transformation in regression models.

The target variable from real world datasets in regression analysis might not satisfy the assumptions of regression analysis such as normality or the assumptions of ordinary least square regression. The error or the residuals may follow a skewed distribution which never allows to have an optimized regression model. This skewed distribution leads to bias. Through box cox transformation the variability across the distribution is made constant.

The models result in predictions that are generally normally distributed. To make comparison of the predictions with the target variable, the target variable needs to satisfy the assumptions of the regression models.

35. Importance of Pearson Correlation Coefficient?

It is a measure of strength between two variables. It is used to investigate the relationship between two continuous quantitative variables. It can take a value from -1 to +1, where -1 implies strong negative correlation, +1: Strong positive correlation and 0: no correlation. It is limited to determining linear correlations.



$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

36. Importance Spearman Rank Correlation Coefficient?

Spearman Rank Correlation Coefficient is determined by applying Pearson Coefficient on rank encoded random variables. It assesses the relationship between two random variables using a monotonic function. The sign of correlation indicates the direction of association between the two random variables. Spearman Rank Correlation of value +1 implies that the two Random Variables are having a perfectly monotone increasing relationship. (for all i and j X_i - X_j and Y_i – Y_j have the same sign).

37. Confidence Interval vs Point estimate?

Point estimate gives a particular value from a sample as an estimate for the parameter of the population. Point estimates generally generate bias in terms of difference between estimate and true value of the population parameter. Confidence interval provides an estimate for the population parameter in terms of range in which the parameter could lie with a confidence level of percentage. This reduces the bias which can be induced due to point estimate. A small interval with high confidence is desirable.

38. What is the covariance for a vector of random variables?

$$\mathrm{cov}(Z) = E\{(Z - E(Z))(Z - E(Z))^T\}$$

If $Z = \begin{bmatrix} X+2Y \\ 1-X-Y \\ 2X-Y \end{bmatrix}$, then

cov(Z) $= E\left\{\begin{bmatrix} (X+2Y)^2 & (X+2Y)(-X-Y) & (X+2Y)(2X-Y) \\ (X+2Y)(-X-Y) & (X+Y)^2 & (-X-Y)(2X-Y) \\ (X+2Y)(2X-Y) & (-X-Y)(2X-Y) & (2X-Y)^2 \end{bmatrix}\right\}$

$\mathrm{Cov}(\vec{X}) = \begin{bmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1,X_2) & \cdots & \mathrm{Cov}(X_1,X_p) \\ \mathrm{Cov}(X_2,X_1) & \mathrm{Var}(X_2) & \cdots & \mathrm{Cov}(X_2,X_p) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}(X_p,X_1) & \mathrm{Cov}(X_p,X_2) & \cdots & \mathrm{Var}(X_p) \end{bmatrix}.$, where X = [X1, X2, …, Xn]

cov(X) is a symmetric matrix.

39. What is a normal distribution?

Characterised by a bell curve, a normal distribution is a probability distribution of a continuous variable. It is defined by mean and a standard deviation. The distribution is symmetric about the mean and the data near the mean are more frequent in occurrence than the data far from the mean. Data far from mean are termed outliers. PDF of a normal distribution is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

40. Write the formula for Bayes rule?

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

P(A|B) = posterior, P(B|A) = likelihood, P(A) = prior,

P(B) = evidence (P(B) must be greater than 0).

For k mutually exclusive (no two events occur at the same time) and exhaustive (at least one event occurs all the times and contain entire range of possible events) events:

$$P(A_j | B) = \frac{P(A_j \cap B)}{P(B)} = \frac{P(B| A_j)P(A_j)}{\sum_{i=1}^{k} P(B| A_i) \cdot P(A_i)} \quad j = 1, \ldots, k$$

41. If two random variables are related in a deterministic way, how are the PDFs related?

Let X be a random variable and Y = f(x) be the deterministic relationship between x and y; If we have pdf(x) then PDF(y) = PDF(x)/|f'(x)|; Y becomes a new random variable whose PDF can be found out from PDF of x. If two random variables are related in a deterministic way, their PDFs are also related in a deterministic way.

Ex: if y = x^2 and PDF(X) = $f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

then PDF(Y) =

$$\begin{aligned} f_Y(y) &= \frac{f_X(x_1)}{|g'(x_1)|} + \frac{f_X(x_2)}{|g'(x_2)|} \\ &= \frac{f_X(\sqrt{y})}{|2\sqrt{y}|} + \frac{f_X(-\sqrt{y})}{|-2\sqrt{y}|} \\ &= \frac{1}{2\sqrt{2\pi y}} e^{-\frac{y}{2}} + \frac{1}{2\sqrt{2\pi y}} e^{-\frac{y}{2}} \\ &= \frac{1}{\sqrt{2\pi y}} e^{-\frac{y}{2}}, \text{ for } y \in (0, \infty) \end{aligned}$$

42. What is Bayes' Theorem? How is it useful in a machine learning context?

Bayes Theorem: The probability of event A conditional on event B is defined as:

P(A|B) = P(A & B) / P(B)

As B is an event already occurred its probability is non zero. P(B)>0;

Also P(A and B) = P(A|B) * P(B) = P(B|A) * P(A);

Bayes theorem finds one of its applications in Machine Learning as Naive Bayes Classifier. It assumes that each of the variables of the data are independent to each other (the naive part). The probability of the datapoint generating any output is determined based on the data itself. For example in a classification problem, the probability of a data point with n variables belonging to a class is dependent on the probability of values of each of the n variables that belong to the class.

P(H|D) = P(D|H) * P(H) / P(D); H - Machine Learning Hypothesis, D - Data;

The variables of the data can be assumed to be conditionally independent:

$$p(C_k \mid x_1, \dots, x_n) \propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

43. Why is “Naive” Bayes naive?

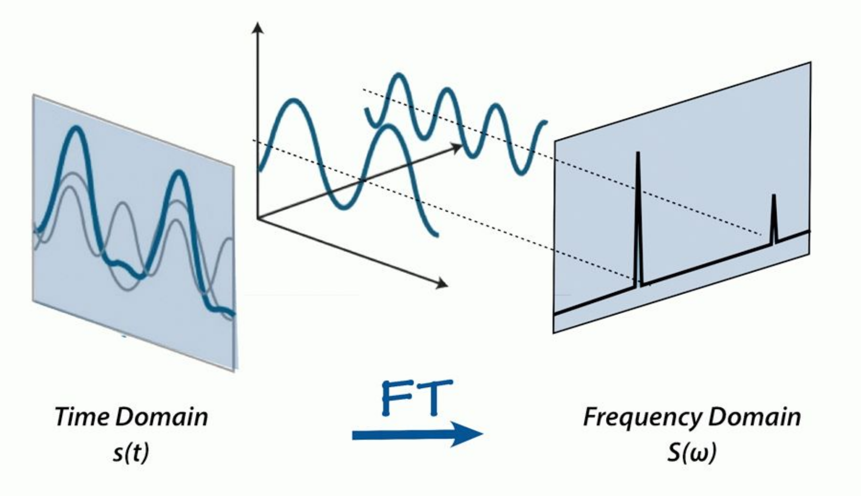
The conditional independence of the variables of a dataframe is an assumption in Naive Bayes which can never be true in practice. The conditional independence assumption is made to simplify the computations of the conditional probabilities. Naive Bayes is naive due to this assumption.

44. What is a Fourier transform?

Fourier transformation decomposes a function of time into its constituent frequencies. Fourier transform of f(x) is given as:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i x \xi}\, dx$$

A time domain random variable is converted into a frequency domain random variable



45. What is the difference between covariance and correlation?

Covariance indicates the linear relationship between variables (degree of which the two random variables are linearly associated). Correlation on the other hand indicates the strength and direction of the linear relationships between variables. Covariance is used to determine how much two random variables vary together while correlation is used to determine when a change in one variable can result in change in another. Correlation is unaffected by scale and location of the distribution which can thus be used to compare two random variables.

46. What is the difference between skewed and uniform distribution?

Uniform distribution: All the observations are equally spread across the range of the distribution, Skewed distribution: the observations are concentrated at one side of the distribution range.

47. Is it possible to capture the correlation between continuous and categorical variables? If yes, how?

Correlation is a measure of the *relationship* between two variables. It includes usage of means which is not defined for a categorical variable.

Let's say we have a categorical variable called Gender which has the levels Male and Female. There is a continuous variable called expenditure. We can observe the distribution of the expenditure variable among the two groups of Gender using t-test(testing means), KS tests, any regular univariate plots(boxplot, violin plot, pdf with hue(each plot for each group) applied).

We can use Logistic Regression to develop a model that can predict categorical variable from continuous variable. If the Logistic Regression classifier has a high degree of accuracy then there exists a correlation between the variables.

For a binary categorical variable we can use Point- Biserial Correlation Coefficient which assumes that the continuous variable is normally distributed.

48. A data scientist at Google was asked that: Tossing a coin ten times resulted in 8 heads and 2 tails. How would you analyze whether the coin is fair? What is the p-value? (We have the approach in one of the above questions) Choose a test statistic: **Probability of getting heads P(H)**, Null Hypothesis: **P(H) = 0.5**, Alternative hypothesis **P(H) != 0.5**, Observed that we got 8 heads and 2 tails: p-value = **P(Heads = 8 | Null hypothesis)** (don't get confused with P(H>=8)), Choose threshold or significance level: significant p-value = **0.05**, P(H=8 | Null H.) = $^{10}C_8$ * $(0.5)^8$ * $(1 – 0.5)^2$ = (10*9/2) / $(0.5)^{10}$ = 0.043945; p-value is less than our significance level; thus we can reject the Null hypothesis which says that the probability of getting heads is 0.5. The coin is unfair.

Dimensionality Reduction

1. What is dimensionality reduction?

   The process of reducing dimensionality from high to low ensuring least loss of information. The dimensionality of a data set is the number of random variables that define each data point. Through dimensionality reduction we are obtaining a smaller set of random variables (principal components) which can equally represent the data set that preserves the largest variance.

2. Explain Principal Component Analysis?

   PCA is a technique to get reduced dimensions and keeping maximum information intact (preserve maximum variance).

   • Original features of the dataset are converted to the Principal Components which are linear combinations of the existing features (linear combination means : combination of two or more original features, or splitting one original feature and taking one new feature from it). The feature that causes highest variance is the first Principal Component. The feature that is responsible for second highest variance is considered the second Principal Component, and so on. The data is generally normalized before applying PCA.

3. What is t-SNE?

   T-distributed stochastic embedding neighborhood is a tool to visualize high dimensional data. It transforms the dataset into lower dimensionality by projecting (non-linear projection) neighborhood data points into a more convenient space.

4. How to apply t-SNE and interpret its output?

   t-SNE plots are applied for visualizing high dimensional data, but its results will be misleading. The goal is to find a representation of the high dimensional data in a low dimensional space. Its algorithm is applied by varying the parameters of the algorithm. Its performance changes as the perplexity changes.

   tSNE does not always produce the same output. The transformation algorithm is non-linear and is stochastic in nature. Depending on the dataset a specific hyperparameter value will be able to get proper outputs from tSNE. Cluster size in the tSNE plot has no information. Distances between clusters have no information. tSNE is flexible and makes adjustments to find structure in the data.

5. What is the “Crowding problem”?

   In stochastic neighborhood embedding neighborhood points get projected onto the same location resulting in squashing of data points. And non-neighbor data points can also be seen to crowd with neighborhood data points due to reduction in dimensionality. Well represented clusters in high dimensionality get squashed in low dimensionality due to unavailability of space to represent them neatly.

6. Importance of PCA?

   PCA reduces dimensionality and allows to explore data with less compute. PCA is successfully used in computer vision tasks, data mining, finance. The visualizations can be used to interpret clusters, tends and outliers.

7. Limitations of PCA?

   PCA results in loss of information when datasets are non linear. Data points may get crowded on low dimensionality.

8. You are given a train data set having 1000 columns and 1 million rows. The data set is based on a classification problem. Your manager has asked you to reduce the dimension of this data so that model computation time can be reduced. Your machine has memory constraints. What would you do? (You are free to make practical assumptions.

   - Reduce or close RAM consumption by other applications
   - Sample the dataset randomly to fit the ram
   - Remove correlated variables (correlation for numerical features and chi-square test for categorical variables)
   - Use dimensionality reduction techniques (PCA)
   - Build a simple model such as linear model using SGD
   - Using domain knowledge to select features.

9. Is rotation necessary in PCA? If yes, Why?

   Yes, rotation (orthogonal) is necessary because it maximizes the difference between variance captured by the component. This makes the components easier to interpret. Not to forget, that's the motive of doing PCA where we aim to select fewer components (than features) which can explain the maximum variance in the data set. By doing rotation, the relative location of the components doesn't change, it only changes the actual coordinates of the points.

   If we don't rotate the components, the effect of PCA will diminish and we'll have to select more number of components to explain variance in the data set.

10. You are given a data set. The data set contains many variables, some of which are highly correlated and you know about it. Your manager has asked you to run PCA. Would you remove correlated variables first? Why?

    Correlated variables inflate the variance explained by a particular component. PCA will give more importance to correlated variables. Thus we need to remove correlated features before applying PCA. This will help us get a good representation of the dataset.

Data Science

Machine Learning

Classification and Regression models:

k-Nearest Neighbors

1. Explain about K-Nearest Neighbors?

   kNN: basic, supervised learning, classification + regression; Here the "neighbors" are objects from a training dataset with a known target class. In k-NN classification, the object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small and odd).

2. KNN is a non-parametric, lazy algorithm. It stores all available cases and classifies new cases based on a similarity measure.

   KNN is a non-parametric lazy learning algorithm: it does not make any assumptions on the underlying distribution of data. It stores all training data for evaluating similarity at test time. kNN does not explicitly have a training phase. The model makes predictions for the data after the query.

3. Failure cases of KNN?

   kNN fails when the query point is an outlier or when the data is extremely randomized which has no useful information.

4. Define Distance measures: Euclidean(L2) , Manhattan(L1), Minkowski, Hamming

   Euclidean distance: $||x1 - x2||_2 = (summ(x_{1i} - x_{2i})^2)^{1/2}$

   $||x1 - x2||_2$ àL2 Norm of vector

   $||x1||_2 = (summ(x_{1i})^2)^{1/2}$

   Manhattan distance: $||x1 - x2||_1 = (summ(abs(x_{1i} - x_{2i})))$

   à L1 norm of vector

   $||x1||_1 = (summ(abs(x_{1i})))$

   Minkowski distance: Lp norm: $||x1 - x2||_p = (summ(abs(x_{1i} - x_{2i}))^p)^{1/p}$

   P>2

   Distances are between two points and Norm is for vectors;

   Hamming distance: Number of locations where there is a difference in two vectors

Text processing (binary BOW or Boolean vector)

X_1 = [0, 1, 1, 0, 1, 0, 0]

X_2 = [1, 0, 1, 0, 1, 0, 1]

H(X1, X2) = 1+1+0+0+0+0+1 = 3

5. What is Cosine Distance & Cosine Similarity?

As distance increases similarity decreases. Cosine similarity is measured by the cosine of the angle between two vectors. Cosine distance is 1s complement of cosine similarity.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$$

It is useful for finding similarities between text and can work well with sparse data.

6. How to measure the effectiveness of k-NN?

To measure performance of any ML model: Split into train and test sets without having intersection, Train the model on train set (Using CV), Predict the outputs of the model on test set, Generate a performance metric between predictions and the target values to understand the performance of the model.

7. Limitations of KNN?

Large time complexity and space complexity at run time, cannot be applied for low latency applications.

8. How to handle Overfitting and Underfitting in KNN?

KNN is tuned for selecting the best number of nearest neighbors, k. For k =1 we will have overfitting and for k = n (dataset size) we will have overfitting. Cross validation is applied to check for best k which gives least error. Varying k from 1 to n we will reach a value that best fits the model on the dataset.

9. Need for Cross validation?

Cross Validation is required to tune the model hyper parameters. Test dataset is generally used to understand the performance of the models but cannot be used for hyper parameter tuning.

10. What is K-fold cross validation?

While doing 1-fold CV the validation dataset remains untouched during training. As a result useful information may get lost. The models are tuned K times (this k is different from the number of neighbors kNN) for every hyperparameter. The training set is divided into K parts (generally 10). At each fold K-1 different parts of the training data are used to train the model and the remaining

1 part of the training data is used for Cross Validation. As the K-folds CV completes we will have the model trained on the whole training data.

11. What is Time based splitting?

If time stamps are available in the dataset, the dataset is split into train, validation and test set based on time. For this the dataset is first sorted based on time. This allows to train the model on past data and predict on future rather than doing the reverse.

12. Explain k-NN for regression?

The target for Regression will be a continuous random variable. The prediction for a datapoint is made based on mean or median of all k nearest neighbors.

13. Weighted k-NN?

Instead of taking Majority vote, we could give more weightage to points that are closer. One way to do this is to calculate 1/(distance) measures.

14. How to build a kd-tree?

By alternating between axes, the data space is divided into axis parallel hyperplanes into hyper cubes. Project the data points on the corresponding axis of the iteration and divide the space into two halves through the median.Alternate between axes to split the entire dataset.

15. How to find nearest neighbors using kd-tree?

Geometric interpretation: After generating kd-tree (which is also represented in the form of tree and in the form of space separating hyper cubes) for the dataset draw a hypersphere with centre as query point and radius as distance to nearest neighbor in the same hypercube. If the hypercube gets intersected by any of the hyperplanes of the kd-tree we search for nearest neighbors in the adjacent hypercube that corresponds to the intersection hyperplane. We are back tracking the kd tree to check for nearest neighbors.

16. What is Locality sensitive Hashing (LSH)? (Hashing vs LSH)?

LSH computes a hash function such that nearest data points pairs are stored in the same bucket. A bucket relevant to a query point is determined and the k nearest neighbors are searched inside the bucket. This allows us to avoid searching throughout the data space. It is a randomized algorithm which provides a different hash function every time.

17. LSH for cosine similarity?

Cosine similarity is given by cosine of the angle between two vectors. Cosine similar data points are stored in a bucket of the hash table. The keys for the buckets in hash table are generated from the signed directions of the data points with respect to the random hyperplanes.

18. LSH for Euclidean distance?

The hash function for the Euclidean distance contains the values of the region in which data points fall with respect to the axes of the data space. On each axis the space is divided into m regions. The hash function will generate an m dimensional vector where each cell of the vector has a numerical value that represents the region in which the datapoint falls with respect to the axis corresponding to the cell. Data points that fall in the same region with respect to an axis, will fall in different regions on another axis.

19. In k-means or kNN, we use Euclidean distance to calculate the distance between nearest neighbors. Why not Manhattan distance?

    Both Euclidean distance and Manhattan distance are used in KNN. The selection depends on the dataset. Euclidean distance becomes invalid when it gives a measure of similarity between two data points which is not true (example - such as distance between to cities by road) For calculating distance for a road trip we need manhattan distance.

20. How to test and know whether or not we have an overfitting problem?

    If the performance of the model is very good on the train set and poor on the test set then the model is overfitting on the train set. We can test the performance by making predictions on the test set.

21. How is kNN different from k-means clustering?

    KMeans generates K clusters out of the dataset without having a target variable. Kmeans is a clustering algorithm and there is no training phase. In kNN each data point is classified into a class by considering the classes of k Nearest neighbors.

22. Can you explain the difference between a Test Set and a Validation Set?

    A test set is a dataset that is used to evaluate the performance of a trained ML model. A validation set is a dataset that is used during the training of the model to select or tune best values for the hyper parameters of the model.

23. How can you avoid overfitting in KNN?

    Overfitting in kNN occurs when k is small. Increasing k generally uptio 51 reduces overfitting in KNN. We can also use dimensionality reduction or feature selection techniques to avoid overfitting which can happen due to the curse of dimensionality.

24. Other KNN attributes: KNN does more computation on test time rather than on train time. Training involves storing all the data points in the RAM. Manhattan, Minkowski, Euclidean, Jaccard and all other distance measures can be used in KNN. It can be used for both classification and regression. Hamming distance is used for categorical variables. It classifies data points based on Majority vote by the k nearest neighbors. When k increases variance decreases and bias increases, the model moves from being an overfitting model to a well fit model or to an underfitting model. Increasing K can reduce noise (reducing variance). The decision boundary of the model becomes smoother as k is increased. The training time for any value of k for KNN is the same.

Classification Algorithms in various situations

1. What is Imbalanced and balanced dataset?

   Imbalanced and balanced datasets are applicable to a categorical target variable; The dataset is balanced when the number of data points that belong to each class is similar; if the number of data points in a class are different from other classes then the dataset is imbalanced;

2. Define Multi-class classification?

   In a multi class classification problem we will have a dataset with class labels more than 2.

3. Explain Impact of Outliers?

   Outliers impact the decision surfaces of the models. It will inflate the errors and trick the models into optimizing for the outliers. The models trained on a dataset containing outliers generally have poor generalization capability. This can be tackled by outlier removal or model regularization and using CV to avoid model overfitting on training dataset.

4. What is the Local Outlier Factor?

   There will be different clusters in the dataset. And these datasets might have different local densities. A sparse cluster may make the model interpret that an outlier to the denser cluster is not an outlier as the outlier distance to the dense cluster is approximately the same as the average distance of the sparse cluster. For every data point compute average of k NN points and sort to remove global outliers. Compute local densities and compare each point with the local densities of its k Nearest Neighbors, sort and remove outliers.

5. What is k-distance (A), N(A)?

   K_distance (A) = Distance of kth nearest neighbor of A to A. N(A) = Neighborhood of A which is a set of nearest neighbors.

6. Define reachability-distance(A, B)?

   It is given by max(k_distance(A), distance(A,B)), if B is in the neighborhood of A then reachability_distance(A, B) = k_distance(A). If B is not in the neighborhood of A then reachability_distance(A, B) = distance(A, B).

7. What is Local-reachability-density(A)?

   It is the inverse of average reachability distance of A from its neighbors.

8. Define LOF(A)?

   It is the ratio of Average local reachability distance of points in the neighborhood of A and Local Reachability distance of A. If the local outlier factor of point A is large then the density of the nearest neighbors is small thus A is an outlier.

9. What is the Impact of having different scales on the model & how does Column standardization solves the problem?

   The features in the dataset that have higher values will dominate in the training of the model. The model only learns from these higher values. The distance measures will not be proper between two data points. Even if the smaller feature had a large variation, the variation in the larger scaled feature does not allow the model to observe the change in smaller feature which will result in making an important feature insignificant. And this results in poor performance.

10. What is Interpretability?

    An ML model needs to be highly interpretable. The model is said to be interpretable when in addition to generating predictions the decision making process should be explainable. Example: KNN makes predictions based on its nearest neighbors. The nearest neighbors can be explored to determine why the model has made a certain prediction.

11. Handling categorical and numerical features?

    Machine Learning models understand numbers. The categorical features should be converted to numerical features that best represents the feature. Numerical features can be normalized to give it as an input to an ML model. For categorical features we can either use One hot encoding, Response encoding or use domain knowledge for converting categorical data into numerical data.

12. Handling missing values by imputation?

    Missing values are interpreted as NaN by ML models. This impacts the model performance. It can be mean, median or mode imputed (Statistical imputation), or train a ML model on non missing data to predict missing data taking it as test data (Model based imputation).

13. Bias-Variance tradeoff?

    Every model optimizes on a certain cost function. The error made in predictions contains bias and variance. This error is termed a generalization error of the ML model. Bias arises due to simplifying assumptions and Variance arises due to sensitivity to the small fluctuations in training data. Generally high bias models have low variance and high variance models have low bias. It is desirable to have low bias and low variance. Thus a tradeoff is made between bias and variance to optimize the performance of the model.

Performance Measurement of Models

1. What is Accuracy?

   It is defined as the ratio of correctly classified data points to total number of data points.

2. Explain about Confusion matrix, TPR, FPR, FNR, TNR?

   A confusion matrix has row values as the number of predicted class labels and column values as the number of actual class labels (row and columns can be vice versa). Each of the cells compare the actual and predicted class labels. The diagonal elements are generally preferred to be high. Non-zero off diagonal elements indicate that the model is predicting something that is not true.

   True Negatives: Predicted Negatives that are true or correct

   False Negatives: Predicted Negatives that are false, these points are actually positives

   True Positives: Predicted Positives correctly;

   False Positives: Predicted Positives are wrong, these are actually negatives;

   TPR (True Positive Rate) = TP / P = TP/ (TP+FN)

   TNR = TN/N = TN/ (TN + FP)

   FPR = FP / N

   FNR = FN / P

   Model is good if TPR, TNR are high and FPR, FNR are low;

   A dumb model makes one of TPR or TNR = 0;

3. What do you understand about Precision & recall, F1-score? How would you use it?

   Precision = TP/ (TP + FP); of all the predicted positives how many are actually positive

   Recall = TP / (TP + FN); of all actual positives how many are predicted positive

   F1 score = 2 * Pr * Re / (Pr + Re); high precision will result in high precision and high recall; f1 score is harmonic mean of precision and recall;

   F1 score = 2/(1/PR + 1/Re) = 2TP/(2TP + FP + FN);

   Based on the business problem the models are trained to optimize its performance on the metric. Precision, Recall and F1_score are some of the performance metrics that are available to measure the performance of classification models.

4. What is the ROC Curve and what is AUC (a.k.a. AUROC)?

   An ROC curve is a graphical plot that illustrates the generalization ability of a binary classifier model. It is a curve between TPR and FPR where the data is generated by varying the classification thresholds. AUROC ranges from 0 to 1. This is the area under the ROC Curve that allows us to understand the performance of the model. Preferred value for AUC is a value >

0.5. An AUROC of 0.5 implies the model is a dumb model. While AUROC less than 0.5 implies that the model is trained to make reversed predictions (predicting 0 for 1 and vice versa).

5. What is Log-loss and how it helps to improve performance?

   Log loss uses probability scores from the output of the model. It is used as a loss function for the model to optimize upon. A perfect model would have a log loss of 0. Log loss is defined as the average negative logarithm of probability of correct class label.

   Log - loss = (1/n) summ (1 to n) (yi * log(pi) + (1 - yi) * log(1 - pi)), it penalizes even for small deviations from the actual class label.

6. Explain about R-Squared/ Coefficient of determination.

   It is mathematically defined as 1 minus ratio of sum of squares of residues (difference in actual value and predicted value) and sum of squares of errors (difference in predicted value and expected value of the actual values). It is the proportion of the variance in the dependent variable that is predictable from the variance of independent features. The best value for the coefficient of determination is 1 (which says that there is no residue due to the model). Coefficient of determination less than 1 implies that the model is worse than a simple mean model (a model that predicts output as mean for all query points)

7. Explain about Median absolute deviation (MAD)? Importance of MAD?

   Coefficient of determination being dependent on the square of residues is highly impacted by outliers. Median absolute deviation is calculated as the median of absolute values of the differences between the values of each data point from the median of the data points (Median(|xi - Median(x)|). MAD is robust to outliers. This gives a better statistic for determining the dispersity or the spread of the distribution.

8. Define Distribution of errors?

   The distribution of errors contains all the errors that have been generated by the model. We can plot PDF and CDF of errors to compare two models. While errors generally follow a log normal distribution, the model that has Cumulative Distribution Function curve above is a better model.

9. Which is more important to you– model accuracy, or model performance?

   Model performance is important to me. Model accuracy is one of the types of performance measurement metric. If the dataset is well balanced then model accuracy is sufficient to train the model. Generally good model performance implies the model has better generalization ability on the test set. Model performance is a general term which contains model accuracy.

10. Can you cite some examples where a false positive is more important than a false negative?

    False positive implies that a data point is predicted positive which is actually a negative class data point. We will false positives important when we need to reduce incorrectly predicting negative class data points and it is alright to incorrectly predict negative for an actual positive data point. This implies that the model should avoid accepting negative data points. Example: Recruitment process generally comes into this category. It is alright to reject a skilled candidate but it is costly to accept an unskilled candidate. In a financial transaction, it is alright to send an alert for a non fraudulent transaction rather than not sending an alert for a fraudulent transaction.

11. Can you cite some examples where a false negative is more important than a false positive?

    In medical treatments, a person without a disease can be predicted positive and be sent to further tests to ensure his condition. While it is important to not predict negative (no disease) for a person having a disease.

12. Can you cite some examples where both false positive and false negatives are equally important?

    For an airplane manufacturing company, rejecting a perfectly manufactured airplane will result in loss of money and accepting a damaged airplane will result in loss of lives.

13. What is the most frequent metric to assess model accuracy for classification problems?

    AUROC

14. Why is Area Under ROC Curve (AUROC) better than raw accuracy as an out-of- sample evaluation metric?

    Raw accuracy is biased towards the majority class in the dataset. A dumb model can have 99% accuracy predicting all data points to belong to class 0 when the dataset has 99% of the data points belonging to class 0. It does not care about class 1. While AUROC takes care of the misclassifications in both the classes. A dumb model which had 99% accuracy will have 0.5 AUROC which truly tells us the performance of the model.

Naive Bayes

1. What is Conditional probability?

   Probability of an event occurring given that another event has occurred.

2. Define Independent vs Mutually exclusive events?

   Two events are independent when occurrence of one event has no impact on the occurrence of the other event. Two events are mutually exclusive events when they cannot occur one after the other.

   P(A|B) = P(A) and P(B|A) = P(B) → A and B are independent events

   P(A|B) = P(B|A) = 0 → A and B are mutually exclusive events.

3. Explain Bayes Theorem?

   Bayes theorem describes the probability of an event based on prior knowledge of conditions that might be related to an event. Bayes theorem is mathematically stated as:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

   Posterior probability = Likelihood * prior / evidence, P(B) !=0;

4. A test has a true positive rate of 100% and false positive rate of 5%. There is a population with a 1/1000 rate of having the condition the test identifies. Considering a positive test, what is the probability of having that condition?

   Given: TPR = 1 = P(Positive|Condition), FPR = 0.05 = P(Positive|No condition),

   P(Condition) = 0.001

   P(Condition|Positive) = ?

   We have Bayes Theorem as:

   P(A|B) = P(A) * P(B|A) / P(B)

   P(Condition | Positive) = P(Condition) * P(Positive|Condition) / P(Positive)

   And,

   P(B) = P(A & B) + P(-A & B) = P(B|A) * P(A) + P(-A)*P(B|-A)

   P(Positive) = P(Condition) * P(Positive|Condition) + P(No Condition) * P(Positive | No condition)

   P(C | P) = (0.001 * 1) / ((0.001 * 1) + (1-0.001) * (0.05)) = 0.001 / (0.001 + (0.999*0.05))

   P(C | P) ~ 0.0196

5. How to apply Naive Bayes on Text data?

   Naive Bayes is applied on the vectorized text data by computing the probabilities of each word belonging to a class. The assumption of Naive Bayes that the features of the data are conditionally independent. Probability of a classifying a text into a certain class is proportional to the multiplication of probability of the class with individual probabilities of the words belonging to the class.

6. What is Laplace/Additive Smoothing?

   At the end of training in Naive Bayes we will have all the likelihoods and priors computed. Laplace smoothing is applied to incorporate new words that did not occur in the training data. As the word did not occur in the training set, the probability of the word belonging to any class will be 0. This will make the probability of the whole text to belong to a class becomes 0 (multiplication with 0).

   Probability of individual words per class is computed as:

   $P(W'|y=1) = (0 + \alpha)/(n1 + \alpha k)$

   N1 = # of data points for y = 1

   Smoothing coefficient, α != 0; generally = 1

   k = number of unique values W' can take (for a categorical feature it can k = # of unique categories)

   When α = large; the likelihood will be around 0.5;

7. Explain Log-probabilities for numerical stability?

   Multiplication of individual word probability values for text data results in a value that reaches 0. This is avoided by applying a log on the probabilities to have addition of log of individual word probabilities.

8. In Naive Bayes how to handle Bias and Variance tradeoff?

   The Laplace Smoothing parameter Alpha is varied to have an optimum bias and variance for the model. When alpha is 0 we will have high variance and when alpha is very large we will have likelihoods of the words equal to 0.5 resulting in an underfitting model.

9. How to handle Numerical features (Gaussian NB)?

   Numerical features are assumed to be Gaussian. Probabilities are determined by considering the distribution of the data points belonging to different classes separately. (Probabilities are calculated separately for each class).

Logistic Regression and Linear Regression

Logistic Regression and Linear Regression

1. Explain about Logistic regression? It is a statistical model that uses a logistic function to model a binary dependent variable. It can output a value that corresponds to the probability of the data point to belong to a class. It computes a weighted sum of the input features and through application of the logistic function it provides the probability of the instance to belong to a class. If the logistic value is greater than 0.5 (or the value of weighted inputs is greater than 0) the instance is classified as a positive class data point.

2. What is sigmoid function & Squashing? A mathematical function that has an S shaped curve bound between 0 and 1. $S(x) = \frac{1}{1 + e^{-x}}$. The sigmoid function reduces the impact of the outliers by contracting or squashing the distribution to an interval [0, 1].

3. After analysing the model, your manager has informed you that your regression model is suffering from multicollinearity. How would you check if he’s true? Without losing any information, can you still build a better model? Multicollinearity can be checked using correlation analysis of the features and remove features with high correlations. And we can build a better model by using Regularization (L1 or L2 norm)..

4. What are the basic assumptions to be made for linear regression? The dependent variable or the target is binary. Less correlation between independent variables. The dataset is linearly separable.

5. What is the difference between stochastic gradient descent (SGD) and gradient descent (GD)? In SGD only one data point is used per iteration to calculate the value of the loss function. While for GD all the data points are used to calculate the value of the loss function.

6. When would you use GD over SGD, and vice-versa? GD optimizes better and is preferable for small size datasets. While SGD allows faster convergence and is preferable for large size datasets.

7. How do you decide whether your linear regression model fits the data? We can use performance metrics such as Root Mean Square Error, Mean Absolute Error, R^2, or adjusted R^2 to evaluate the predictions.

8. Is it possible to perform logistic regression with Microsoft Excel? Microsoft Excel has an inbuilt Logistic Regression feature.

9. When will you use classification over regression? Classification is used for predicting the belongingness of a datapoint to a finite number of classes. While regression outputs a continuous value.

10. Why isn't Logistic Regression called Logistic Classification? Logistic Regression behaves like a classification algorithm when its continuous output is combined with a decision rule (such

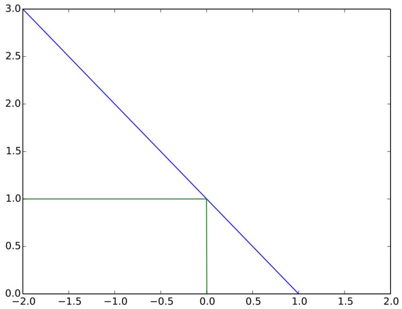
as the logistic value > 0.5) to predict belongingness of the data point to a class. In addition to a regression model it estimates the probability of class membership of data points.

External Resources: (To be processed)

- https://www.analyticsvidhya.com/blog/2017/08/skilltest-logistic-regression/
- https://www.listendata.com/2017/03/predictive-modeling-interview-questions.html
- https://www.analyticsvidhya.com/blog/2017/07/30-questions-to-test-a-data-scientist-on-linear-regression/
- https://www.analyticsvidhya.com/blog/2016/12/45-questions-to-test-a-data-scientist-on-regression-skill-test-regression-solution/
- https://www.listendata.com/2018/03/regression-analysis.html

Support Vector Machine

1. Explain About SVM? SVM stands for **Support Vector Machine** which is a classification and Regression Technique both (SVC- Support vector classification and SVR-Support vector Regression). It assumes data to be almost linearly separable and finds a hyperplane which widely separates points i.e **Margin Maximizing**. The points through which the margin maximizing planes pass through are called support vectors.
2. What is Hinge Loss? Just like in Logistic Regression we have Log loss similarly in SVM we have a Hinge Loss (max(0, 1-t). Hinge loss is defined as a loss which is used for **maximum margin** by classifying points .It also approximates 0-1 loss.

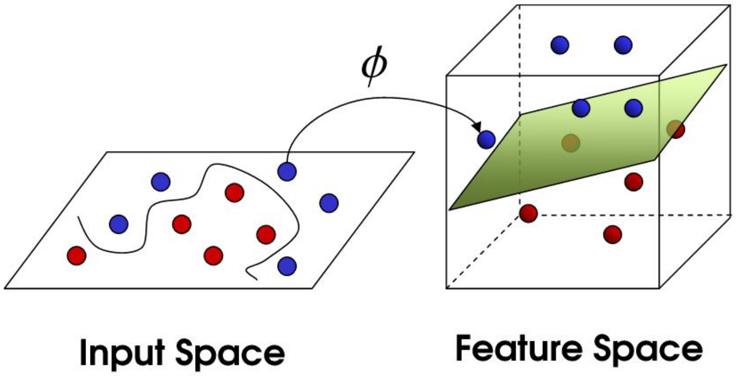


3. Dual form of SVM formulation? If we have data which is non-linear then we can separate that with a powerful technique called kernelization instead of feature transform doing explicity.So for that simplified equation is called Dual form of SVM and is represented as follows:

$$\underset{\alpha}{maximize} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(X_i^T . X_j \right)$$

Such that **0 ≤ αi ≤ C for ∀i, and ∑αiyi = 0**

4. What is a Kernel trick? SVM became the most popular algorithm because of its kernel trick. Kernel trick is simply a dot product of two vectors x and y in high dimension feature space therefore they are also known as **“generalized dot product**”. The kernel trick avoids explicit mapping that is needed to get linear learning algorithms to learn a non linear decision boundary. It can be thought of applying feature transformations internally.



5. What is a Polynomial kernel? Polynomial Kernels are the simplest kernel which separates non linear data in higher dimension. It is defined as follows

   $$K(x, y) = (x^\mathsf{T} y + c)^d$$

   Where d is dimension let suppose we take d = 2 (Quadratic Kernel) which will be able to separate all the conic sections(circle,ellipse,parabola,hyperbola).

6. What is RBF-Kernel? Radial basis function kernel is used for defining decision boundaries between the classes when boundaries are hypothesized to be curve shaped. RBF-SVM is a general purpose kernel similar to working of kNN.

7. Find Train and run time complexities for SVM? Train time complexity O(n^2)Run time complexity O(kd): k-support vectors, d-dimensionality.

8. Explain about SVM Regression? SVR gives us the flexibility to define how much error is acceptable in our model and will find one appropriate line (or hyperplane in higher dimension) to fit the data.

9. When to use Logistic Regression over SVM? Logistic Regression can be used when the number of features and the data set size are smaller.

10. What are the advantages and disadvantages of Logistic Regression and SVM? SVM is effective in high dimensional data, as it considers support vectors it is memory efficient, risk of overfitting is less in SVM (impacted less due to outliers), SVM can be applied for Regression problems. Not suitable for large size datasets (N), does not perform well when there is overlapping of the classes. Logistic Regression provides a probabilistic interpretation of the class of the data points, It performs well when the data is linearly separable, it overfits when the training examples are less than the number of features, it cannot be applied on non-linear data.

11. Give some situations where you will use an SVM over a RandomForest Machine Learning algorithm and vice-versa.

Random Forest	**SVM**
It is used in Multiclass problem	It is mainly used for binary classification , but if need to used for multiclass it should be converted to binary class by OVR
Here , data is used as it is	Here, standardisation/Normalization is required because it “maximize margin” thus relies on distance
Gives probability belonging to class	Gives distance to boundary and you need to convert it into probability.
It can work when N is large	It is not preferable when N (dataset size) is large

12. What is a convex hull? Convex means that the polygon has no corner that is bent inwards. The convex hull of a set of points is defined as the smallest convex polygon that encloses all of the points in the set.
13. What is a large margin classifier? SVM is a type of classifier which classifies positive and negative points by finding an optimal hyperplane π. It tries to find the largest margin among three parallel planes i.e π+,π,π- to avoid overfitting therefore called a large margin classifier.
14. SVM being a large margin classifier, is it influenced by outliers? (Yes, if C is large, otherwise not)
15. What is the role of C in SVM? C in SVM allows us to avoid misclassifications in the data set. Larger C implies small margin thus avoiding misclassifying data points.
16. In SVM, what is the angle between the decision boundary and theta? They are orthogonal.
17. What is the mathematical intuition of a large margin classifier? SVM has a decision boundary with optimization formulation as:min θ: $\frac{1}{2} \Sigma(\theta_j^2) = \frac{1}{2} ||\theta||^2$

    s.t. $p^{(i)} . ||\theta|| \{>= 1 \text{ if } y^{(i)} = 1, <=1 \text{ if } y^{(i)} = -1\}$

    where $p^{(i)}$ is the projection of point $x^{(i)}$ onto the vector θ.

    θ is the normal vector to the decision boundary. When θ is large we will have a small margin as projection of x on θ will be small. Further the objective function makes θ small which will result in a large margin as projection of x on θ will be large. As projection increases θ can become smaller thus the objective leads to having a large margin classifier.

18. What is a similarity function in SVM? Why is it named so? Similarity functions in SVM are the kernel functions. The kernel functions quantify the similarity over pairs of data points.
19. How are the landmarks initially chosen in an SVM? How many and where? Landmarks are chosen near the points where the classes are linearly separable. Generally there will be k+1

support vectors useful to generate an SVM decision boundary, where k is the dimensionality of the dataset.

20. Can we apply the kernel trick to logistic regression? Why is it not used in practice then? Kernel trick is time consuming as the number of training points increases. For SVMs it is feasible as the computations required for the weight vector are limited to support vectors which are generally finite. While for logistic regression though it takes for computations the result due to kernelized logistic regression is similar to non kernelized logistic regression.

21. What is the difference between logistic regression and SVM without a kernel? Only in implementation – SVM is much more efficient and has good optimization packages.

22. How does the SVM parameter C affect the bias/variance trade off? (C = 1/lambda; lambda increases means variance decreases) As C increases margin becomes harder and bias reduces.

23. How does the SVM kernel parameter sigma^2 affect the bias/variance trade off? σ^2 is a gaussian kernel parameter, as it increases the feature transformation becomes smoother and variance decreases.

24. Can any similarity function be used for SVM? (No, have to satisfy Mercer's theorem)

25. Logistic regression vs. SVMs: When to use which one? (Let's say n and m are the number of features and training samples respectively. If n is large relative to m use log. Reg. or SVM with linear kernel, If n is small and m is intermediate, SVM with Gaussian kernel, If n is small and m is massive, Create or add more features then use log. Reg. or SVM without a kernel)

26. External Resources: 1. https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/

Decision Trees

1. How to Build a Decision Tree? Decision Trees are built on if else comparisons. Given a dataset the data points are compared based on a decision on an attribute at each level of the tree. It can be thought of dividing the data space with axis parallel hyperplanes.It is top down from a root node.
2. What is Entropy? Entropy is a measure of disorder or uncertainty in a bunch of examples. It is mathematically defined as

$$E(S)=\sum_{i=1}^{c}-p_i\log_2 p_i$$

. If all classes are equally probable then entropy will be maximum.
3. What is information Gain? Information gain is dependent on the decrease of entropy after splitting the data set on an attribute. It is given as **E(s) - average of E(child nodes of s)**, in other words entropy of the distribution before splitting minus the entropy after splitting. It is the amount of information gained after knowing the value of an attribute.
4. What is Gini Impurity? Gini impurity and Entropy are used in decision tree algorithms to decide the optimal split from a root node and subsequent splits. Gini impurity tells us about the probability of misclassifying an observation.

$$\mathrm{I}_G(p)=\sum_{i=1}^{J}p_i\sum_{k\neq i}p_k=\sum_{i=1}^{J}p_i(1-p_i)=\sum_{i=1}^{J}(p_i-{p_i}^2)=\sum_{i=1}^{J}p_i-\sum_{i=1}^{J}{p_i}^2=1-\sum_{i=1}^{J}{p_i}^2$$

5. How is splitting done for numerical features? For a numerical feature there are n-1 split points. As it is very large we can use percentiles of the distribution of the feature to generate splits and there by compute Information Gain.
6. How to handle Overfitting and Underfitting in DT? In Decision Trees, we can change the depth of the tree to handle Overfitting or Underfitting in the model. As Depth increases Overfitting on the training set arises.
7. What are Train and Run time complexity for DT? Train time complexity = O(nd logn) n - dataset size, d-dimensionality, Run time space complexity O(nodes) and time complexity O(depth) ~O(1)
8. How to implement Regression using Decision Trees? We will use Mean Squared Error or Mean (or Median) Absolute Deviation for generating splits. At each node mean or median is used as the value for the query data points that reach that node.
9. You are working on a time series data set. Your manager has asked you to build a high accuracy model. You start with the decision tree algorithm, since you know it works fairly well on all kinds of data. Later, you tried a time series regression model and got higher accuracy than the decision

tree model. Can this happen? Why? Time series data is known to possess linearity. A Decision Tree works best to detect nonlinearity which can fail to provide robust predictions on linear relationships. Thus we can get higher accuracy with linear models on time series data as compared to decision trees.

10. Running a binary classification tree algorithm is the easy part. Do you know how a tree splitting takes place i.e. how does the tree decide which variable to split at the root node and succeeding nodes? In decision trees an attribute is used for splitting at nodes depending on the information gain it can provide. We can use Entropy or Gini Impurity to determine the information gain potential that an attribute has. It is desirable to select the attribute at a node that gives low Entropy or Gini impurity.

Summarising Decision Trees: A Decision Tree can be used to build models for Classification as well as Regression problems. It is built in top-down fashion. The goal of feature selection while building a decision tree is to find features or attributes which lead to split in children nodes which have minimum entropy and Maximum information gain. A data segment can be said to be pure if all of the data instances belong to the single class. Entropy value of 0 represents that data sample is pure and Entropy value of 1 represents data sample has 50-50 split belonging to two categories.Minimal entropy leads to overfitting. Pre pruning results in Underfitting. Post pruning is preferrable.

Ensemble Models

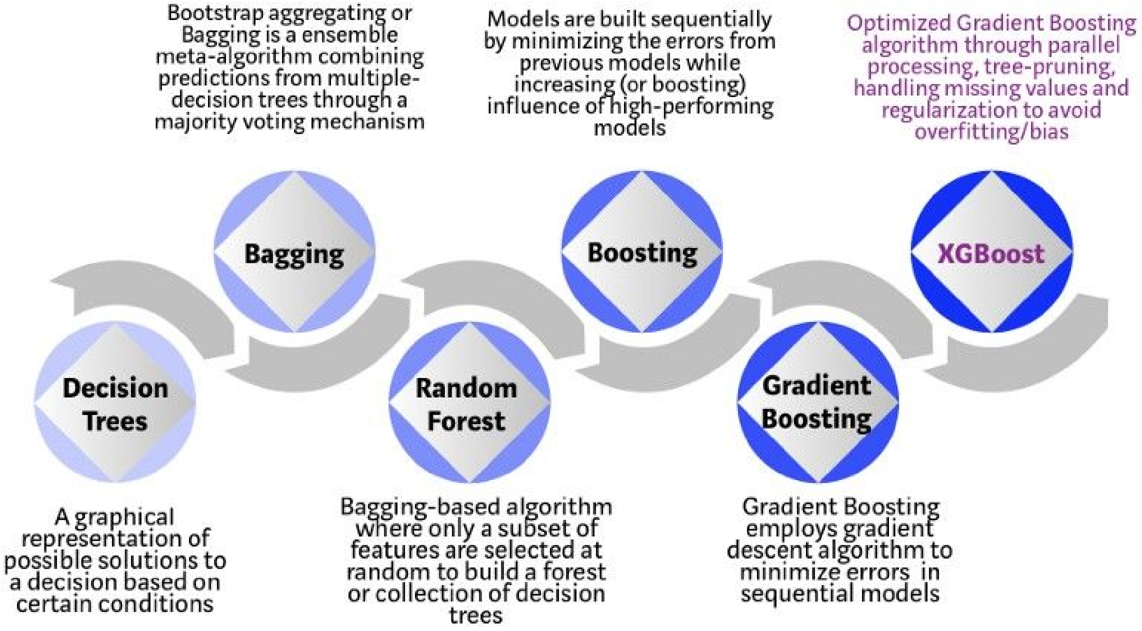
1. What are ensembles? Ensemble methods are meta-algorithms that combine several machine learning techniques (base models or learners are generally different) into one predictive model. This results in reducing bias, variance or improving predictions. The multiple base models may individually perform poor but when combined they can become a powerful model. Base learners can be thought of as different experts who can identify different aspects of the data. When combined they result in a better model.

2. What is Bootstrapped Aggregation (Bagging)? Bootstrapping involves training models (base learners) on different samples that are drawn with replacement. Aggregation involves utilising the concept of a majority for these models. Base learners are generally low bias high variance models. Through bagging we achieve a low bias and reduced variance model.

3. Explain about Random Forest? Random forests are an ensemble learning model based on bagging. This has decision trees with good depth (low bias high variance) as base learners. In addition to bagging Random Forests employ random sampling of features during training (to find the best attribute for splitting).

   Random Forest: Decision Tree (reasonable depth) base learner + row sampling with replacement + column sampling + aggregation (Majority vote or Mean/Median). Bagging works to reduce variance.

4. Explain about Boosting? The objective of Boosting is to reduce bias by ensembling low variance and high bias base models. Several weak models are trained sequentially, each trying to correct its previous model. The models are trained on the residuals or the errors generated by the previous model. At the initial stage a model is trained to fit on the whole dataset.


Bootstrap aggregating or Bagging is a ensemble meta-algorithm combining predictions from multiple-decision trees through a majority voting mechanism
Models are built sequentially by minimizing the errors from previous models while increasing (or boosting) influence of high-performing models
Optimized Gradient Boosting algorithm through parallel processing, tree-pruning, handling missing values and regularization to avoid overfitting/bias
Bagging
Boosting
XGBoost
Decision Trees
Random Forest
Gradient Boosting
A graphical representation of possible solutions to a decision based on certain conditions
Bagging-based algorithm where only a subset of features are selected at random to build a forest or collection of decision trees
Gradient Boosting employs gradient descent algorithm to minimize errors in sequential models

5. What are Residuals, Loss functions and gradients? Error generated by predecessor predictor is a residue. The loss functions are defined on the target variable and the prediction of the predecessor model. Thus they also are functions of residuals. For regression we can have squared loss whose gradient can be directly determined from residuals at the previous stage.

6. Explain about Gradient Boosting? A boosting algorithm that employs a gradient descent algorithm to minimize errors in sequential models. It uses gradient descent to fit new predictors to the residuals made by the previous predictor. For example: A Gradient boosted algorithm with 3 stages having Decision Trees (shallow depth) as base learners will be having the three models in an ensemble such that the first decision tree is trained to fit approximately on the dependent variable. The second stage base learner is fitted on the residual errors of the first stage model. And the third stage base learner is fitted on the residual errors of the second stage model.

7. What is Regularization by Shrinkage? The number of base learners in an ensemble, when high, can lead to overfitting on the training dataset. Shrinkage is a form regularization that allows us to reduce variance produced due to ensembles. Shrinkage or learning rate has a value between 0 and 1 and is multiplied to reduce the impact of each additionally fitted base-learner.

8. Explain about XGBoost? When boosting algorithms meet random sampling we will Extreme Gradient Boosting. Sampling is done in terms of features to further reduce correlations between base learners.

9. Explain about AdaBoost? Adaboost corrects its predecessors by concentrating on the training instances that were misclassified. A base learner is initially trained on the dataset. Using the predictions of this base learner the second base learner increases the weights of the misclassified training instances and gets trained on the updated weights dataset. The algorithm adapts boosting methods to the misclassifications of the predecessor model.

10. How do you implement Stacking models? To stack or have an ensemble of models the models need to be as different as possible. A stacking model is trained on the predictions of the base model taking them as meta features for a new model. This generally has poor latency performance.

11. Explain about cascading classifiers. The output from the previous model is used as additional information for the new model. Unlike multi expert systems such as voting or stacking ensembles, a cascading classifier is a multi stage ensemble.

Clustering

1. What is K-means? How can you select K for K-means? K means is an unsupervised algorithm used for grouping the dataset into K clusters. Unlike KNN which concentrates on k nearest data points, k-means is based on dividing the dataset into k clusters.Each observation is assigned to a cluster with nearest mean. K for K means is determined from its performance plot. We use the elbow method to find the best k.

2. How is KNN different from k-means clustering? KNN is a supervised algorithm that makes predictions based on k nearest neighbors from the dataset. KMeans is an unsupervised algorithm that divides the dataset into k clusters.

3. Explain about Hierarchical clustering? Hierarchical clustering is an algorithm that groups similar data points into clusters. These clusters are distinct from each other. It builds a hierarchy of clusters. There are two types: Agglomerative - grouping into clusters from each individual data point. Divisive - splitting clusters into clusters and then into data points.

4. Limitations of Hierarchical clustering? No mathematical objective function. High time and space complexity, once a method is applied it cannot be undone (once a cluster is formed it is difficult to get back the previous state of clusters).

5. Time complexity of Hierarchical clustering? Time complexity: $O(n^3)$ and Space complexity: $O(n^2)$

6. Explain about DBSCAN? Density based spatial clustering of applications with noise: This clustering algorithm groups data points that are in dense regions and separates them from sparse or low density regions. It converts data points into core points, border points and noise points. It starts by labelling each point as a core point and using a number of minimum points with a radius it determines noise points and border points.

7. Advantages and Limitations of DBSCAN? It is resistant to noise, number of clusters need not be specified, and can handle different sizes of clusters. Limitations: Varying densities and high dimensionality data, sensitive to hyper parameters: depends on distance measure which causes curse of dimensionality.

Recommender Systems and Matrix Factorization

1. Explain about Content based and Collaborative Filtering? Content based filtering makes recommendations based on user preferences for product features. It works on item-item similarity where it recommends items based on a comparison between the **content** of the items and a user profile. Collaborative filtering mimics user-user recommendations. Users who agreed in the past tend to agree in the future.

2. What is PCA, SVD? PCA and SVD are two eigenvalue methods used to reduce dimensionality without losing important information. PCA requires the input matrix to be a square matrix. SVD can work on any rectangular matrix.

3. What is NMF? A matrix is decomposed into two smaller matrices which have all elements non negative numbers and when the smaller matrices are multiplied (matrix multiplication) they result in the original matrix.

4. How to do MF for Collaborative filtering? Let A be the similarity matrix which contains user ratings for each item. And let decomposition (A = B'*C) be possible. We can find the decomposition matrices by using SGD. The problem is converted into an optimization function where the loss function contains difference between input matrix A and B'*C. This allows us to initially guess a solution and move towards an optimal solution after the optimization process.

5. How to do MF for feature engineering? After getting the decomposed form of the input matrix containing user ratings, the decomposed vectors can be used for representing users similarity matrix (B) and items similarity matrix (C) respectively.

6. Explain the relation between Clustering and MF? The formulation for clustering can be converted into a Matrix factorization form. We can decompose the input matrix into its components and KMeans can be thought of as a combination of Matrix factorization, Column constraints and a binary variable.

7. What is Hyperparameter tuning? The inter dimensionality of the decomposed matrix is the hyper parameter (d' in (n x m) = (nxd')x(d'xm)). We can use the elbow method to select the best dimension that minimizes the optimization function min (A - B'C).

8. Explain about the Cold Start problem? The dataset changes over time new items or users enter into the data set without having any past information such as ratings. Based on user-user or item-item similarity we cannot recommend items to new users or new items to users. This arises because all the elements corresponding to a new user or a new item will be 0. This problem is dealt with considering meta-features such as user location, user browser, device, item company, etc.

9. How to solve Word Vectors using MF? A co occurrence matrix can be built for the words. This co-occurrence matrix can be decomposed using SVD. From this SVD decomposed matrix we can have the vector representation of each word.

10. Explain about Eigenfaces? For image data PCA can be used to get feature vectors. Each n dimensional image is flattened into a row vector where a number of images as row vectors are stacked to form a 2d matrix. Covariance matrix is computed, then dimensionality reduction is applied. The input matrix is then multiplied with the left singular matrix to obtain Eigen faces. The dimensionality of eigenfaces can be hyper parameter tuned using % of explained variance from top k Eigenvalues.

11. How would you implement a recommendation system for our company's users? We will use Matrix factorization and graph algorithms to determine similarities between users and recommend items based on the similarity. The user-item matrix is converted into a user latent matrix and an item latent matrix. Graph algorithms allow us to take care of sparsity in the user-item matrix. It will allow us to divide the problem into candidate generation and personalization. Similar users are grouped together (rather than searching the entire graph for all similarities between all users) and item recommendations are made inside the groups.

12. How would you approach the "Netflix Prize" competition? The best solution in the competition has followed a method to divide ratings based on average of all users on the movie, average of the user on all movies, effect of the movie, and the actual predictable effect of the user on the movie. For example a movie can be rated by a user as 4. Now the rating is cleaned with the effects from other users and movies. This can be termed as normalization of global effects. We can apply a neighborhood model on item-item similarity matrix or on user-user similarity matrix.

13. 'People who bought this, also bought…' recommendations seen on Amazon is a result of which algorithm? Apparel are recommended using weighted similarity between brand, color and visual features. Similarities are determined using distance between vectors from each of the brand, color and image.

Basics of Natural Language Processing (NLP)

1. Explain about Bag of Words? A simplifying representation of texts which disregards grammar and sequence information. It counts the number of occurrences of each word in the text and vectorises the text by using each word in the vocabulary as a feature.

2. Explain about Text Preprocessing: Stemming, Stop-word removal, Tokenization, Lemmatization. Text cannot be given as input to Machine Learning models in strings format. ML models consume numerical input data. Stemming is used to reduce words to their root forms by dropping unnecessary suffix characters. E.g. Tastes, Tasty, Taste have stem word tasti. These three words indicate or relate to the same meaning. Thus it would be beneficial to consider them as single words. Tokenization: breaking a text into a set of meaningful pieces or words of the text which are called tokens. Lemmatization takes lemma of the words based on its intended meaning in the text depending on the context words.

3. Explain about uni-gram, bi-gram, n-grams? Uni-grams consider a single word of the text at a time for counting, while n-grams considers n successive words at a time.

4. What is tf-idf (term frequency- inverse document frequency)? This considers the frequency of the word in the text as well as in the whole dataset or corpus. TF IDF is a multiplication of two frequencies where term frequency gives the probability of finding a word in text while inverse document frequency considers the occurrence of the word in the whole document. IDF allows us to concentrate on words that are less frequent in the whole corpus or ‘n’ data points. While TF concentrates on the words that are more frequent in a text or a data point of the corpus. TFIDF gives more weightage or importance to words that occur rarely in the corpus and frequently in a text. IDF uses log of the inverse ratio of the probability of finding the word across the corpus.

5. Why use log in the IDF? Usage of logarithm can be understood from Zipf’s law which as a summary states that words in a language follow power law distribution where some words such as ‘the’ are frequently found in usage, while some words such as ‘civilization; are less frequent in usage. Taking log will bring down the dominance of this rare words dominating in the computations for similarity.

6. Explain about Word2Vec? This is a vectorization method for words where each word is vectorized into a d dimension dense vector. If two words are semantically similar then vectors of these words are closer geometrically.

7. Explain about Avg-Word2Vec, tf-idf weighted Word2Vec? For words we can get vectors from word2vec. These vectors then are averaged to get the vector form for the whole statement or the text of the data point. Avg-word2vec considers simple average of the sum of vectors of the words present in the text while tf idf-weighted-word2vec outputs a sum of multiplication of tf idf value of the word and the word2vec of the word.

Deep Learning

1. Explain about Multi-Layered Perceptron (MLP)? A perceptron is a neural network unit. An MLP is a feed forward neural network that is composed of multiple layers of perceptron. Neurons are stacked to form a layer and layers are stacked to form a neural network. It consists of one input layer, one output and at least one hidden layer. Each neuron except in the input layer uses a non linear activation function. With MLPs we can have complex mathematical functions to act on input to generate an output. MLP is a graphical way of representing functional compositions (f(g(x)) : f of g of x).
2. How to train a single-neuron model? Training in neural networks implies learning weights of edges between neurons. The loss function such as RMSE for the output should be defined initially. Using the loss we can define an optimization problem for the network to work upon. Training mainly consists of solving this optimization function. And this is achieved using Stochastic Gradient Descent. Weights are randomly initialized and through SGD weights are updated on every epoch to reach an optimum value for the optimization function. SGD consists of computing derivatives of the loss function with respect to the weights. A learning rate is generally used to control the update size for each weight.
3. How to Train an MLP using Chain rule? For training MLPs backpropagation is used. The process is similar to training a single neuron. Gradients of loss function with respect to weights are computed using chain rule. (Tip: Follow the path of the edges from loss function to the weight to use chain rule for computing gradient of the loss function with respect to that weight).
4. How to Train an MLP using Memoization? While computing gradients some gradients occur a lot such as the gradient of the loss function to the output of the neurons of the previous layer. Using memoization we should avoid computing the same gradient repeatedly. Memoization allows us to compute gradients once and look up a hash function to retrieve the gradient whenever needed.
5. Explain about Backpropagation algorithm? Backpropagation is composed of chain rule and memoization. After defining the loss function, we initialize weights of the neuron edges randomly. We forward pass the input to compute the value of loss. Then using chain rule and memoization we back propagate the loss through edges to update weights. The forward and the backward passes are repeated to converge to an optimum loss value. Note that back propagation works only if the activation functions of each neuron are differentiable. On the whole, speed of the training depends on computing derivatives.
6. Describe Vanishing and Exploding Gradient problems? In MLPs we have a large number of layers and due to chain rule multiplication of gradient values that are less than 1 results in the overall gradient to be small (or too large for gradients > 1). This will not help in training the

MLP. Vanishing gradients - partial derivatives <1 - no weight update, exploding gradients partial derivatives >1 - no convergence.

7. Explain about Bias-Variance tradeoff in neural Networks? Bias- Variance tradeoff in Neural Networks is controlled with the number of parameters in the network in turn by the number of layers. As the number of weights will be high we will have overfitting of the Network on the training data. We can avoid overfitting by using Regularizations or Dropout. Overfitting implies that the network is having high accuracy on the training set but fails miserably on the test set.

8. What is sampled softmax? Sampling the target vector to compute the value of the loss function. For Natural Language Processing or equivalent purposes, the target consists of large dimensionality. The vocabulary for language translation can be in 10,000s. Thus, the output from the neural network will be of large size for each datapoint. Computing softmax functions over such a target is computationally intensive. To avoid this, sampled softmax technique introduces looking only at correct values in the output and a random sample of incorrect values, words in case of NLP. This is a subset of the target vector and an approximate value of loss function can be determined. Introduced by Sebastien Jean et al. in 2015. During inference, sampled softmax cannot be used, as it requires knowing the target. Using sampled softmax allows training models faster over a huge number of target classes, compared to usual softmax. This is used during training only, generating an underestimate of the full softmax loss. For inference softmax is applied using it on a full set.

9. Why is it difficult to train a RNN with SGD? SGD in RNN generally results easily in vanishing and exploding gradients. This makes the model impossible to train. For exploding gradients we can use gradient clipping and for vanishing gradients we can use a soft constraint.

10. How do you tackle the problem of exploding gradients? By gradient clipping

11. What is the problem of vanishing gradients? Neural Networks doesn't tend to remember much things from the input layer due to farness from the output layer and the loss function, multiplication of values less than 1

12. How do you tackle the problem of vanishing gradients? By using LSTM, LSTM retains previous cell state, For MLPs we can use proper activation functions such as ReLU which can result in dead state

13. Explain the memory cell of a LSTM. LSTM allows forgetting of data and using long memory when appropriate. A cell of LSTM contains 3 gates that control the flow of data. These are: the forget gate, input gate and sigmoid or the output gate. The forget gate decides on the amount of information to be passed from previous cell states, the input gate adds value to the output of the forget gate to make it useful for the next cell, while the output gate provides the output for the cell. (Reference included as it is very important: http://colah.github.io/posts/2015-08-Understanding-LSTMs/)

14. What type of regularization do one use in LSTM? Weight regularization based on L1 norm or L2 norm is used in LSTM.

15. What is the problem with sigmoid during backpropagation? We will have vanishing gradients with sigmoid activations. Very small, between 0.25 and zero. Results in no training.

16. What is transfer learning? Transfer learning helps us use an already trained model to work on our new dataset instead of building from scratch. The major advantage is in extracting basic features from the images as edges and other image features need to be extracted from all images irrespective of the dataset. Training a new model from scratch will approximate the performance of the pre trained models. The datasets that we have generally come in a small number of data points. Training the model from scratch with this new dataset will require a lot of time and the model generally overfits on the training set and fails to generalize well on the test set.

17. What is Backpropagation through time? In RNN, backpropagation is unrolled over time. RNN has layers repeating over time. Backpropagation refers to using chain rule with memoization and weight updates. The input for RNN is a sequence data where RNN works on parts of each input at each timestep. BPTT acts as Back propagation for the unfolded RNN.

18. What is the difference between LSTM and GRU? LSTM has three gates forget, input and output. The Gated Recurrent Unit contains no separate output gate. Cell states of the previous cell are merged into output of the current cell state to provide full cell state as output. Additionally a gate controller is available to control the forget gate and the input gate. Controlling implies multiplying the values with a multiplier that scales the value such as multiplying the cell state of the previous cell with 0.1 will allow us to retain only 10% of the information from the sequence so far.

19. Explain Gradient Clipping. Gradient clipping is a regularization technique to control exploding gradients problem. We can use L2 norm gradient clipping where each of the gradients are divided by the sum of squares of the gradients to clip all of them to 1. Multiplying these gradients with a threshold will allow us to clip all the gradients to the threshold.

20. Adam and RMSProp adjust the size of gradients based on previously seen gradients. Do they inherently perform gradient clipping? If no, why? Yes Adam and RMSprop clip gradients inherently. The thresholds for clipping can also be defined. The gradient has division by root of the average of the square of gradients.

21. External sources: https://www.analyticsvidhya.com/blog/2017/01/must-know-questions-deep-learning/

More Questions on Machine Learning and Deep Learning

1. How do you decide which algorithm to use for a given dataset? What is your criteria to narrow down on a list of suitable Machine Learning Algorithms? EDA tells us about which algorithm to narrow down to. KNN is suitable when distance matrices are readily available, Logistic Regression and SVMs require Linearly separable data points, Naive Bayes works as a base learner, Decision Trees can perform well on all types of data, Neural Networks are less interpretable. Selecting a model depends on the data and its EDA can be exploited to choose suitable candidates.

2. You are given a data set. The data set has missing values which spread along 1 standard deviation from the median. What percentage of data would remain unaffected? Why? 1 standard deviation range for a Gaussian distribution covers ~68% of the data points. If the 68% of data points are affected by missing values, then 32% (100 - 68) of the data would remain unaffected.

3. You are given a data set on cancer detection. You've built a classification model and achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it? Datasets on diseases and the health care sector are generally imbalanced. A dumb model can result in high accuracies by predicting all patients having no disease. Accuracy is not a good measure for these datasets. Using AUROC, precision, recall and f1_scores will give a better estimation of the performance of the model.

4. You are assigned a new project which involves helping a food delivery company save more money. The problem is, the company's delivery team isn't able to deliver food on time. As a result, their customers get unhappy. And, to keep them happy, they end up delivering food for free. Which machine learning algorithm can save them? This problem has no pattern in the data regarding food delivery. The problem is strictly a route optimization problem for which Machine Learning algorithms cannot be applied. A machine learning algorithm can be applied when there is a pattern, data and when the problem cannot be solved mathematically.

5. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why? Ensemble model techniques such as Bagging can be used to reduce variance to allow the model to have good generalization ability. We can also use regularization which penalizes for high variance and also can pre process data using dimensionality reduction techniques or selecting k best features.

6. After spending several hours, you are now anxious to build a high accuracy model. As a result, you build 5 GBM models, thinking a boosting algorithm would do the magic. Unfortunately, neither of the models could perform better than benchmark score. Finally, you decided to combine those models. Though, ensembled .are known to return high accuracy,

but you are unfortunate. Where did you miss? Ensemble model requires base learners to be as different as possible. The base models should be uncorrelated else they will learn the same aspects as each other from the dataset and do not combine to become a powerful model.

7. You've built a random forest model with 10000 trees. You got delighted after getting a training error of 0.00. But, the validation error is 34.23. What is going on? Haven't you trained your model perfectly? Low training error indicates overfitting. The random forests have perfectly learnt the pattern in the training data but failed to generalize well on test and validation data. Tuning the number of base learners can reduce overfitting.

8. You've got a data set to work having p (no. of variable) > n (no. of observation). Why is Ordinary Least Square a bad option to work with? Which techniques would be best to use? Why? This is due to the curse of dimensionality. The value of least squares tends to infinity as the dimensionality increases. Regularized least squares optimization function will work well in this case. Specifically L2 regularization will work as it heavily penalizes large variances.

9. You have built a multiple regression model. Your model R^2 isn't as good as you wanted. For improvement, if you remove the intercept term, your model R^2 becomes 0.8 from 0.3. Is it possible? How? R^2 the coefficient of determination moving from 0.8 to 0.3 implies that the model is moving towards a simple mean model which has $R^2 = 0$ and outputs the mean of the training set for all query points. The removal of intercept means that the output or the target variable is free from mean or it has made to be of 0 mean. When the mean of the target becomes 0 the sum of squared errors increases in turn reducing R^2. Note: $R^2 = 1 – \sum(y – y')^2/\sum(y – y\_mean)^2$.

10. You are given a data set consisting of variables having more than 30% missing values? Let's say, out of 50 variables, 8 variables have missing values higher than 30%. How will you deal with them? We can remove the features, create a new feature to indicate whether the value is missing or we can do missing value imputation using mean, median or filling them with a new category such as 'Unknown'.

11. Which data visualisation libraries do you use? What are your thoughts on the best data visualisation tools? I use matplotlib and seaborn in Machine Learning to do EDA and to compare the performance of the models in various situations. Best data visualisation tools should be easy to use and provide excellent documentation regardless of the size of the dataset.

12. How can we use your machine learning skills to generate revenue? Machine Learning allows us to find patterns in the data. It can thus make predictions on future datasets with satisfactory performance. This will allow us to make decisions prior to the occurrence of a data point. Thus the skills generate revenue to the company.

13. What are the last machine learning papers you’ve read? The last research paper was on generating 6 Degrees Of Freedom for estimating orientation of an object from an image. A 3D detection box is produced around the object taking its 3d model for reference. The roll, pitch and yaw angles along with the coordinates are estimated based on projecting the 3d model on a 2d sheet. Mask R-CNN is used to model this problem by the authors of the research paper. I am yet to implement this model.

14. Do you have research experience in machine learning? No.

15. What are your favorite use cases of machine learning models? NLP and Anomaly detection: text classification, classifying transactions, demand predictions

16. Where do you usually source datasets? Open source datasets: such as kaggle and directly from company websites or competition sites.

17. How do you think Google is training data for self-driving cars? Google is collecting data from sensors, images from camera and the GPS. While it is important to have labels for this data, we can see the labels being generated using Image Captchas to learn the objects in the images.

18. What do you understand by Type I vs Type II error? False Positive vs False Negatives.

19. You are working on a classification problem. For validation purposes, you’ve randomly sampled the training data set into train and validation. You are confident that your model will work incredibly well on unseen data since your validation accuracy is high. However, you get shocked after getting poor test accuracy. What went wrong? The dataset might have imbalanced classes where random sampling did not make validation and train sets to have similar class distributions. Stratified sampling will help us in this case. Stratified sampling will help us get a consistent proportion of classes in training, validation and test dataset.

20. State the universal approximation theorem? What is the technique used to prove that? It states that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on compact subsets of $\mathbf{R}^n$, under mild assumptions on the activation function. The derivatives of the feedforward network can also approximate the derivatives of the underlying function arbitrarily well. Universal approximation theorem indicates that Neural Networks with certain properties can approximate any complex function of transforming input to output. https://arxiv.org/pdf/1910.03344.pdf

21. Given the universal approximation theorem, why can't a MLP still reach an arbitrarily small positive error? UAP generally requires diverse activation functions. MLPs may fail to learn the parameters to generalize well and generate small positive errors. In addition MLPs can overfit on training dataset and might choose a wrong function to represent the dataset.

22. What is the mathematical motivation of Deep Learning as opposed to standard Machine Learning techniques? Machine Learning was built on theory first and experiments second for development. While Deep Learning allowed people to develop models first and do

experiments and then based on the results theory was built. Deep Learning takes direct motivation from the human biological brain that has connected neurons which can approximate any mathematical function. Deep Learning gets its motivation from the limitations of Machine Learning techniques to measure the similarity between test data points and the train data points, and that Deep Learning models have easily succeeded in measuring these similarities with small errors.

23. In standard Machine Learning vs. Deep Learning, how is the order of number of samples related to the order of regions that can be recognized in the function space? As the dataset size increases Deep Learning performs well over Machine Learning. The order of regions for DL depends exponentially O(2^k) on the number of samples and for ML, it depends linearly. It can be thought in terms of the number of functions that the model can fit on the dataset.

24. What are the reasons for choosing a deep model as opposed to a shallow model? 1. Number of regions O(2^k) vs O(k) where k is the number of training examples 2. # linear regions carved out in the function space depends exponentially on the depth.

25. How Deep Learning tackles the curse of dimensionality? Deep Learning assumes that data is generated by composition of functions at multiple levels. This allows us to exponential gain over the order of regions which tackles the challenges raised by the curse of dimensionality.

26. How will you implement dropout during forward and backward pass? It is an inexpensive yet powerful regularization method that prevents the model from overfitting. Each neuron in the training phase is present with a probability ‘p’ which is randomly generated and thresholded. With dropout we do random sampling of the neurons at every iteration for training of the model. This will prevent the model from training all neurons or having all weights updated at each iter.

27. What do you do if Neural network training loss/testing loss stays constant? 1. Check for code errors, 2. If the loss is constant it means the gradients have vanished, change the activation functions, 3. Tune the hyperparameters (number of layers, neurons, type of activation function) of the NN, 4. Check for data normalization, 5. Try transfer learning.

28. Why do RNNs have a tendency to suffer from exploding/vanishing gradients? How to prevent this? Gradients in RNN depend on the length of sequences of the data. Larger length results in multiplication of gradients to explode or vanish. We can prevent this by using LSTM NN which has a forget gate that allows us to retain the information that is important in long sequences. It converts multiplication of gradients into addition. We can use gradient clipping to deal with exploding gradients, where the weights are clipped element wise or by norm. While for vanishing gradients we might require to go for LSTM or GRUs.

29. Do you know GAN, VAE, and memory augmented neural networks? Can you talk about it? GAN: Generative Adversarial Networks is based on game theoretic scenario where the generator has to compete with adversary. The generator network attempts to make copies of data points and the discriminator network attempts to distinguish samples drawn from training

data and the samples generated by the generator network. It can be visualized as a zero-sum game where the discriminator gives a probability that the samples generated are from training data. VAE: Variational AutoEncoders: AutoEncoders are used to generate lower dimensional representation of the training data. Variational AE adds a probabilistic approach to the dimensionality reduction method. MANN is an RNN with an external memory unit which is used for computations. MANN has been shown to learn faster and generalize well than LSTMs.

30. Does using full batch means that the convergence is always better given unlimited power? In case of a convex loss function it is useful to increase the batch size to converge faster to the optimized solution. With non-convex optimization functions full batch training may get stuck at local minima, local maxima or at saddle points. A mini batch can escape these points by making random updates towards the optimum solution. It also takes care of the outliers or the noise present in the dataset by training with mini batches.

31. What is the problem with sigmoid during backpropagation? Derivative of the Sigmoid function lies between 0 and 0.25. During chain rule multiplication the gradients will reach 0 resulting in vanishing gradients and no weight updates.

32. Given a black box machine learning algorithm that you can't modify, how could you improve its error? As the model can't be modified we cannot tune its hyperparameters nor we can use an ensemble. More data can be added, Data pre processing can be rechecked to treat missing values, outliers, and various feature engineering techniques can be applied to see which can improve the error.

33. Why do we NOT use Perceptron as often as a Logistic-Regression or Neural-Network? Perceptron is an artificial neuron using a step function as an activation function which is not differentiable. Logistic Regression or Neural Networks have non-linear activation functions which are differentiable.

34. How does ReLU introduce non-linearity when it looks "linear"? It is easy to observe that Sigmoid introduces non-linearity as it is a nonlinear function. RELU's output is not a straight line. It bends at the x axis. With the non-linearity of RELUs, arbitrary shaped curves can be built. **g(z) = max{0, z}.** (Non differentiable at 0). Ref : https://www.quora.com/Why-is-ReLU-non-linear

35. Why do we prefer dropout for regularization? Why not simply use L2 or L1 reg like in LogisticRegression? At every iteration dropout randomly selects some nodes and removes them along with all of their incoming and outgoing connections. So each iteration has a different set of nodes and this results in different sets of outputs which is similar to ensemble technique in machine learning. Due to this, dropout is preferred when we have a large neural network structure. Ref : https://leimao.github.io/blog/Max-Pooling-Backpropagation/

36. How does dropout work at test-time? If a dropout layer of p=0.5 is added during training, then the trick is to multiply the output of the last hidden layer by 1-p such that the output does not get impacted. Ref: Reference

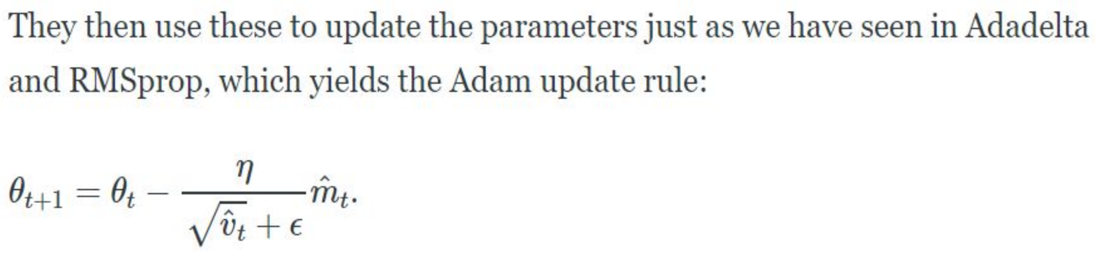
37. Why is Max-pooling popular in CNNs? Why not any other function like mean, median, min etc? The choice of pooling operation is made based on the data at hand. Average pooling method smooths out the image while max pooling extracts the most important features like edges. Ref : reference

38. Why do individual learning rates per weight (like in AdaGrad) help as compared to one learning rate for all weights? Adagrad performs smaller LR updates for parameters (weights,biases) associated with frequently occurring features and high LR for infrequent features. It uses a different LR for every parameter at every step in order to converge faster(reach a local/global minima) when compared to SGD which uses a single LR for a training sample.

39. Write the weight update functions for Adam.When does ADAM behave like Adadelta? Adadelta stores an exponentially decaying average of past squared gradients(vt). Adam, in addition, also keeps an exponentially decaying average of past gradients similar to momentum(mt). Adam behaves like Adadelta if there is no mt in its weight function. Ref

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$
$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}$$



They then use these to update the parameters just as we have seen in Adadelta and RMSprop, which yields the Adam update rule:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t}+\epsilon}\hat{m}_t.$$

40. Number of parameters & hyper-params in a max-pooling layer? 9. How do we differentiate and back-prop through a max-pooling layer? Parameters (Weights, biases) is 0 for a max-pool layer. Hyper-parameters: Filter size, # of filters, stride, padding, Backpropagation: While back propagating through a max pool layer, the gradient is 1 for the neuron with max value and 0 for all other neurons which means there is no gradient for non-max values. Ref : https://leimao.github.io/blog/Max-Pooling-Backpropagation/

41. You are training a model and its train loss is not changing from epoch to epoch. What could be the possible reasons? Weights are not being updated, due to which train loss remains unchanged. If a RELU is being used and negative values are sent, then most of the neurons are not active and the model is not being trained. While training, the model would have reached a saddle point or minima and further training is not changing the loss.

42. What's the loss function of an autoencoder? Mean Squared Error is typically used as a loss function while training an autoencoder. It's based on the problem we are solving. Ref : ref

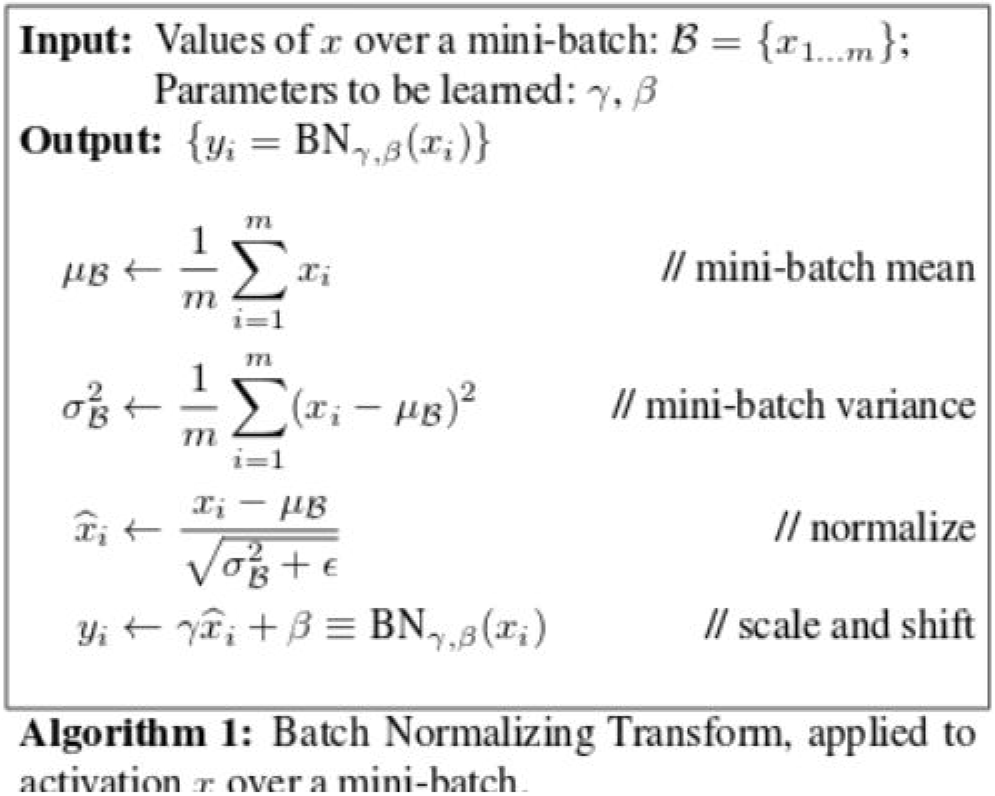
43. Why is hierarchical softmax used in Word2Vec? To reduce the computational complexity of Word2Vec from O(V) to O(logV) use hierarchical softmax. V is the size of the vocabulary: ref

44. How do we update weights in a Negative sampling based training of Word2Vec model? In Negative sampling based training of Word2Vec, we consider 1 positive pair and K negative pairs as 1 training sample. For a center word as input, we perform C (window size) weight updates to predict the context word vectors. With negative sampling only a fraction of word vectors in the output weight matrix are updated. This will reduce the computational complexity.

x=one hot encoded center word representation, W-input=N-Dim vector representations of all words,h= hidden layer, W-output=Output weight matrix

1. In Forward Propagation : x.W-input=h; sigmoid(W-output*h) is the predicted vector for 1st neighboring context vector.
2. In Backward Propagation : Positive samples will have 1 as y and negative samples will have 0. Compute ypred-ytrue to get the prediction loss. Compute new weights using SGD formula and use them for the next positive pair. Ref : ref

45. What are the trainable prams in a BatchNorm layer? Ref, BN affects the output of the previous activation layer by subtracting the batch mean and dividing it by batch's standard deviation. These are the 2 trainable params that BN adds to a layer.



Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\ldots m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_\mathcal{B} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_\mathcal{B}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_\mathcal{B})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_\mathcal{B}}{\sqrt{\sigma_\mathcal{B}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

46. Why is ResNet able to learn models of significantly larger depth than earlier VGGNet? ResNet is able to learn models of larger depth as its architecture has skip connections due to which there is no vanishing gradient problem while VGG has vanishing gradient problem.

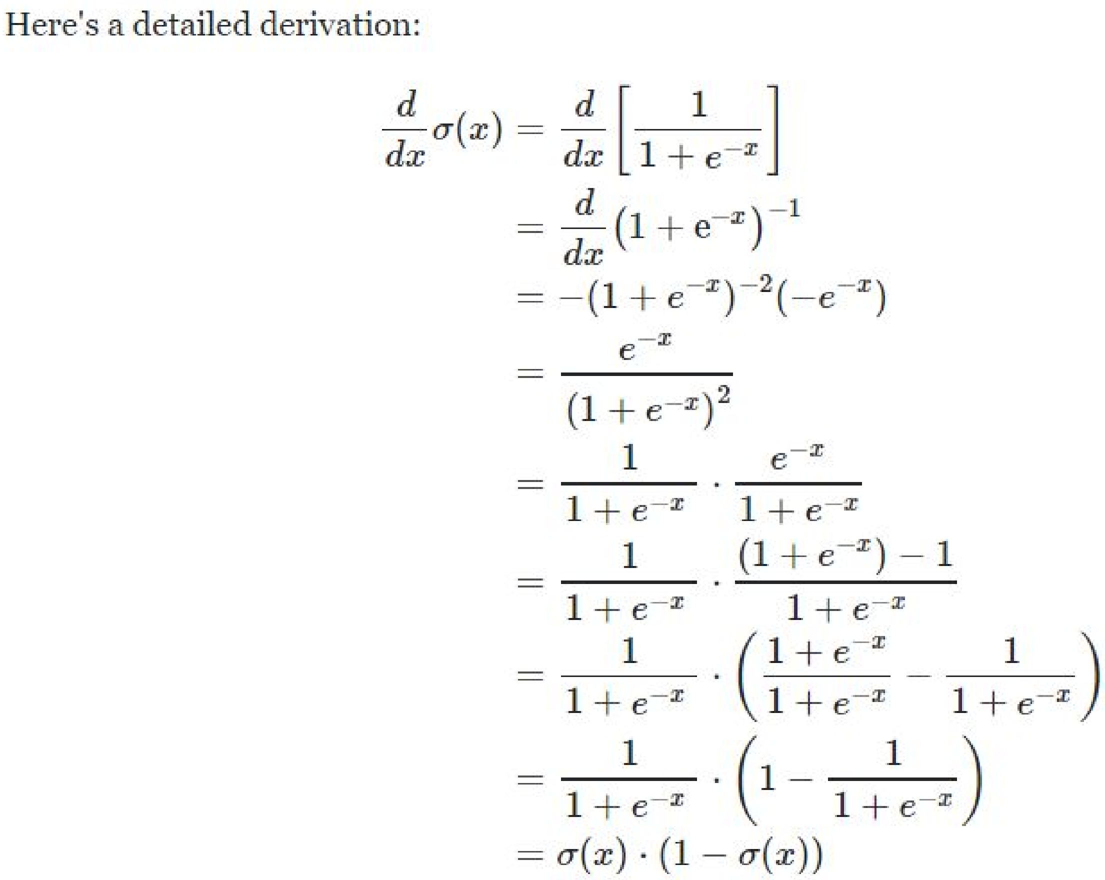
47. You have 5000 images with 5 class-labels and you want to build a CNN model? How would you go about building a classification model? We can create a sequential model, add convolutional layers, maxpool,batch normalization, dropout and finally pass it through a Dense layer(5, activation='softmax').

48. Why does data augmentation help in object recognition tasks? Ref, Image data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset. Doing this way, it helps reduce overfitting and improves generalization.

49. Why does a GPU help in deep-learning much more than a multi-core CPU with say 8 or 16 cores? GPU (Graphics Processing Unit) have a large number of simple cores which allow parallel computing through thousands of threads computing at a time. Whereas CPUs have few complicated cores which run processes sequentially with few threads at a time. Bandwidth is one of the main reasons why GPUs are faster for computing than CPUs. The High bandwidth, hiding the latency under thread parallelism and easily programmable registers makes GPU a lot faster than a CPU. CPU can be used to train the model where

data is relatively small. GPU is fit for training the deep learning systems in a long run for every datasets.CPU can train a deep learning model quite slowly. GPU accelerates the training of the model. Ref : ref

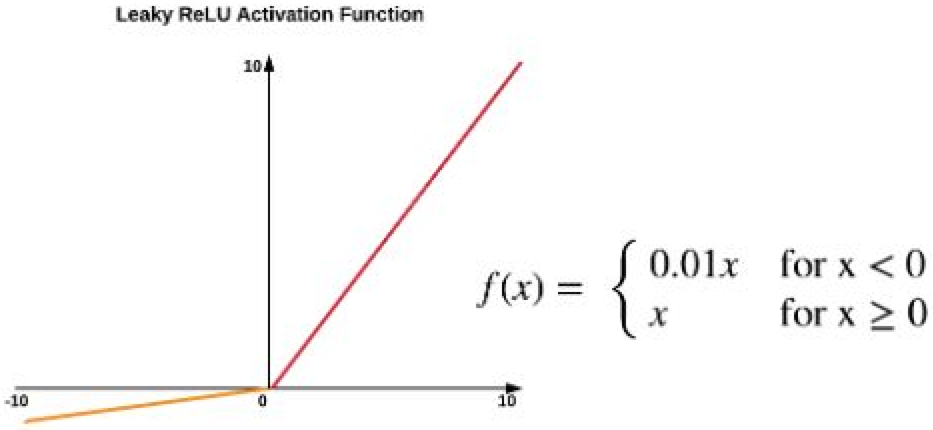
50. Derive the derivative of a sigmoid function? ref



Here's a detailed derivation:

$$
\begin{aligned}
\frac{d}{dx}\sigma(x) &= \frac{d}{dx}\left[\frac{1}{1+e^{-x}}\right] \\
&= \frac{d}{dx}\left(1+\mathrm{e}^{-x}\right)^{-1} \\
&= -(1+e^{-x})^{-2}(-e^{-x}) \\
&= \frac{e^{-x}}{\left(1+e^{-x}\right)^2} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \frac{(1+e^{-x})-1}{1+e^{-x}} \\
&= \frac{1}{1+e^{-x}} \cdot \left(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right) \\
&= \frac{1}{1+e^{-x}} \cdot \left(1 - \frac{1}{1+e^{-x}}\right) \\
&= \sigma(x)\cdot(1-\sigma(x))
\end{aligned}
$$

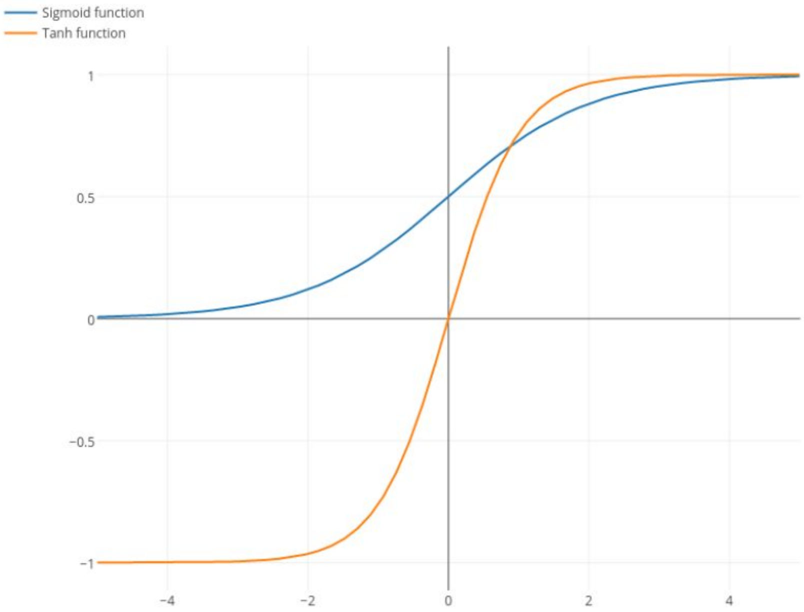
51. Why do we need Leaky ReLu? ref



$$f(x) = \begin{cases} 0.01x & \text{for x} < 0 \\ x & \text{for x} \geq 0 \end{cases}$$

Leaky Relu's are a solution for dying Relu's. A RELU neuron is dead if negative values flow through it as the slope of RELU in the negative range is always zero. This ends up with the network having many dead neurons resulting in sparsity. The dying problem is likely to occur when the learning rate is too high or there is a large negative bias. Lower learning rates can be used to mitigate the problem or variant of RELU which is Leaky RELU can be used. Leaky RELU has a small slope for negative values, instead of complete zero which will avoid dying neurons.

52. Why is tanh (sometimes) better than sigmoid for training a NN?



a. Sigmoid updates are slower than tanh updates as the gradient of sigmoid has a maximum at 0.25 while tanh derivative can reach 1, making the updates larger.

b. Convergence is usually faster if the average of each input variable over the training set is close to zero. Thus the inputs are normalized. If sigmoid is used, either

53. How to fix exploding gradients in a MLP?

Reference: https://machinelearningmastery.com/exploding-gradients-in-neural-networks/

Exploding gradients occur when large gradients accumulate and result in very large updates to a NN model during training. The model becomes unstable and does not learn from the training data.

The explosion occurs through exponential growth by repeatedly multiplying gradients through the network layers that have values larger than 1.

The model will have poor loss or become NaN. The weights update a lot without settling.

Fixing the explosion:

a. Re-Design the Network Model: fewer layer, smaller batch size, truncated Backpropagation through time

b. Using LSTM network for RNNs as the gates will control the gradients.

c. Using Gradient Clipping

54. You want to detect outliers using Neural Nets? How would you go about doing it? Autoencoders work well for Outlier detection tasks in addition to Dimensionality reduction. It tries to minimize the reconstruction error (L(x,x')~square(||x-x'||). Once the model is trained and values are predicted using an autoencoder, we can filter/find the outliers by finding the indices where error > 0.1. Ref : ref

55. How to find the best hyper parameters? We use cross validation to select the best hyper parameters. Cross validation can be done using Grid Search or Random Search. Input: Set of hyper parameters with list of possible values for each, Grid Search:

56. What is transfer learning? Models trained on a dataset can be utilised to make predictions on another but related dataset. A model trained on cars can be used to identify trucks. This is also applied when training models from scratch is infeasible or when there is not enough data.

57. Compare and contrast L1-loss vs. L2-loss and L1-regularization vs. L2-regularization. L1-loss is the least absolute deviations from predicted values to actual values while L2-loss has squared errors. L2-loss is impacted by outliers but provides a single solution, while L1-loss being robust it can have multiple solutions. Loss function is used for optimization to provide the best solution. While regularization is used to control overfitting. Regularization in terms of absolute value of weights is added to loss functions which makes that loss function is minimized when error is minimized in addition to the minimization of weights of the model. L2-regularization is computationally efficient due to the availability of analytical methods. But L1 regularization is suitable for sparse data as it will avoid calculating weights for sparse features.

58. Can you state Tom Mitchell's definition of learning and discuss T, P and E? A computer program is said to learn from experience E with respect to some clash of tasks T and performance P, if its performance at tasks in T, as measured by P, improves with experience E.

59. Consider linear regression. What are T, P and E? T: Predict the value of a continuous variable value for each data point, P: Correctly Predicting a value for the continuous variable, E: a database of training examples with label information.

60. What can be different types of tasks encountered in Machine Learning? Classification, Regression, Recommendation, Clustering

61. What are supervised, unsupervised, semi-supervised, self-supervised, multi-instance learning, and reinforcement learning? **Supervised:** Training data contains dependent (target = label) and independent (features) variables, Model gets trained by showing examples,

Unsupervised: Training data contains independent variables, Model gets trained to form groups of training data, **Semi-supervised:** some of the data points have target information while some do not have target information, used when collecting target data is infeasible for all data points), **Multi-instance learning:** Each data point has multiple dependent variables (Text tagging), **Reinforcement Learning:** Training models with rewards and penalties using which the model trains its decision process, it tries to maximize its reward, this is different from Supervised and Unsupervised Learning, it does data collection and model evaluation, all on its own.

62. Loosely how can supervised learning be converted into unsupervised learning and vice-versa? Supervised Learning can be converted to Unsupervised Learning by discarding the label of the data points. Unsupervised Learning can be converted to Supervised Learning by training an Unsupervised model for a fewer number of data points, manually label clusters by analysing the data points in each cluster, the whole cluster is assigned a label. Using this data with labels we can do supervised learning.

63. Derive the normal equation for linear regression. The objective of Linear Regression modeling is to determine the weights of the line that generalises the dataset with least error. The loss function generally used is Least Squared error. Normal equation that defines the weight vector of the model is given as: $\theta = (X^TX)^{-1} * (X^TY)$. Predicted value for a data point is given by: $y\_pred = \theta^TX$. The cost function for linear regression model is Cost = $(1/2m)*$ summ(1 to m) $(X\theta - Y)^2$, m = number of training data points. Now, the equation can be transformed into: Cost = $\theta = (X\theta - Y)^T * (X\theta\text{-}Y)$. Optimizing this cost function for minimum we have gradient: grad(Cost, θ) = $2(X^TX)\,\theta - 2(X^TY)$. And for minimum value for the cost function the gradient should be 0. $2(X^TX)\,\theta - 2(X^TY) = 0 \rightarrow (X^TX)^{-1}\,(X^TX)\,\theta = (X^TX)^{-1} * (X^TY)$.

64. Discuss training error, test error, generalization error, overfitting, and underfitting. *Training error*: the error between predicted and true values of the data that is used to train the model. *Test error*: Predictions generated on test data by a trained model are compared with the true values of the test data. *Generalization error*: Error between prediction and true values of the target variable for previously unseen data. *Overfitting* occurs when Train error is low, test error and generalization error are high, *Underfitting* occurs when all train, test and generalization errors are high.

65. What do you mean by affine transformation? Discuss affine vs. linear transformation. Affine transformation is a composition of linear transformation and a translation. It maps points in a space to another space, maintains or preserves collinearity and parallelism while changing the origin of the data points. Linear transformation is a sub element of affine transformation where the origin is fixed and carries all other characteristics of affine transformation.

66. Compare representational capacity vs. effective capacity of a model. The Representational Capacity is a family of functions the learning algorithm specifies when varying the parameters

in order to reduce a training objective. Effective capacity specifies the function the learning algorithm has selected to approximate the function that optimizes on the training error.

67. Discuss VC dimension. Vapnik-Chervonenkis dimension is used to quantify model capacity. It measures the capacity of a binary classifier. It is the largest possible value of m for which there exists a training set of m different data points that the classifier can label arbitrarily. The definition of VC dimension is: if there exists a set of n points that can be shattered by the classifier and there is no set of n+1 points that can be shattered by the classifier, then the VC dimension of the classifier is n. VC dimension of 3 data points is 2. While VC = 3 for 4 data points. For 3 data points if we take all its geometric distributions the three points can be separated by a binary classifier by a 2D line. The data points are assumed to be non-collinear.

68. What are nonparametric models? What is nonparametric learning? Algorithms that do not make strong assumptions about the mapping function from independent features to dependent features are called nonparametric models. Ex: kNN. These models assume that the data distribution cannot be defined in terms of a finite set of features. Nonparametric learning does not have a training phase. They utilise a similarity metric to make predictions.

69. What is an ideal model? What is Bayes' error? What is/are the source(s) of Bayes errors? An ideal model simply is equipped with the underlying probability distribution that generates the data. Bayes error is the lowest possible error that is analogous to irreducible error. This arises in an ideal model due to random noise.

70. What is the no free lunch theorem in connection to Machine Learning? In machine learning there is no model that can fit or work best for all problems. A model that best works for one problem tends to be not so good for another problem.

71. What is regularization? Intuitively, what does regularization do during the optimization procedure? (expresses preferences to certain solutions, implicitly and explicitly) By regularization we add information to the process to tackle an ill-posed problem, it expresses preferences to one of the solutions that does not overfit on the data sets. During optimization regularization controls the growth of optimization parameters which can happen haphazardly.

72. What is weight decay? Why is it added? Weights are multiplied by a factor <1.This prevents weights from growing too large.

73. What is a hyperparameter? How do you choose which settings are going to be hyperparameters and which are going to be learnt? (Hyperparameters are either difficult to optimize or not appropriate to learn - e.g. learning model capacity by learning the degree of a polynomial or coefficient of the weight decay term always results in choosing the largest capacity until it overfits on the training set) Hyperparameters are the parameters or arguments of the model that remain constant while training.

74. Why is a validation set necessary? When we train a model with some parameters, we need to check how it performs on the unseen data, thus to check that we use validation set, if the

model under-performs we update the parameters, and keep doing it until we get good performance on the unseen data i.e. validation set. Once we get good output, we can be sure that our model is well trained and can perform well on new unseen data. **Inshort, we use Validation Set to select perfect hyperparameters for our model given the data.**

Programming Exercises

Interview Experiences

Candidate: Karthik Kumar

Companies Attended:

1. Automated Returns
2. Ogenie
3. MasteryGo
4. Meslova
5. GreenDeck
6. Precily
7. Gibbr Technologies

Outside AAIC:

1. Tekflo
2. PayOK
3. Turing

Company: PhD interviews at IITB (IEOR and CS departments for AI and ML subjects)

Candidate: Karthik Kumar Billa

Date: 4th and 5th December 2019

Self Assessment: Performance: 3 different departments (Computer Graphics - 20% performance, AIML 70% performance, IEOR - 65%)

A. Computer Science department:

    First round: Written: attempt 3 out of 8 questions, passed due to solutions to conditional probability question. (Indicate (compulsory) 2 of the departments you are interested in, I was interested in AI only, as other departments were Data Structures, Algorithms, Network Design, etc. Computer graphics was relevant and was able to write answer for a filtering based image processing question)

    Second round:

        Subject: Computer Graphics, Questions: Tell me about yourself, why are you interested in this field (honestly said I had no other option and I was able to understand the question in the first round related to this subject only). Do you know the Math behind CNN as you have worked on CNN on MNIST dataset. I did not remember but I gave an approximate idea how CNN works. (Rejected)

Subject: AI, Questions on Probability and AI introduction (IBM’s watson, Deep Blue), etc. Was not able to answer a follow up question in conditional probability. (Rejected)

Subject: IEOR, Questions: Introduction, on Machine Learning project flow, how do you do training, pre-processing of data, What if the target variable is a vector rather than a single value for each data point, What are the loss functions in regression, Dimensionality reduction, (reducing dimensionality without losing information and preserving variance - follow up question: what is information and preserving variance in what). (Rejected)

Company: Automated Returns

Candidate: Karthik Kumar Billa

Month: January

Self Assessment: Rounds 1: 90%, 2: 90%, 3:40%

Round1: Take home assignment: given a dataset, do eda, and build an ML model and DL model, pick a metric and compare performances, write doc strings and comments and add a conclusion.

Round2: Take home assignment based Screening interview: Walk through the solution and showcase your portfolio and he was very much interested in my DL case study. (Impressed)

Round3: With CEO: Was browsing about CEOs past experience and was prepared with his company works. It so happened that he sold his previous company and started a new company and looking to hire new people for this company. The preparation itself became useless. His previous company had worked on time series data and I was preparing for FFT transformation etc. Questions: He gave an introduction to what he was doing and I showed interest in both his previous company and the new company. Asked to explain my portfolio 1 project and had several follow questions (50 minutes). Then he asked 3 questions to compare and have a rating for the interview:

1. If a person loves buying 10 mangoes everyday and found that 3 mangoes were faulty on weekdays and 4 on weekends. What is the percentage of damaged mangoes will you find when you go to the same vendor? The question was tricky. I jumped with a probability approach but the answer was “With a single data point we cannot estimate the trend for a population, it might happen that he goes to the vendor in the morning where the vendor sells previous days leftovers, etc.” (Important to define Business Problem)

2. If your house has 4 walls facing South, What is the colour of the Bear you would see outside your window. I already know the Answer (White, the house is located at the North Pole and the Bear will be a Polar Bear).

3. Profit calculation at the end of 5 years based on year wise loss and profits.

Result: Rejected

Company: Ogenie

Candidate: Karthik Kumar Billa

Month: February

Round1: 65% performance for Data Science role: Take home assignments

3 Tasks:

Task 1: Sentiment Analysis Using Machine Learning Model: Given Dataset:

Performance 100%

Task 2: Find presence of a given array in another array

Performance 100%

Task 3: Build a search engine which helps us to search keywords like java, spring mvc but our job description dataset contain keywords like j2ee, android, hibernate, struct 2, hadoop, big data. Still it is able to show you similar job opening from database by ranking job opening according to keyword weightage

Performance: 10%

Round 1 for Python Developer Role: Take home assignment:

Given an ML model deploy it using Django, 2 tasks only able to develop task 2 solution with 100% performance

Round 2: 50%

Posts applied for: Data Scientist and Python Developer: 12 minutes: Tell me about yourself. Tell me about any project that you have done outside Kaggle (I have none outside kaggle, all are Kaggle based case studies). Asked questions on self case studies. What is the metric used? What is its equivalent? Good until now. Then things started where I have become a dinosaur. What are the current techniques used for textual data preparation specifically for Deep Learning models (I have no proper answer for this)? Stemming, lemmatization, stop words removal are old what are the new ones, wanted to continue with telling one hot encodings, but he moved on to the next question. Which one is better lemmatization or stemming:(my answer: depends on task at hand: for human-like conversations lemmatization is better as it considers presence of root word in dictionary or considers semantic meaning). What framework are you using for deep learning? (keras) How good are you at Model deployment? (I was only able to build a Django based web app during the 1st round of interviews with your company). How do you rate yourself out of 10 your python skills? (5 to 6) Thank you. You will receive mail shortly regarding the results of this interview.

Result: Rejected

Company: MasteryGo

Candidate: Karthik Kumar Billa

Round 1: MCQ: 50% performance
Round 2: for Jr Deep Learning Engineer: face2face interview Topics: LSTM, CNN, Segmentation models, XGBoost loss function(objective argument), RNN vs LSTM, Write code for a given architecture with given loss function(try to get equipped with defining custom loss functions, also know about average, mean, min, etc. layers in keras), try to understand keyword arguments of xgboostclassifier, LSTM. One programming exercise to print numbers of an array in a given sequence without using numpy. Main question: analysis of a video which was captured when students were taking an exam (how do you solve the problem, define everything that you want or you assume, such as manual labelling the data for training set preparation, counting number of people in the video, orientation of face of the student, etc.). What do you know about Attention models? Time: 2 hours, Tip from me: Prepare your notes properly

Result: Not given: Assume Rejected

Company: Meslova

Candidate: Karthik Kumar Bila

3 Rounds:

This is interview experience with Meslova Systems Private Limited: dated 6th March 2020

Location: Hyderabad

Relocation: anywhere in India at client location,

Job roles: Python Developer and ML Engineer (Trainee: 3 months probation)

Status: awaiting results of interview and further communication

Salary: Less than 6 LPA, not yet decided:

Number of rounds attended: 3, Bring own laptop: Start Time: 10:30AM, 3 rounds End Time: 3:30 PM

1st round:

4 sections: Python, SQL, AI, HTML

Use notepad to answer: No need to execute code, don't use internet: 1 hour time limit

Section A: 5 questions on python programming similar to HackerRank Challenges: reverse a string, Fibonacci, etc.

Section B: 4 questions on SQL: Easy to write

Section C: 7 questions on AI (questions are identical to interview questions available at the end of Lecture Videos), effect of multicollinearity on R square and p values, situations where SVM is preferred over Random Forests and vice-versa (linearly separable data for SVM and non-linear data for RandomForest), etc.

Section D: 2 questions to build an html page on :NAV drop down menu and Modal Dialog (I don't know about these things, not attempted)

Attempted Section A, B and C with more than 80% confidence

2nd round:

AI technical discussion Face2Face interview: 1 hour: 2 interviewers one with ML background and other with DL background.

Questions on case studies (UNets, and other architectures) and AIML+DL aptitude, stressed on ability to learn new things and checked for knowledge on current algorithms or ML+DL concepts (Autoencoders, Mask RNN, LSTMs). Tested for ability to understand AIML and DL concepts, asked questions on how to solve ML and DL problems end to end. Interviewer was interested to see hobbies.

Attempt Confidence: 80+%

3rd round:

Python programming skills: asked about programming core questions such as what is an object? what is a class? what is inheritance? What are types of inheritance?

How do you write or define class in a program? And many more.

Write a program for fibonacci series(repeated), how much do you rate your python skills, how much do you rate your ML skills, how comfortable are you with Django and Flask. What is flask? Explain django workflow. As implementation of projects in real time in industry is different from course based case studies and projects, If I say your rating is 5 or 6 how will you justify your ML rating of 8 or 9 (at last answered that I will learn things whatever required to complete or implement projects.)

Attempt Confidence: 55%, Was not easy, but answered almost all questions.

3rd round Discussion:

Interviewer: "Are you willing to relocate anywhere in India? We will consider you as a fresher and you will be facing a probation period of 3 to 6 months. Any questions from you?"

Me: "Will I be able to successfully complete the probation period or should I fear that I will be put off after 3 months. And how is the work here?"

Interviewer: "The work here is wonderful. It will definitely help in the development you are looking for. You will be able to successfully complete the probation."

Me: "I want to experience the industry centered ML programming. I am up for the challenge."

Interviewer: "Our HR will communicate with you for further process."

Conclusion:

Pros: The interview process was encouraging.

Cons: Low salary. Looking for negotiations with HR in the next round.

Result: Rejected

External Resources

My Interview Setup:

1. Tell me about yourself: “They are not asking about you, not about your resume”. “Keep it brief 1 to 3 minutes, no life story”. “Story of related background”. “Start by adjectives”. “How you started, accomplishments, motivation, what you have done, what you can do for the company and why they should hire you”. “Practice this: Make good eye contact, good tone of voice, don’t memorize”.

   Ex 1: My name is Kathik Kumar. I am interested in Artificial Intelligence and its applications. I have done my Post Graduation in Manufacturing Engineering where I was introduced to the field of AI. I have completed online courses to learn the applied nature of AI and ML.

   I have worked on real world problems such as Steel Defect Detection and Credit Card Fraud Detection. Steel Defect Detection was a Deep Learning challenge where the dataset was available in the form of Images. The task was to detect and locate the type of defect present on the steel surface in the image. This is a Supervised Learning problem. I have used pretrained models trained on Imagenet dataset and fine-tuned the model on to this steel dataset. The model gave a performance of 87% on defined metric, the Dice Coefficient. I have developed a working solution for the challenge.
   The Credit Card Case Study was a Machine Learning Challenge where I performed Feature Engineering to input the data to an XGBoost Classifier. The task was to detect whether a transaction is a fraudulent transaction given its details. I have developed a model that achieved a metric score of 93.2% which is equivalent to the top 1.2% solution in the competition. The code for the projects are available on Github and medium.

   As part of the courses I have worked on Natural Language Processing, Unsupervised learning, Image classification, Demand prediction, Malware detection, Apparel recommendation and Facebook friend recommendation. I have learnt coding by competing online.

   I am also a blogger. Additionally I am writing a book which is titled OneStop4ML and also have a goal to establish a Research and Training Institute for AI.

   Presentation and communication are key skills required for passing an interview.

   Your knowledge can help you to score 30%, but your presentation will give you 100.

   What do the recruiters look for? Results, what results did you give? And they want such results at their company. What do they want? And throw an irresistible smile. How can you represent the company to their clients? You can build a powerpoint presentation and post it online. Tell good qualities:

“I have excellent communication skills and also have good organizing skills where I plan things weeks in advance”.

Talk about your approach for completing a project. Strengths: Hard working but not smart? Give examples. Weakness: Getting burnout easily. They can pose a question on loyalty.

1. Walk me through your portfolio: I have worked on two projects which were themed on Credit Card Fraud Detection and Steel Surface Defect Detection. Credit Card Fraud Detection Project is a Machine Learning Project which has an objective to predict whether a transaction is fraudulent. The goal of the project is to provide an optimized Machine Learning Model for this objective. This will help improve performance of the fraudulent transaction alert system. The dataset is a time series dataset with binary classification and comes under Supervised Learning. The success of the project lies in Feature Engineering. As opposed to general time series data projects, the features here are user centric. Initially correlated features are removed through similarities and missing values analysis. Then feature engineering such as label encoding, frequency encoding and mean aggregations were used. It was also seen that users have similar values over a bunch of categorical features. The values of these features are again string concatenated to generate another feature which is again frequency encoded. An XGBoost classifier is directly hyperparameter tuned with 6 fold Cross Validation to generate an AUROC of 95.39% on test set. This generated a top 1% solution for the competition on Kaggle.

   Steel defect detection is tackled with Deep Learning to detect and classify defects in steel using Image Segmentation. We have achieved a dice coefficient of 88% on the unseen test set. Pre-trained models are used to cater for the small size of the dataset. The architecture is modified to suit the problem. Custom layers are added to the neural networks. The training of the models is consistent over training, test and validation datasets.

Appendix

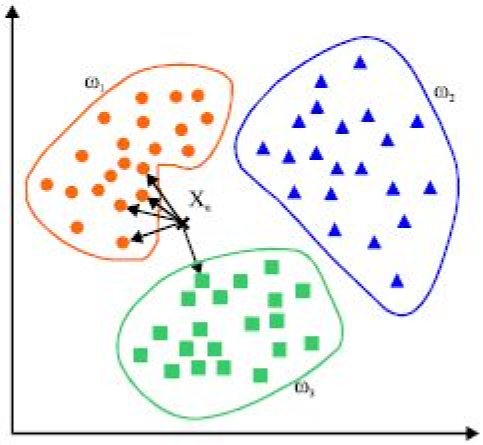
Take home assignment tasks

1 Pager Summary

T-distributed Stochastic Neighbor Embedding:

- **Definition:** t-SNE (t-distributed - looks like normal distribution with fatter tails more outliers) is a machine learning algorithm used for visualization. It is used to visualize high dimensional data in low dimensional plots. Nonlinear dimensionality reduction technique, preserves global as well as local structure.
  - Perplexity: indicates the balance between local and global aspects of the data. Number of close neighbors each data point has. t-SNE produces different outputs with the same inputs at every iteration owing to its stochastic nature.
    - i. Hyperparameter values have to be changed to avoid mis-interpretation of t-SNE results
    - ii. Cluster sizes in t-SNE plot mean nothing (expands dense clusters and contracts sparse clusters)
    - iii. Distances between in the plot may not mean anything
    - iv. Random noise doesn't always look random
    - v. You can see some shapes, sometimes
    - vi. For topology you need more than one point
- **Math:** It constructs a probability distribution over pairs of high-dimensional objects where similar objects (neighborhood) are assigned a higher probability. t-SNE then defines a similar probability distribution in low dimension (embedding - projecting each point from high dimension to low dimension) minimizing KL divergence between the two distributions. It optimizes on KL divergence $\mathrm{KL}\left(P \| Q\right) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$, where p and q are conditional probabilities of two points being in neighborhood in high and low dimensionality respectively.
- **Geometric intuition:** Points in high dimensional clusters are non linearly transformed into low dimensional clusters (random representation). Through the objective function of having similar points together and dissimilar points far from each other neighborhood points are iteratively brought together.
- **Assumptions:** t-SNE is non deterministic. Assumes local structure has linearity (to measure Euclidean distance for similarity.
- **Interpretability:** Less interpretable
- **Outliers:** As p in KL divergence for outliers will be small it has no effect on KL. Thus t-SNE disregards optimizing location of outliers
- **Imbalanced Dataset:** Some of the data points in the majority class in high dimensional space can be considered as neighbors to minority class in low dimension space
- **Applications:** computer security research, music analysis, cancer research
- **Limitations and workarounds:** Stochasticity, non interpretable transformation, does not learn a function to apply tSNE on new data.

K-Nearest Neighbors:



- **Definition:** kNN is a non-parametric method used for classification and regression. It predicts the value of a target variable based on k closest training examples. It is a lazy learner which has no training phase (Uses majority vote, weighted vote or average of k Nearest Neighbors). Computations happen only at run time.
- **Math:** Nearest neighbors are computed using similarity measures such as shortest Euclidean distances (Hamming distance for categorical).
- **Geometric intuition:** prediction of the target variable for a data point in space is influenced by nearest training examples. Taking odd values for K avoids ties in voting
- **Assumptions:** Assumes data points are in metric space and that similar data points are geometrically close to each other
- **Loss Function:** KNN does not have a loss function as there is no training phase
- **Time complexities:** No training, Runtime complexity: O(nd), Run time space complexity: O(nd)
- **Overfitting and Underfitting:** K = 1 - Overfitting, K = n - Underfitting, Select best K using Cross Validation
- **Interpretability:** Highly interpretable as K nearest neighbors can be observed to understand the assignment of a class to a data point
- **Outliers:** decision surfaces are influenced by outliers, outliers needs to be detected and removed
- **Imbalanced Dataset:** we can upsample the dataset
- **Important Hyperparameters:** K and Similarity measure
- **Applications:** Text classification, Recommender systems
- **Advantages:** Easy to understand, no training, less hyperparameters to tune
- **Limitations and workarounds:** KNN assigns class to outliers which is incorrect - remove outliers, large run time space complexity and high latency - use KD-tree or Locality Sensitive Hashing to reduce number of computations for nearest neighbors search
- **Variation:** Weighted KNN: weigh neighbors by the reciprocals of their distance
- **Comparison with other models: vs Naive Bayes:** NB is faster and parametric, **vs Linear Regression:** KNN is better when data has less noise, **vs Support Vector Machines:** SVM is robust to outliers and outperforms KNN when number of features is large and number of training examples is less, **vs Neural Networks:** NN need large training data and has lot of hyperparameters to be tuned.

Naive Bayes:

1. **Definition**: Naive Bayes is a probability based algorithm which uses probability related concepts for classification tasks. Naive Bayes algorithm is based on the Bayes theorem which joins two conditional probabilities.
    - Bayes theorem is given as p(A|B) = p(A) * p(B|A) / p(B) where p(A)= prior probability, p(B|A) = likelihood probability and p(B)= evidence. This theorem is used to get probability of class label for given datapoint in Naive Bayes algorithm.
2. **Math**: Naive Bayes uses Bayes theorem. Let's say x is a datapoint which have d components corresponding to it ie. d features so x= (x1,x2,x3, ……., xd) and let's say there are k classes such as c1, c2, c3 …...ck. p(ck | x) = probability of a class label given a x point. p(ck |x)= p(ck)* p(x|ck) /p(x). We are calculating p(ck |x) for all k classes for a given point with respect to d features. Whichever class has highest value of p(ck |x) for a given point x, we declare that class as predicted class for that point x.
3. **Geometric intuition**: For Naive Bayes , there is no decision surface which will decide whether that point is from positive class or negative class. In Naive Bayes, we are not finding hyperplanes to decide the class labels of the data points.
4. **Assumptions**: In Naive Bayes, basic assumption is features are conditionally independent of each other given the class label. This is a basic assumption to make probability calculation simpler.
5. **Loss function**: There is no loss function in Naive Bayes that we are minimizing or maximizing it. Naive Bayes is a purely probability based algorithm.
6. **Time complexity**: Training time complexity is o(n) but during test time, time complexity is o(dc) which is less . So Naive Bayes gives faster results than knn.
7. **Overfitting and Underfitting**: Alpha (Laplace Smoothing Coefficient) is hyperparameter in case of Naive Bayes. When alpha=0 then there is overfitting for naive bayes and for alpha= high value then there is underfitting.
8. **Interpretability**: Naive Bayes is highly interpretable and we can easily get feature importance. We can use likelihood probabilities of features with respect to class for giving feature importance. If x is datapoint as x= (xq, x2, x3…...xd) if p(x2 | y=1) corresponding to the 2nd feature is large so that datapoint is corresponding to class 1.
9. **Outliers**: Effect of outliers is reduced by using Laplace Smoothing. In case of text classification, we can use Naive Bayes. If there are words which are occurring less frequently then we can set a threshold to remove these less frequent words. ANother approach is we apply Laplace Smoothing for frequently appearing words ie, outliers (Rare words).
10. **Imbalanced dataset** : We can apply upsampling or downsampling to make imbalanced dataset as balanced dataset. This is problem specific. Depending upon the problem we do upsampling otherwise we keep it as it is ie.imbalanced dataset. Laplace smoothing also helps in tackling imbalanced dataset, impact of alpha will be more for minority class.
11. **Application** : Naive Bayes is mostly used for text classification. It gives better results for text classification. Eg. spam filter in case emails.

12. **Advantage** : It is a simple algorithm . It gives results very fast as we use precomputed likelihood probabilities during the testing phase. It works better when there are categorical features. This algorithm can be extended to multiclass classification.
13. **Limitation** : If the problem needs to keep sequence of words in a text in this case naive bayes may not work better. We don't use naive bayes mostly when there are numerical features . If similarity or distance matrix is given Naive bayes cannot be applied.
14. **Comparison to other algorithms** : Naive bayes gives results within less time when it is compared to Knn. Naive bayes can be the first baseline model for text classification.

Logistic Regression

- **Definition:**

In statistics, the logistic model (or logit model) is used to model the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick.

- **Math:**

It tries to find a hyperplane with weights associated with it as W^TX so that 2 classes can be separated at best. It uses something called a logistic function also known as sigmoid function to find the best hyper plane.

Logistic Function:

For detailed mathematical intuition you can go here: https://medium.com/swlh/logistic-regression-a-beginners-guide-c3d3f9ca5993

- **Geometric intuition:**

Logistic Regression is all about finding the decision boundary which can well separate the both classes.

So we have to find the Hyper Plane as W^TX using the distance of points from the plane.

- **Assumptions:**

· Binary logistic regression requires the dependent variable to be binary.

· For a binary regression, the factor level 1 of the dependent variable should represent the desired outcome.

· Only meaningful variables should be included.

· The independent variables should be independent of each other. That is, the model should have little or no multicollinearity.

· The independent variables are linearly related to the log odds.

· Logistic regression requires quite large sample sizes.

· Data Points should be linearly separable

- **Loss Functions:**

-[y*log (y) + (1-y)*log (1-y)]

- **Time complexities:**

Training Time Complexity means in logistic regression, it means solving the optimization problem.

Train Time Complexity = O (nd)

Test Time Complexity = O (d)

- **Interpretability:**

Yes it is highly interpretable as it gives the top features based on which a class label was assigned. How to do that? Imagine we have 100 features or 100 dim weights we can easily pick the top 5 values and tell based on these values a class label was assigned.

- **Outliers:**

Less impact due to sigmoid function

We can remove outliers which are very far away from our decision boundary just by keeping a threshold value.

- **Imbalanced Dataset:**

Logistic regression does not support imbalanced classification directly.

Either Upsampling or downsampling is required.

- **Code:**

Import sklearn.linear_model.LogisticRegression

sklearn_model = LogisticRegression()

- **Applications:**

Any sort of Binary classification problem, like Fraud Detection, Spam Filter etc.

- **Advantages:**

Logistic Regression performs well when the **dataset is linearly separable**.

Logistic regression is less prone to overfitting but it can overfit in high dimensional datasets. You should consider Regularization (L1 and L2) techniques to avoid over-fitting in these scenarios.

Logistic Regression not only gives a measure of how relevant a predictor (coefficient size) is, but also its direction of association (positive or negative).

Logistic regression is easier to implement, interpret and very efficient to train.

- **Limitations & Workaround:**

Logistic Regression is also not one of the most powerful algorithms out there and can be easily outperformed by more complex ones.

Also, we can't solve non-linear problems with logistic regression since its decision surface is linear.

Logistic regression will not perform well with independent variables that are not correlated to the target variable and are very similar or correlated to each other.

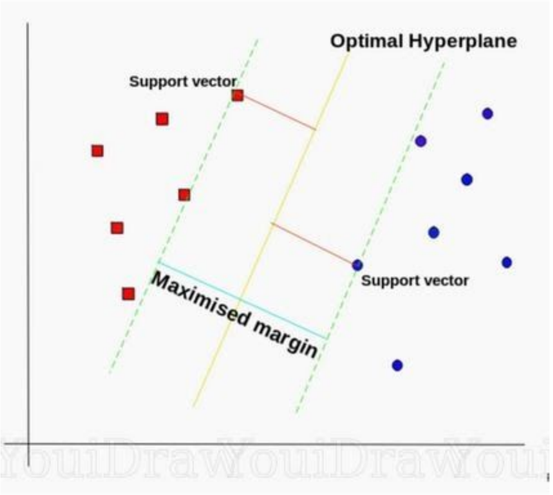
We can use DecisionTree, Random Forest, XGBoost etc. to overcome the limitations.

- **Comparison with other models:**

Logistic regression is suitable to solve any linear problem, but when it comes to non-linear we have to use complex algorithms to solve. If we need low latency applications we use logistic regression, if the number of dimensions are considerable.

SVM : Support Vector Machines

- · **Definition:** "Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

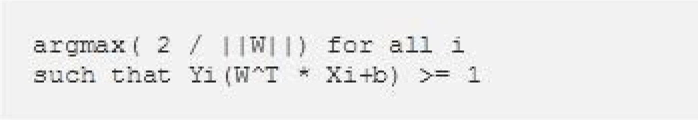


Optimal Hyperplane using the SVM algorithm

- **Math behind SVM:** SVM problem can be formulated as,

$$
\begin{array}{cc}
w.x_i+b \geq 1 & \text{for } y_i=+1 \\
w.x_i+b \leq -1 & \text{for } y_i=-1 \\
\multicolumn{2}{c}{\text{combining above two equation, it can be written as}} \\
y_i(w.x_i+b)-1 \geq 0 & \text{for } y_i=+1,-1
\end{array}
$$

So our optimisation function in case of SVM is —



```
argmax( 2 / ||W||) for all i
such that Yi(W^T * Xi+b) >= 1
```

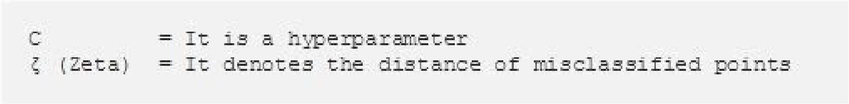
Here Yi(W^T * Xi+b) >= 1 implies that the points are perfectly linearly separable. So all the positive points lie on the positive side of the plane and all the negative points lie on the negative side.As, optimal hyperplane maximize the margin, then the SVM objective is boiled down to fact of maximizing the term *1/|w|*,

The approach that we followed is called the **Hard-Margin SVM** and it is rarely used in real life. So in order to use the SVM in real world applications, a modified version was created, called the **Soft- Margin SVM**.

$$\text{Argmin} \;\; (||W|| / 2) + C * 1/n * \sum_{i=1}^{n} \zeta_i$$

Earlier we were looking to maximize the term 2/||W|| but now since we inverted it, therefore we have changed argmax to argmin.

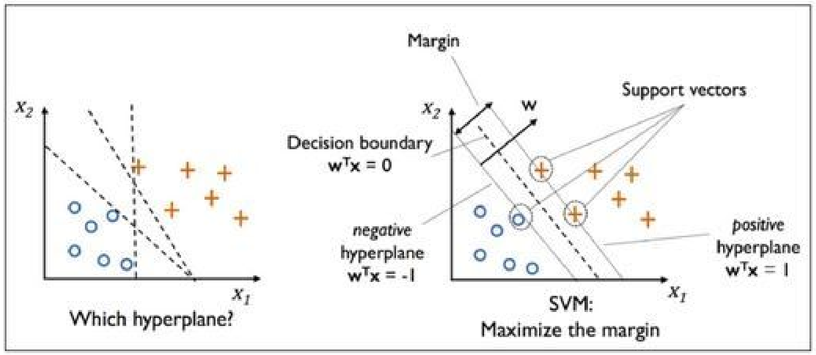
You might have already guessed that ‘**n’** denotes the number of data points. So the 2 newly added terms in this equation are —



```
C          = It is a hyperparameter
ξ (Zeta)   = It denotes the distance of misclassified points
```

C is a hyperparameter and it can be tuned effectively to avoid overfitting and underfitting.

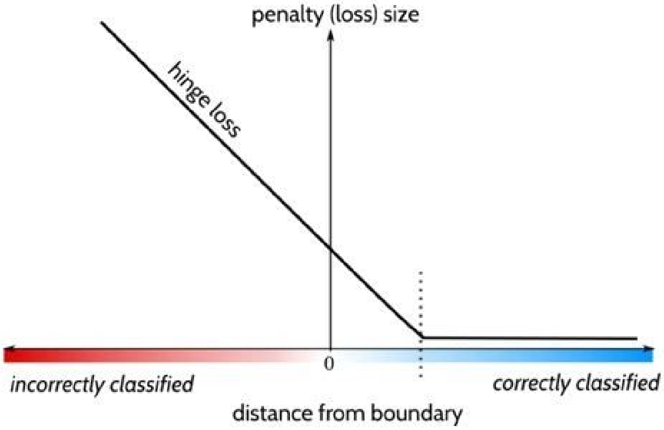
- **Geometric Intuition:** The main idea behind SVM is to find a plane that best separates the positive and negative points and the distance between the positive plane and the negative plane is maximum. The rationale behind choosing the decision boundaries with larger margins is that it reduces the generalisation error and the decision boundaries with smaller margins usually lead to overfitting.



- **Loss Function:** The loss function used in SVM is hinge loss.In simple terms we can understand hinge loss as a function whose value is non-zero till a certain point let’s say ‘z’ and after that point ‘z’ it is equal to zero.

  If Zi is greater than or equal to 1 then loss = 0. (correctly classified)

  If Zi is less than 1 then loss = 1 — Zi. (incorrectly classified)



Hinge Loss

- **Time Complexities:** Let’s say we have dataset which has n data points and m features then,

$$T(n) = O(n^2)$$

at train time

T(n) = O(m.k) at test time, Where k = support vectors

S(n) = O(k) Because we need to store k vectors.

- **Overfitting and Underfitting:** The hyperparameters of SVC are γ(RBF Kernel) and C(soft margin). If C is very large then it is a hard margin SVM. So it is a high variance (overfit). If C is very small then it is a high bias (underfit). If γ is small then the model underfits. If γ is large then the model overfits.
- **Interpretability :** This is the biggest challenge with SVMs. There is no way to get feature importance here. So, we have to do forward feature selection natively which is time consuming.
- **Outliers:** SVM has very little impact on outliers as classification/regression entirely depends on choosing support vectors points only. In case of RBF with small σ, it behaves very similar to KNN with a small value of k.
- **Imbalanced dataset:** SVMs work fine on sparse and unbalanced data. Class-weighted SVM is designed to deal with unbalanced data by assigning higher misclassification penalties to training instances of the minority class. Instead you should calculate the precision, recall and F-Score of the algorithm.
- **Important Hyperparameters:** The hyperparameters of SVC are γ(RBF Kernel) and C(soft margin).
- **Applications:** text classification, genome data classification, face detection, image classification, handwritten character recognition, bioinformatics
- **Advantages:** It works really well with a clear margin of separation
- It is effective in high dimensional spaces.

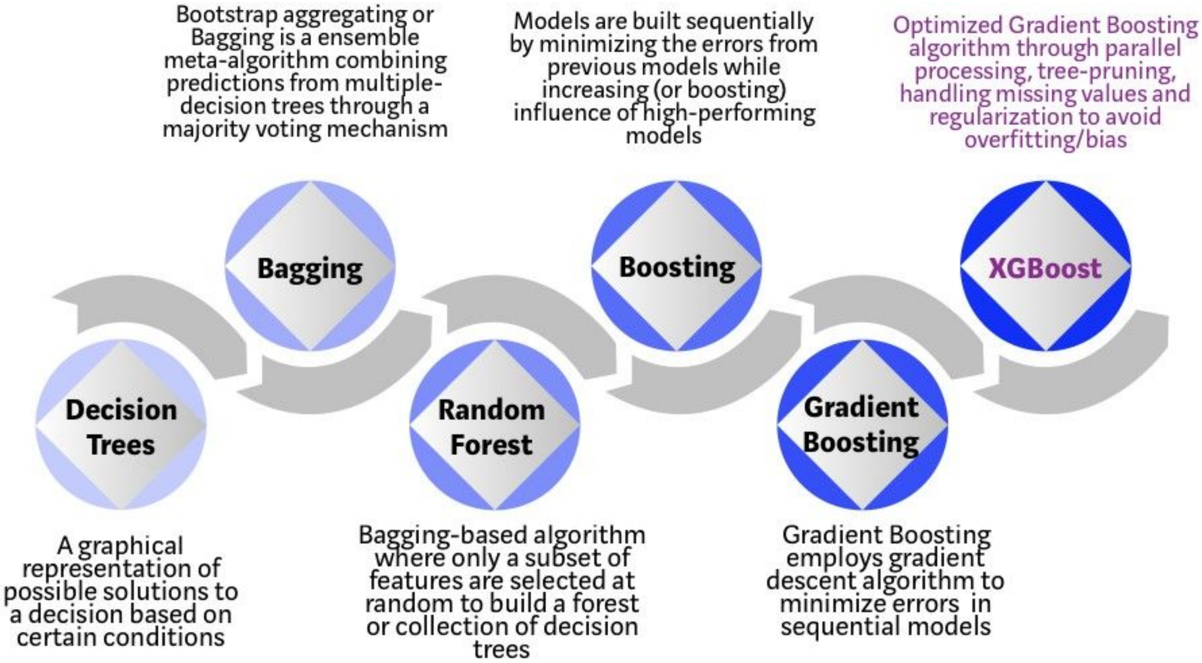
- It is effective in cases where the number of dimensions is greater than the number of samples.
- It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- **Limitations and workarounds:** It doesn't perform well when we have large data set because the required training time is higher
- It also doesn't perform very well, when the data set has more noise i.e. target classes are overlapping
- SVM doesn't directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is included in the related SVC method of Python scikit-learn library.
- Not suitable for low latency internet applications
- **Variations:** Linear SVM, kernel SVM, nu-SVM
- **Comparison with other ML models:** SVM can handle non-linear solutions whereas logistic regression can only handle linear solutions. SVM handles outliers better than LR and KNN. Hinge loss in SVM outperforms log loss in LR. If training data is much larger than no. of features(m>>n), KNN is better than SVM. SVM outperforms KNN when there are large features and lesser training data. Decision trees are better for categorical data and it deals collinearity better than SVM. It is not interpretable like NB and DT. In simplest manner, SVM without kernel is a single neural network neuron but with different cost function.

XGBoost: Extreme Gradient Boosted Decision Trees

- **Definition:** It is an optimized gradient boosted library. It is highly efficient, portable and can solve problems with billions of data points. Boosting is an approach where new models are created that predict the residuals or errors of prior models and then ensemble to make the final prediction. Gradient descent is used to optimize the value of the loss function. XGBoost uses randomization (column sampling + row sampling) in addition to Gradient boosting. Decision Trees internally take care of missing values.
- **Math:** the first model will have prediction, f_1 = argmin (Loss (y_true, y_pred)), second model, f_2 = f1 + res(f1), …, f_m = f_(m-1) + res (f_(m-1)), where f_m is the mth model. The residual can be multiplied with alpha, f_m = f_(m-1) + alpha*res (f_(m-1)). Gradient descent is used to determine alpha that minimizes the loss(y_true, y_pred). Squared error is the default loss function. We can add regularization to reduce overfitting. Using loss function, Information gain is computed for the features to decide on the split.
  - Gain = loss before split - loss after split
  - Reference: [Math behind xgboost]
- **Geometric intuition:** Trees are a data structure used for decision making. It consists of root node, decision node and terminal node. The data points in the space are divided

with axis parallel Hyperplanes that separate them into different classes or spaces where predictions are made based on data points in the divided spaces.

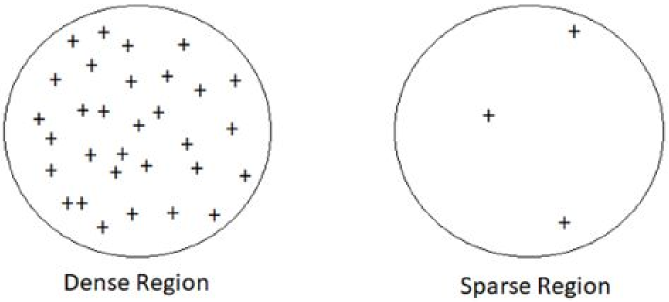
- **Loss Function:** Squared loss, squared log loss, hinge loss (can use softmax objective for multi class).
- **Time complexities:** Train time: O(depth * log(n) * m), Run time: O(depth *m), Run space: O(store each tree)
- **Overfitting and Underfitting:** Large Depth of each tree and large number of trees result in overfitting
- **Interpretability:** Trees are highly interpretable, we can judge a decision by looking at the path the model has taken, we can get features that got involved in the prediction
- **Outliers:** Tree based models are robust to outliers as it is a nested if-else algorithm
- **Imbalanced Dataset:** XGBoost can efficiently work on imbalanced datasets, we can also use weighted XGBoost
- **Important Hyperparameters: XGBClassifier and XGBRegressor are available:** learning_rate, max_depth, n_estimators
- **Applications:** Widely used for classification, regression, ranking systems
- **Advantages:** Faster training time, No preprocessing is required, Handles collinearity, highly interpretable
- **Limitations and workarounds:** Overfits easily - hyperparameter tuning


Bootstrap aggregating or Bagging is a ensemble meta-algorithm combining predictions from multiple-decision trees through a majority voting mechanism
Models are built sequentially by minimizing the errors from previous models while increasing (or boosting) influence of high-performing models
Optimized Gradient Boosting algorithm through parallel processing, tree-pruning, handling missing values and regularization to avoid overfitting/bias
Bagging
Boosting
XGBoost
Decision Trees
Random Forest
Gradient Boosting
A graphical representation of possible solutions to a decision based on certain conditions
Bagging-based algorithm where only a subset of features are selected at random to build a forest or collection of decision trees
Gradient Boosting employs gradient descent algorithm to minimize errors in sequential models

- **Comparison with other models: vs Naive Bayes:** XGBoost is a discriminative model while NB is a generative model, XGB is more flexible to ML problems, **vs KNN:** Both are non parametric, XGBoost readily supports feature interaction, XGBoost works faster at run time, **vs Support Vector Machines:** SVM uses Kernel tricks for non-linear data, XGBDT uses hyper-rectangles and deals collinearity better than SVM, **vs RandomForest** XGB is fast, RF is less prone to outliers, **vs Neural Networks:** XGBDT perform well for categorical values, provides better interpretability, NN outperforms XGBDT when there is sufficient training data

DBSCAN: Density-based spatial clustering of applications with noise

- **Definition:** DBSCAN is an unsupervised learning data clustering algorithm that is commonly used in data mining and ML to find associations and structures in data that are hard to find manually.
- **Geometric Intuition:** Based on a set of points, DBSCAN groups points together that are close to each other based on a distance measurement (usually **Euclidean distance**). It also marks the outliers if the points are in low-density regions or Sparse Region.



Parameters: DBSCAN algorithm basically requires **2** parameters:

1. ***eps:*** Specifies the radius of that region. It means that two points are considered neighbors if the distance between the two points is below the threshold epsilon(eps).
2. ***minPoints:*** the minimum number of points to form a region. It means if we set the minPoints parameter as 5, then we need at least 5 points to form a region.

The algorithm works by computing the distance between every point and all other points. We then place the **points into one of three categories**.

1. **Core point:**

· A point is a core point if it has more than a specified number of m*inPoints* within *eps* radius around it. Core Point always belongs in a dense region.

· For example, let's consider '*p'* is set to be a core point if '*p'* has ≥ *minPoints* in an *eps* radius around it.

2. **Border point:**

· A point is a border point if it has fewer than *minPoints* within a *eps*, but is in the neighborhood of a *core point*.

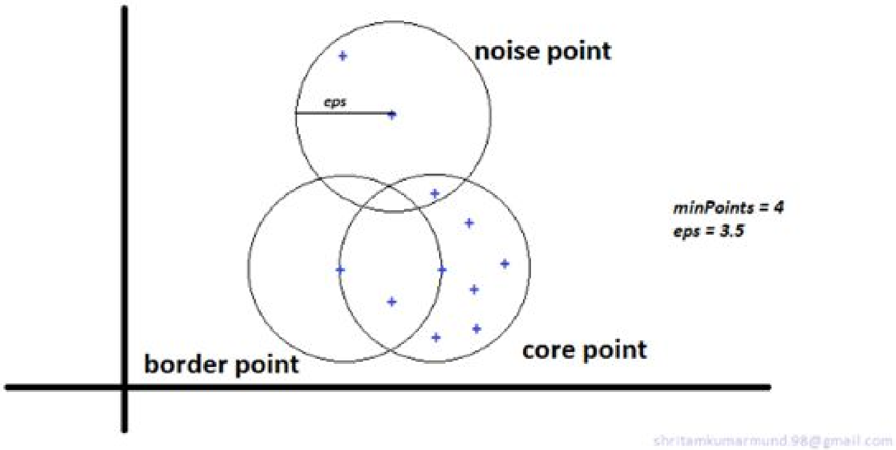
· For example, p is set to be a border point if *'p'* is not a core point. i.e *'p'* has < *minPoints* in *eps* radius. But *'p'* should belong to the neighborhood *'q'*. Where *'q'* is a core point.

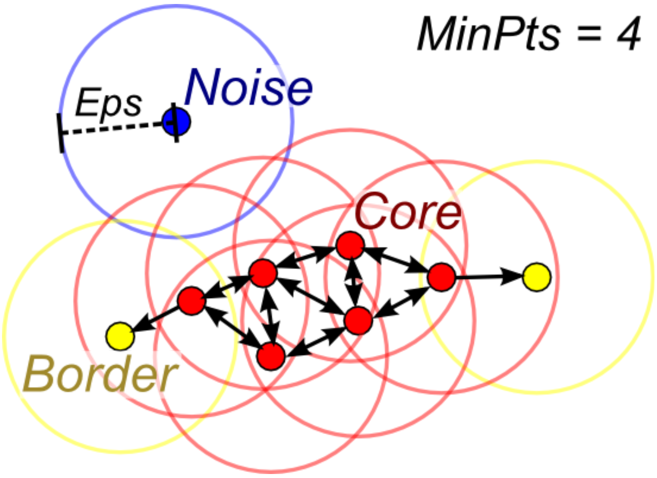
· p ∈ neighborhood of *q* and distance(*p,q*) ≤ *eps* .

3. Noise point:

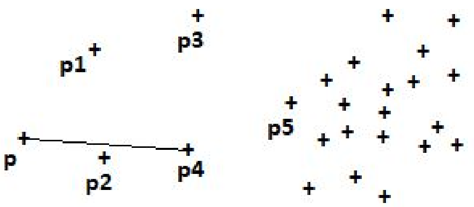
· A noise point is any point that is not a core point or a border point.

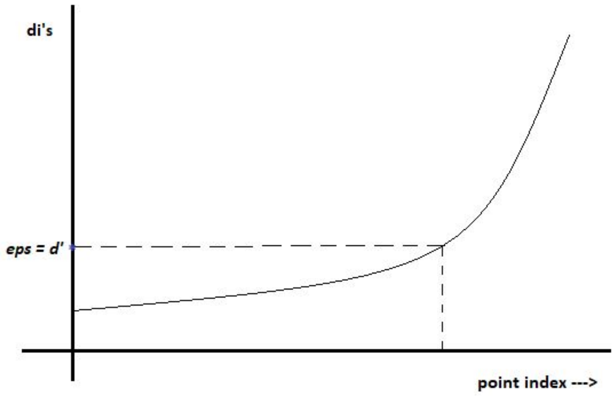
· These are the Outliers point. Which is very nicely handled by DBCAN.





- **How to choose Min Points?**
  - ○ Typically, the rule of thumb for taking min Points is, We should always take the min points to be greater or equal to the dimensionality(d) of your dataset.
  - ○ Typically, people who work most with DBSCAN take min point twice of the dimensionality of data i.e minPoint≈2*d.
  - ○ If the dataset is noisy, we should choose a larger value of minPoint.

  - While choosing the min points, it really depends a lot on domain knowledge.
- **How to determine eps?**
  - Once we choose our min Point, we can proceed with the eps.
  - Let us choose a min point = 4, for each point *p* I'll compute "*di*". where *di*= distance from *p* to the 4th nearest neighbor of *p*. If "*di*" is high, then the chance of *p* is noisy is also high.
  - 

    4th nearest neighbor of p is p4
  - For each point in the data set, I'll have my di's. Now sort all di's in increasing order.
  - Since we have sorted our point in the increasing order of di's, now we'll plot sorted point index and di's with elbow or Knee method and get our best eps.
  - 

- **When does DBSCAN work well?**
  - It can handle Noise very well.
  - DBSCAN can handle clusters of different shapes and sizes.
- **When not!**
  - If your dataset has multiple densities or varying densities, DBSCAN tends to fail.
  - It's extremely sensitive to the hyperparameters. A slight change in hyperparameters can lead to a drastic change in the outcome.
- **Time Complexity: O(n log n)**
- **Space Complexity: O(n)**
- **Code:**

```
# Importing the libraries
import numpy as np
import pandas as pd
```

```
# Importing the dataset
df = pd.read_csv(dataset.csv')

# Using the elbow method to find the optimal number of clusters
from sklearn.cluster import DBSCAN
dbscan=DBSCAN(eps=3,min_samples=4)

# Fitting the model
model=dbscan.fit(df)

#Let’s store cluster labels for each point.
#Noisy samples are given the label -1.

labels=model.labels
```

Contributors

S. No	Content Contributors	LinkedIn/MailId/Github
1	Karthik Kumar Billa (Owner)	https://www.linkedin.com/in/karthik-kumar-billa/
2	Payal	
3	Bhargav	https://github.com/brpy
4	Amod Kolwalkar	
5	Mukund Madhav	
6	Neelasha Roy	
7	Srilaxmi Nemani	
8	J Shankar C	
9	Oindrilla Ghosh	
10	Renuka	
11	Jyoti Jain	
12	Shritam Kumar	
13	Bishal Bose	
14	Jigar Parmar	
15		
16		
17		
18		
19		
20		
21		

22		
23		
24		

FAQs