continuous uniform distribution describes an experiment where there is an arbitrary outcome that lies between certain bounds. The bounds are defined by the parameters, *a* and *b*, which are the minimum and maximum values.

Continuous r.v. - which are measurable but can not be counted like Height, Weight, Temp etc

PMF is for discrete r.v.

Discrete r.v. - Countable, can be infinite but are countable to some extent

PDF is for continuous r.v.

CDF - Cumulative distribution function (CDF) of a real-valued r.v. X evaluated at x, Prob that X takes value <=x.

In the case of a scalar continuous distribution, it gives the area under the probability density function from minus infinity to x

PDF - PDF is used to specify the probability of the random variable falling *within a particular range of values*, as opposed to taking on any one value

This probability is given by the integral of this variable's PDF over that range—that is, it is given by the area under the density function but above the horizontal axis and between the lowest and greatest values of the range.

In general though, the PMF is used in the context of discrete random variables (random variables that take values on a countable set), while the PDF is used in the context of continuous random variables.

Bernoulli Distribution : Success/Failure - 2 outcomes of experiment

Binomial Distribution is the same as Bernoulli distribution for n=1 (single trial). Bernoulli dist B(n,p) : n is Number of trials.

p = probability of success

Log Normal Distribution

If X is log normal, Y = ln(X) is normally distributed. Examples of log normal distribution - The length of comments posted in Internet discussion forums follows a log-normal distribution.

In economics, there is evidence that the income of 97%–99% of the population is distributed log-normally.[1] (The distribution of higher-income individuals follows a Pareto distribution).

For highly communicable epidemics,number of hospitalized cases is shown to satisfy the log-normal distribution.

Pareto Distribution - Originally applied to describing the distribution of wealth in a society, fitting the trend that a large portion of wealth is held by a small fraction of the population.The Pareto principle or "80-20 rule" stating that 80% of outcomes are due to 20% of causes.

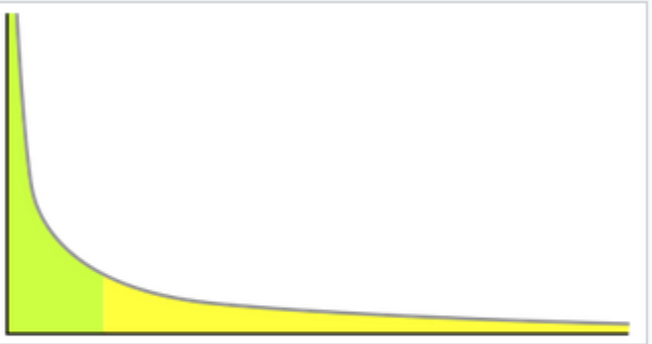
20% of all people receive 80% of all income. i.e. few points with large value, many are having smaller values.

Example : sizes of human settlements (few cities, many hamlets/villages.

The values of oil reserves in oil fields (a few large fields, many small fields.

Power Law - 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.

A few notable examples of power laws are Pareto's law of income distribution.



An example power-law graph that demonstrates ranking of popularity. To the right is the long tail, and to the left are the few that dominate (also known as the 80–20 rule).

log-log plot : is a two-dimensional graph of numerical data that uses logarithmic scales on both the horizontal and vertical axes.

Q–Q (quantile-quantile) plot is a probability plot, which is a graphical method for comparing two probability distributions by plotting their quantiles against each other.

The use of Q–Q plots to compare two samples of data can be viewed as a non-parametric approach to comparing their underlying distributions. If the two distributions being compared are similar, the points in the Q–Q plot will approximately lie on the line *y* = *x*. If the distributions are linearly related, the points in the Q–Q plot will approximately lie on a line, but not necessarily on the line *y* = *x*.

Box cox transform or Power Transform

Convert Non-Gaussian distribution(Pareto distribution, Log normal dist etc) to Gaussian distribution, use box cox transform.

Distribution helps in giving a theoretical model of sample.

Box–Cox transformation [edit]

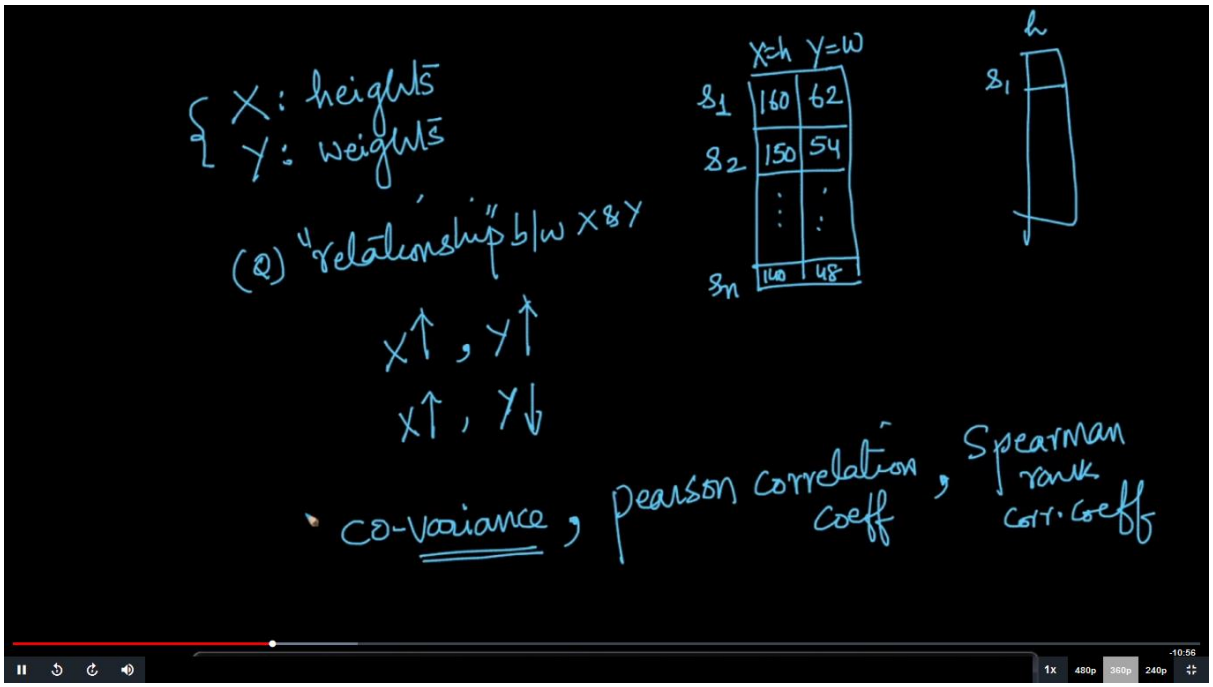
The one-parameter Box–Cox transformations are defined as

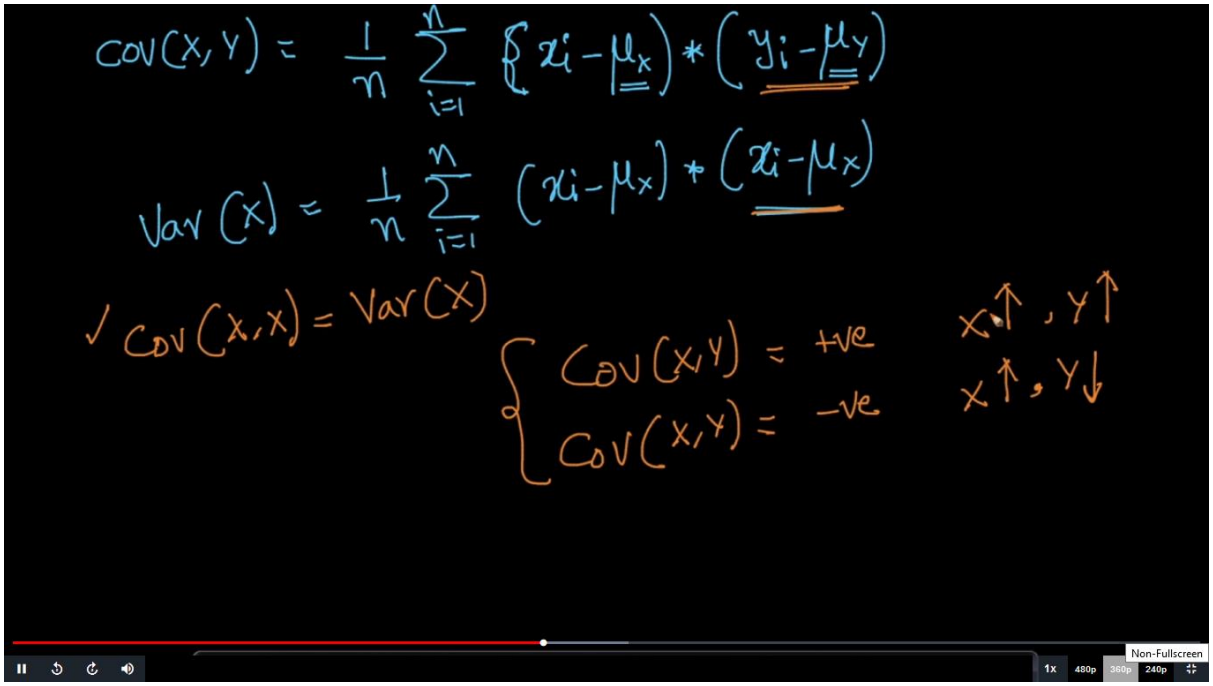
$$y_i^{(\lambda)} = \begin{cases} \dfrac{y_i^{\lambda} - 1}{\lambda} & \text{if } \lambda \neq 0, \\ \ln y_i & \text{if } \lambda = 0, \end{cases}$$

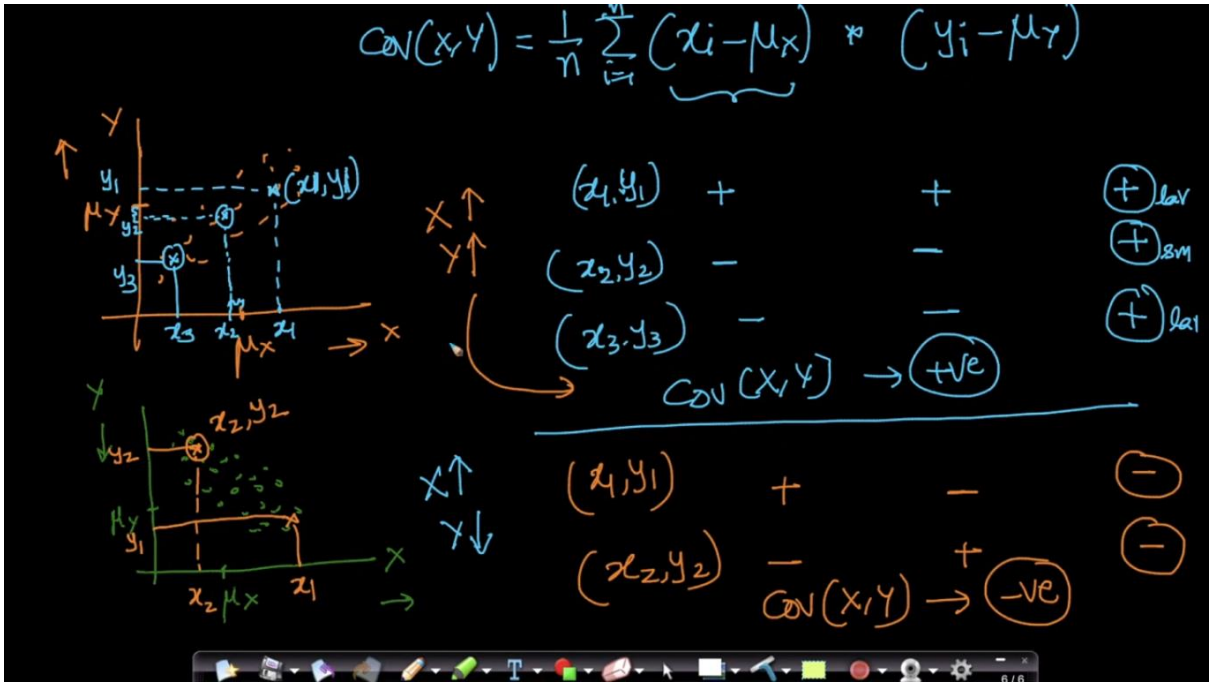
Uniform distribution helps in random number generation.

Covariance between 2 variables

It gives relation b/w 2 random variables.







Since co-variance is scale dependent hence cov(X,Y) of heights and weights in cm and kg would be different from cov(x,y) heights and weights in feet & lbs.
My questions is:
1.Will the sign of co-variance still be same?-i.e.-positve relation will be positive and negative will remain negative even if units are changed?
-if yes then should we bother about some change in numerical values(sign being the same)-as we ultimately only want to discover the trend of linear relationship between X & Y?
-if no then should that mean that the concept of co-variance is not a reliable one to use?

Ans https://soundcloud.com/applied-ai-course/covariance-magnitude-vs-sign

Pearson Correlation Coefficient

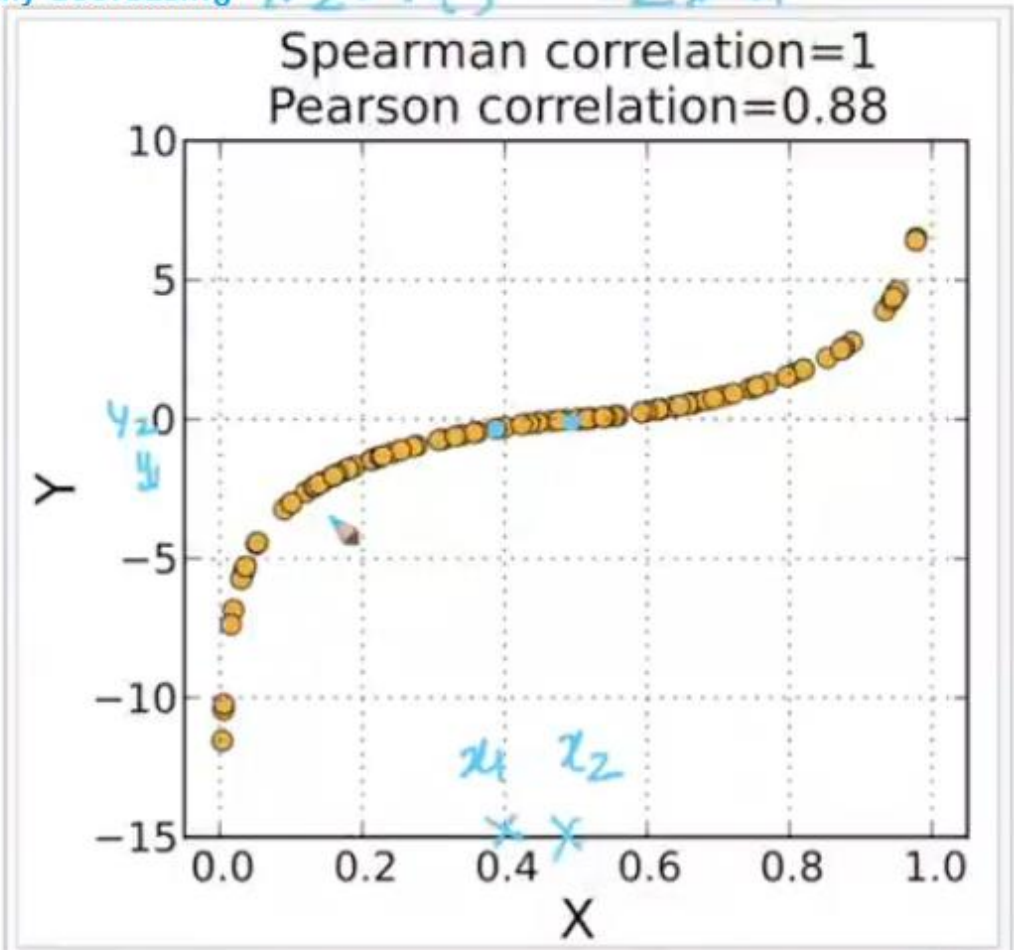
it is essentially a normalized measurement of the covariance, such that the result always has a value between –1 and 1.

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

where:

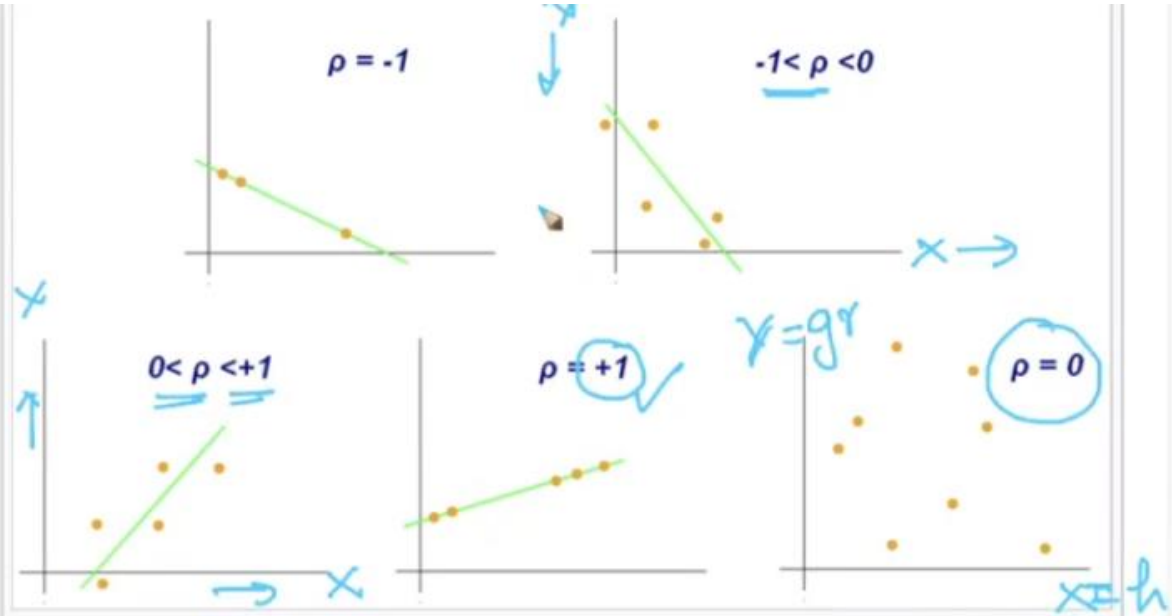
- cov is the covariance
- σ_X is the standard deviation of X
- σ_Y is the standard deviation of Y

monotonically decreasing

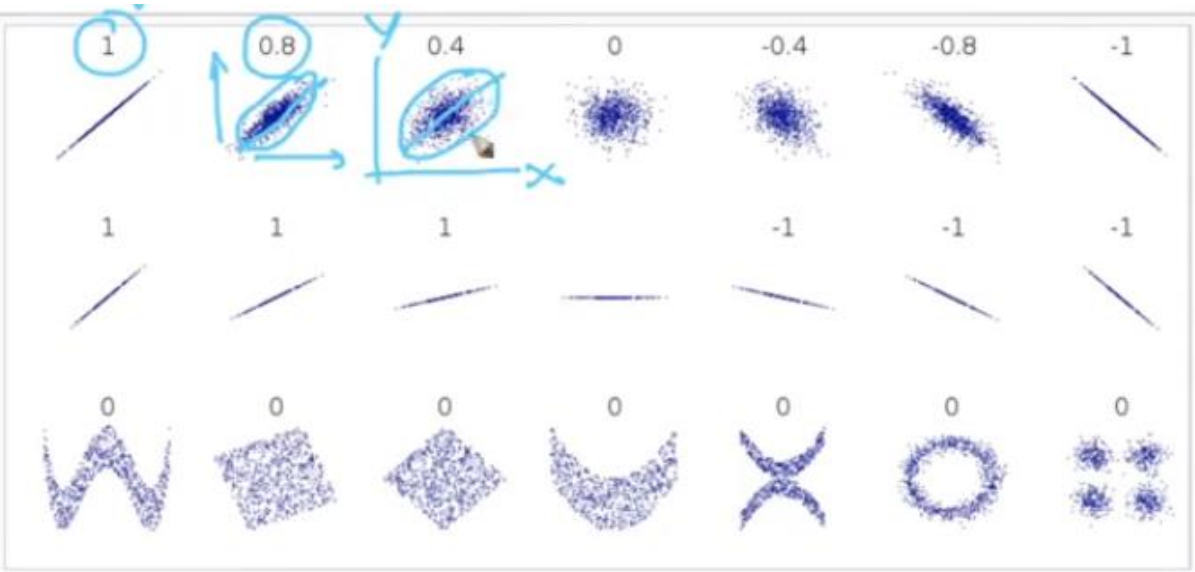


A Spearman correlation of 1 results when the two variables being compared are monotonically related, even if their relationship is not linear. This means that

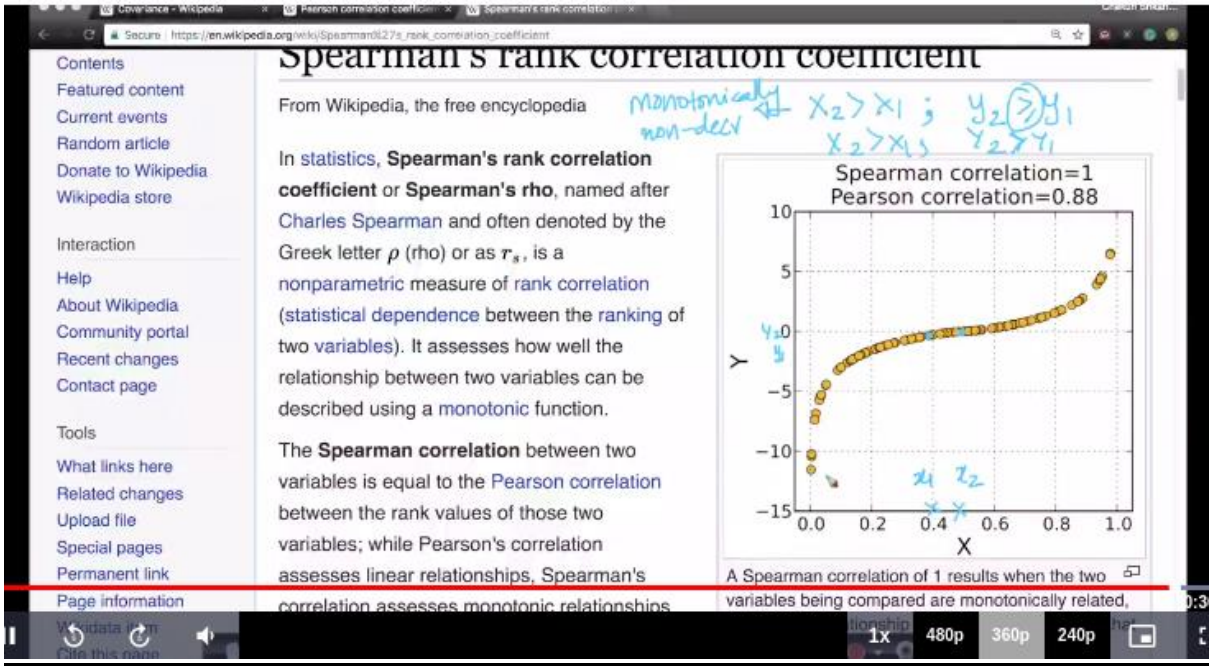
it lies -1<=p<=+1. used for linear relationship



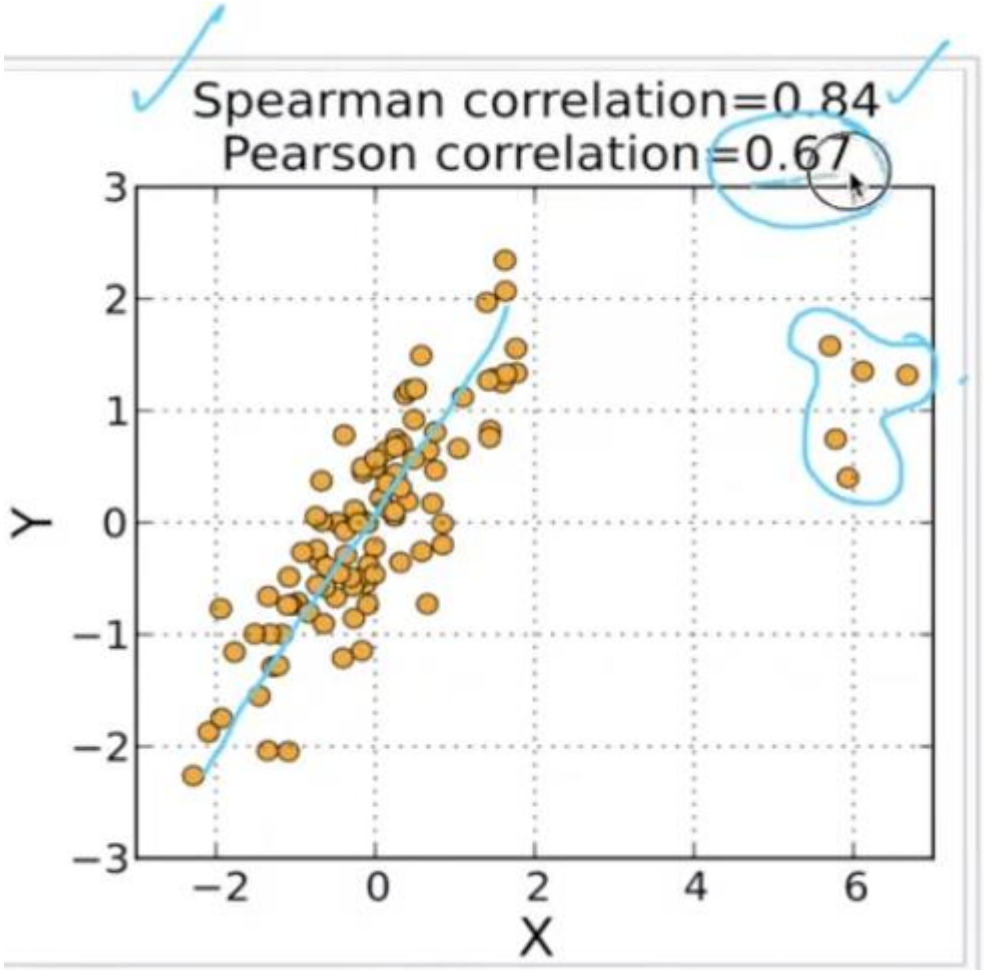
Examples of scatter diagrams with different values of correlation coefficient (ρ)



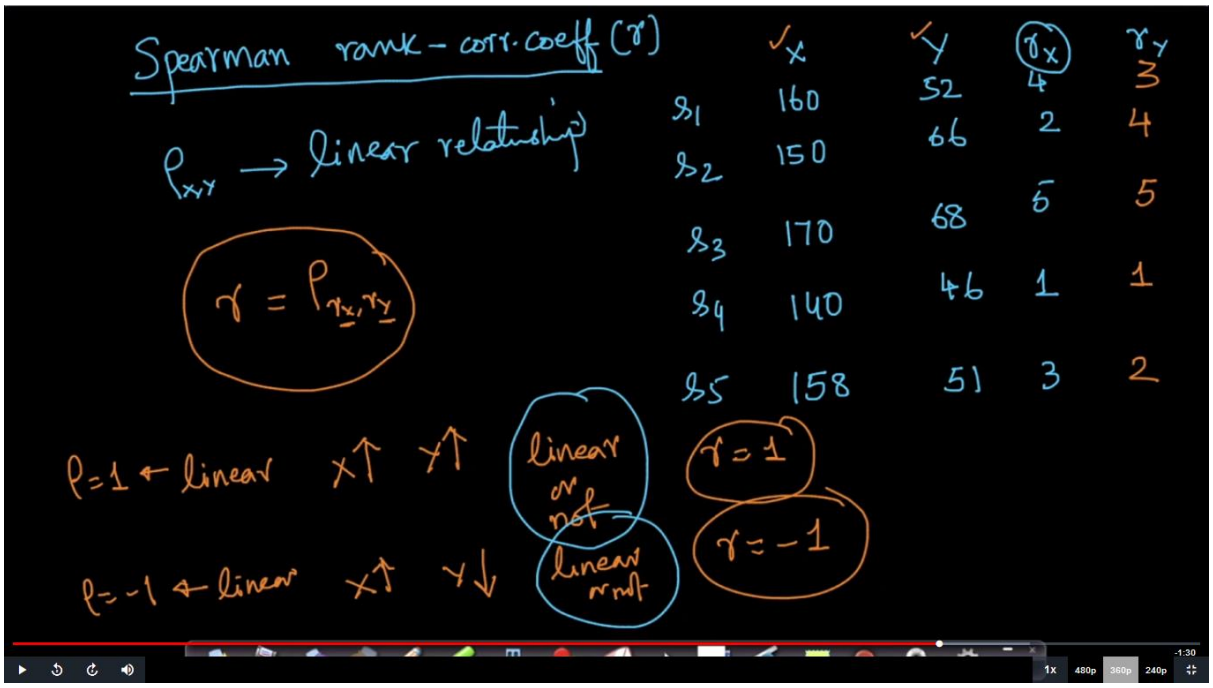
Several sets of (*x*, *y*) points, with the correlation coefficient of *x* and *y* for each set. Note that the correlation reflects the non-linearity and direction of a linear relationship (top row), but not the slope of that relationship (middle), nor many aspects of nonlinear relationships (bottom). N.B.: the figure in the center has a slope of 0 but in that case the correlation coefficient is undefined because the variance of *Y* is zero.

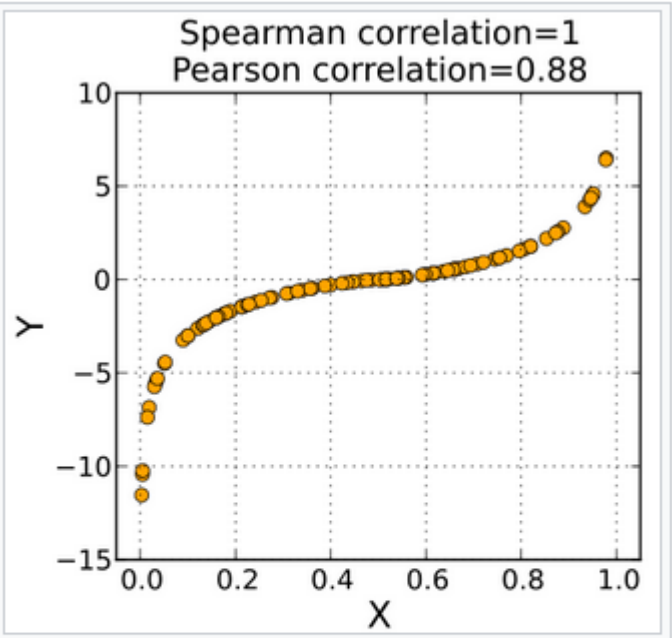


[Spearman Rank Correlation Coefficient]

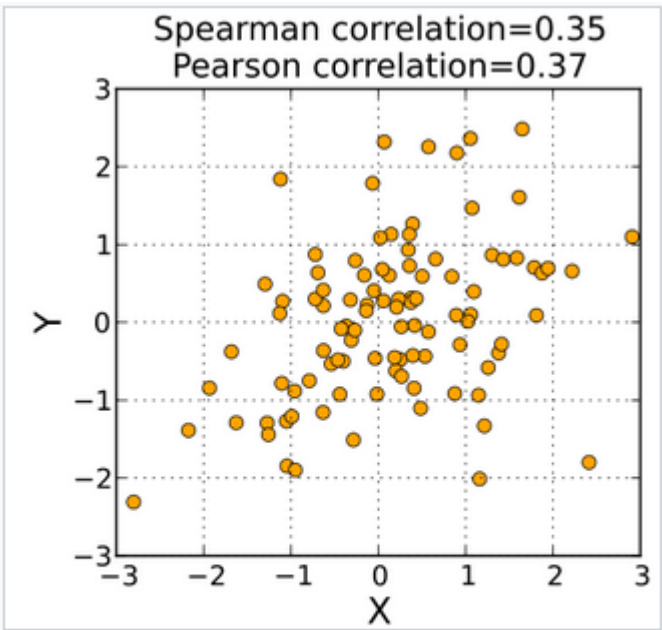


Spearman Rank correlation-coefficient is computed on rank variables rx,ry.

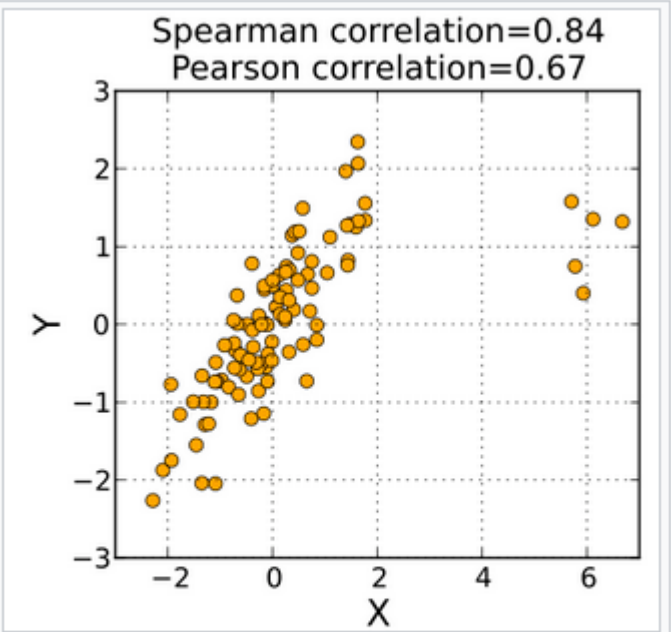




A Spearman correlation of 1 results when the two variables being compared are monotonically related, even if their relationship is not linear. This means that all data points with greater *x* values than that of a given data point will have greater *y* values as well. In contrast, this does not give a perfect Pearson correlation.

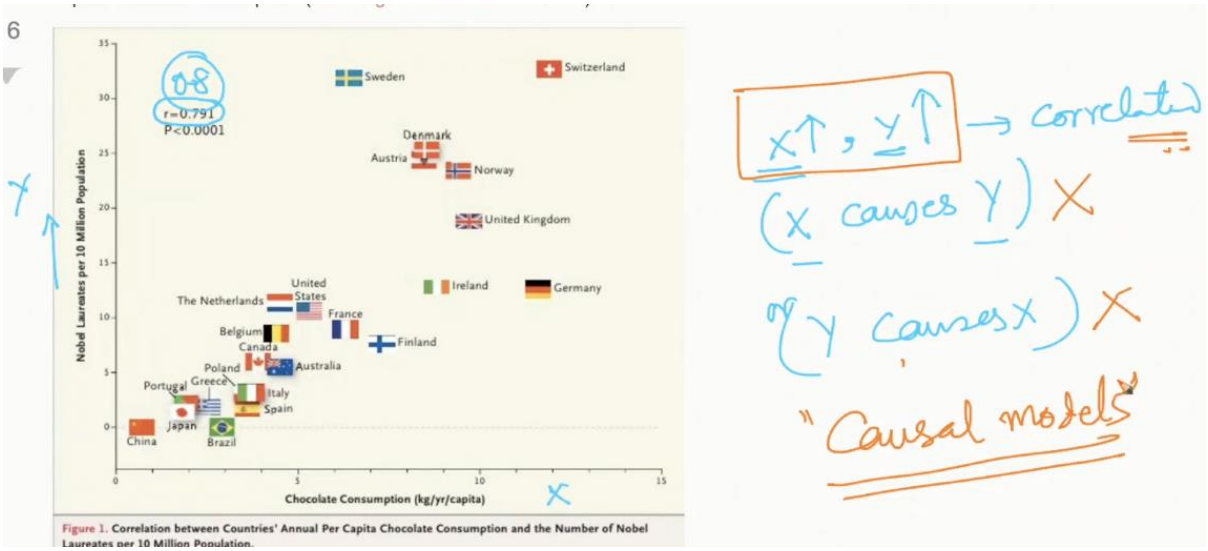


When the data are roughly elliptically distributed and there are no prominent outliers, the Spearman correlation and Pearson correlation give similar values.



The Spearman correlation is less sensitive than the Pearson correlation to strong outliers that are in the tails of both samples. That is because Spearman's ρ limits the outlier to the value of its rank.

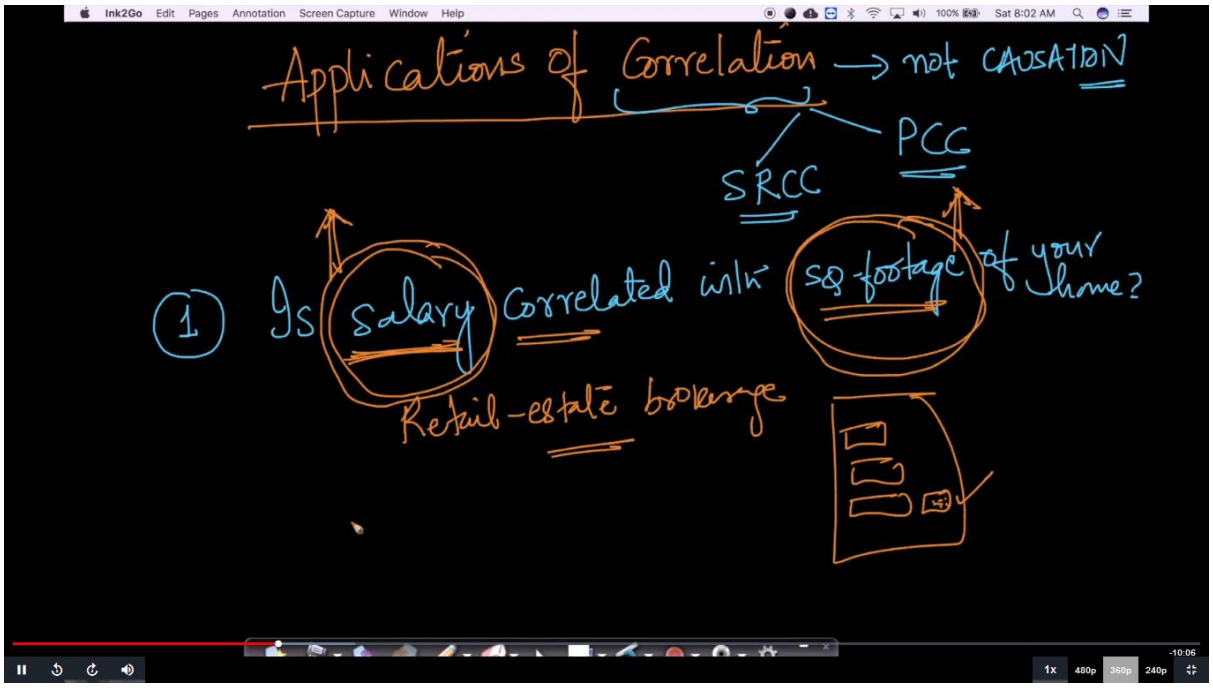
Correlation vs Causation

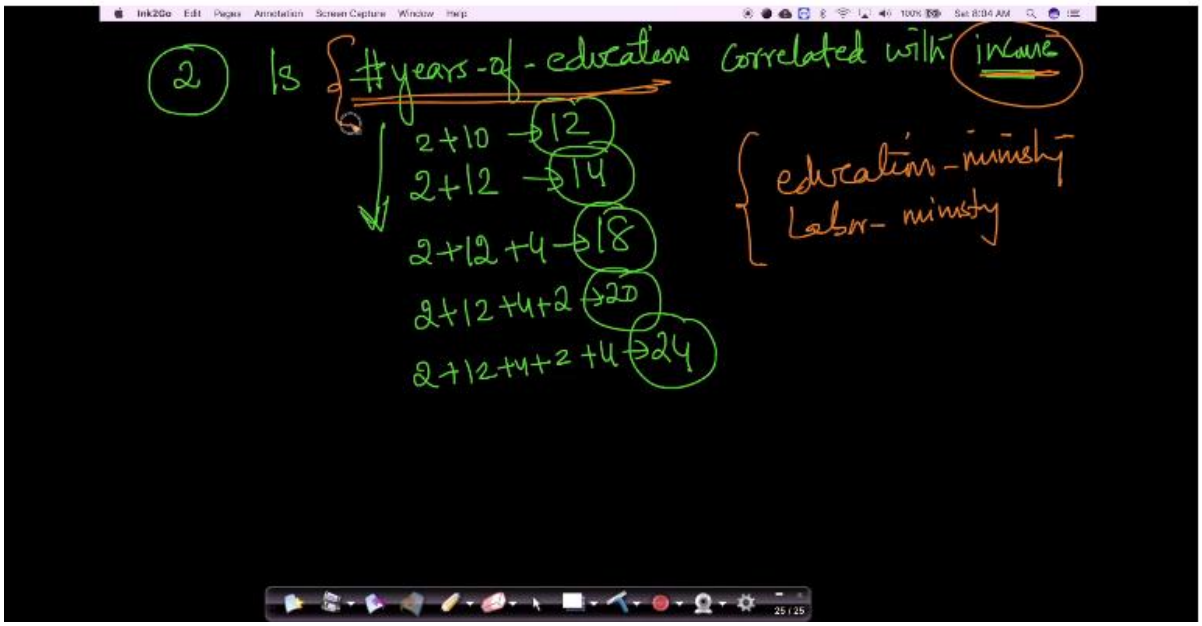


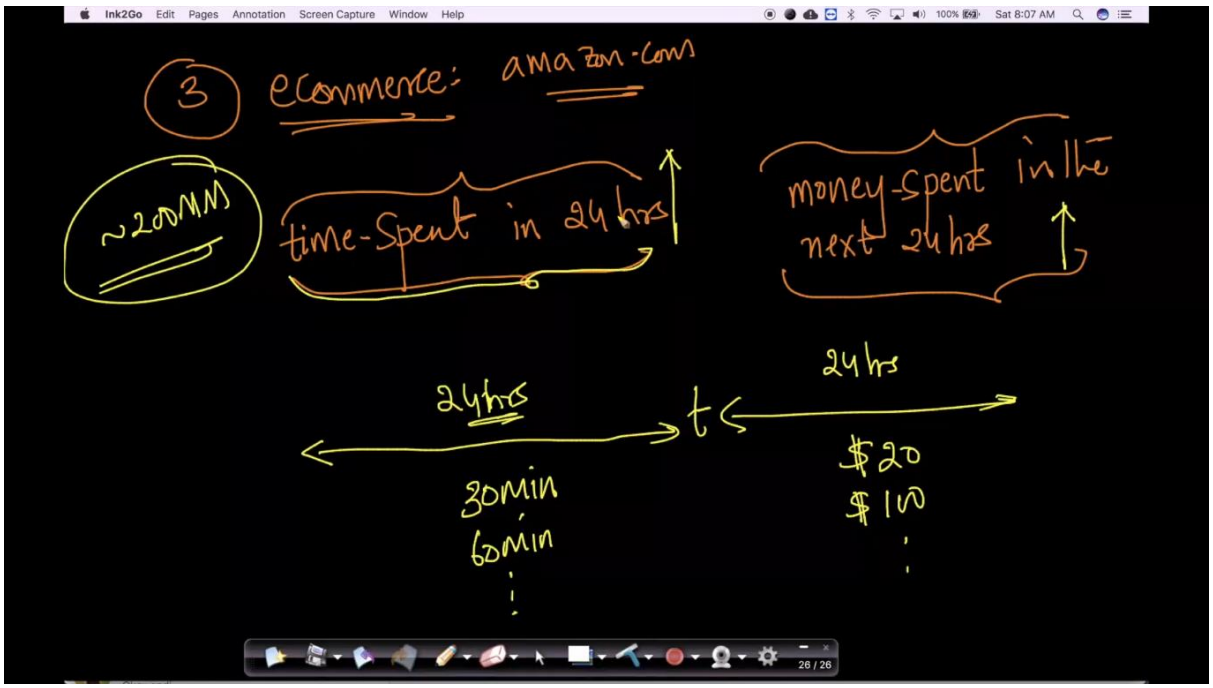
https://statswithcats.net/2015/01/01/how-to-tell-if-correlation-implies-causation/

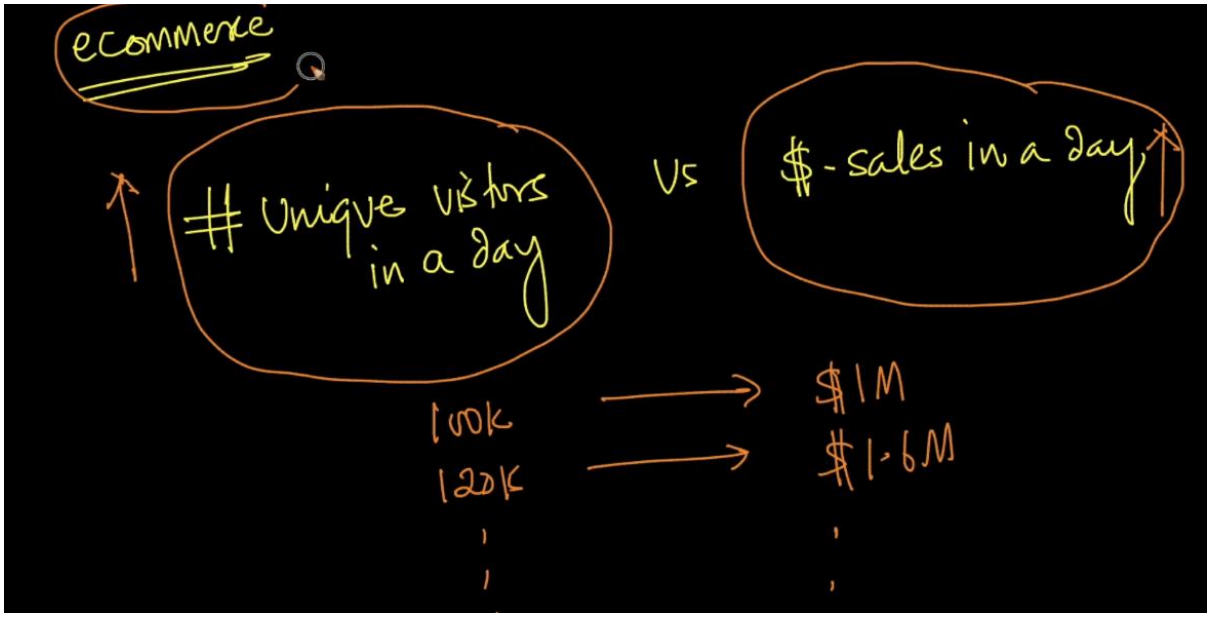
How to use correlations?

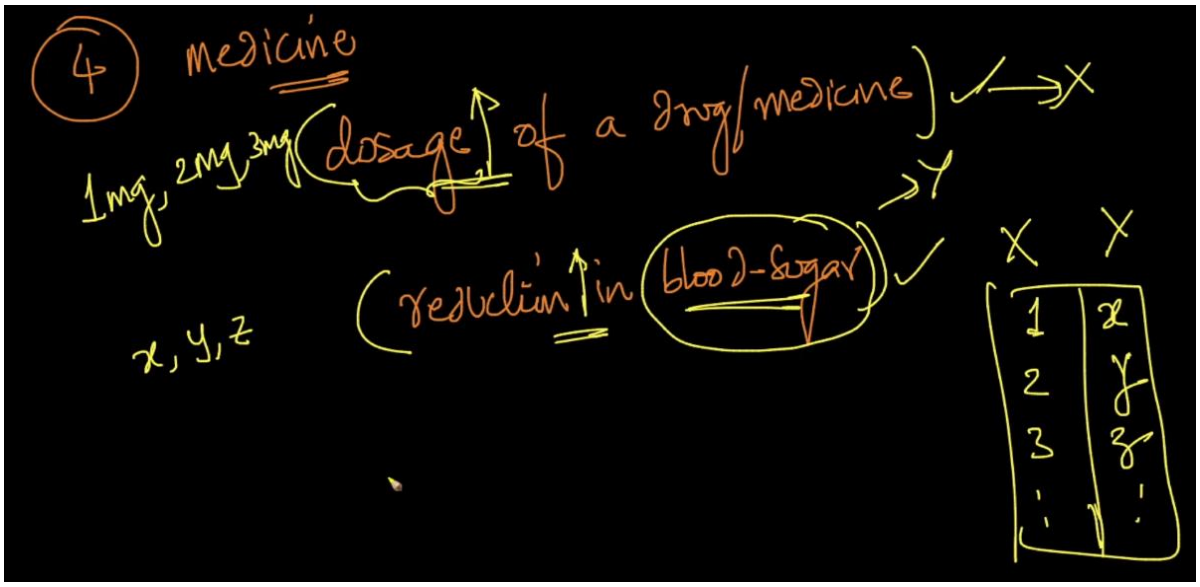
Applications of Correlation



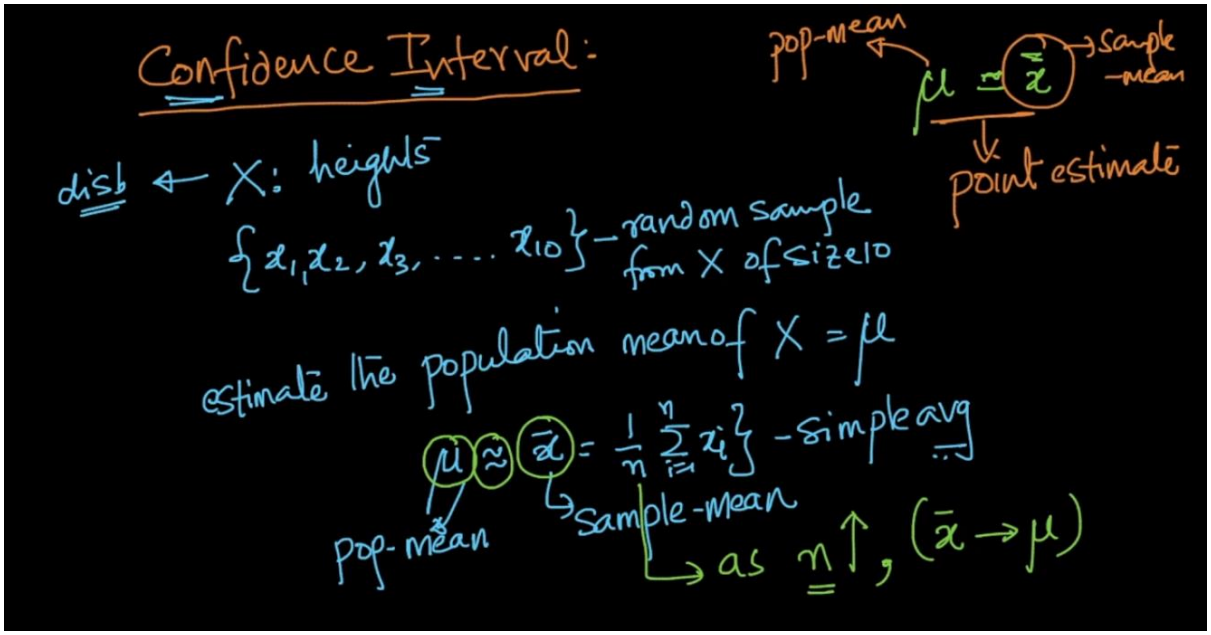

Ink2Go Edit Pages Annotation Screen Capture Window Help
Sat 8:04 AM
2
Is #years-of-education correlated with income
2+10 → 12
2+12 → 14
2+12+4 → 18
2+12+4+2 → 20
2+12+4+2+4 → 24
education-ministry
Labor-ministry
25 / 25

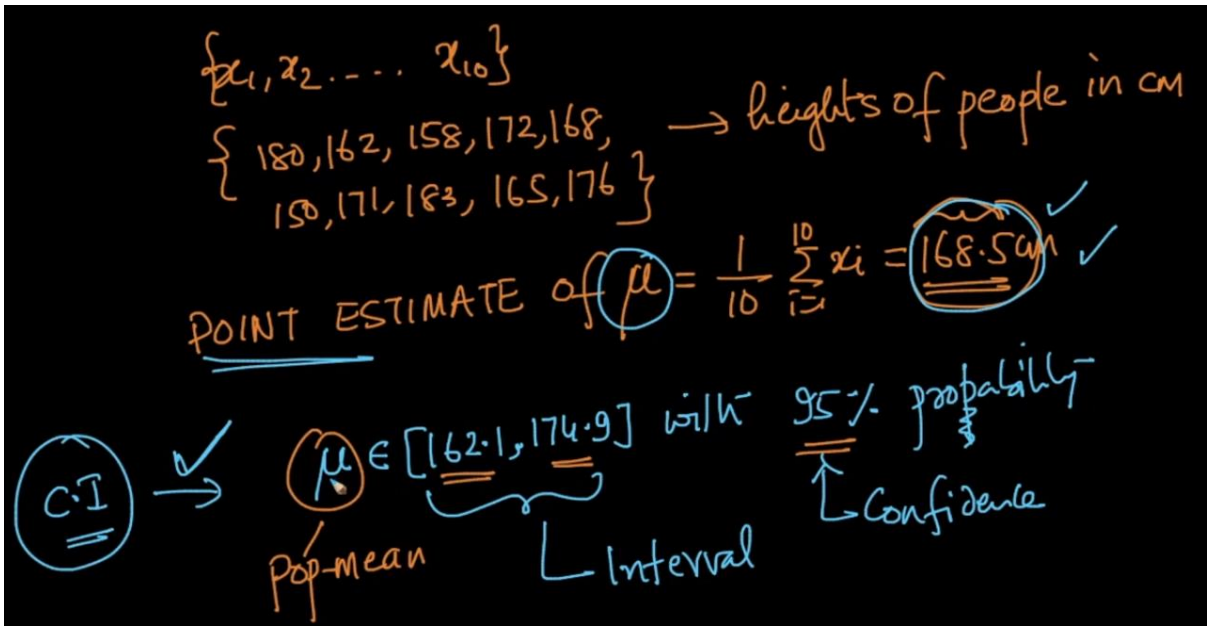

Ink2Go Edit Pages Annotation Screen Capture Window Help
Sat 8:07 AM
3
ecommerce: amazon.com
~200MM
time-Spent in 24 hrs
money-Spent in the next 24 hrs
24 hrs
24 hrs
t
30min
60min
$20
$100
26 / 26





Confidence interval (C.I)





confidence interval says that "how confident we can be that some interval surrounding a sample mean will contain the population mean "

A 95% confidence level does not mean that for a given realized interval there is a 95% probability that the population parameter lies within the interval (i.e., a 95% probability that the interval covers the population parameter).[12] According to the strict frequentist interpretation, once an interval is calculated, this interval either covers the parameter value or it does not; it is no longer a matter of probability.

Central Limit Theorem states that for any distribution of 'X' with a finite mean 'mu' and standard deviation 'sigma', if we select multiple random samples of points from the given population and compute the means for each of these samples, then the distribution of all these sample means will be gaussian with mean = population mean and SD = population SD/sqrt(n) where 'n' denotes the sample size.
Note: All these samples selected should be of same size.

If the entire population data is given for us, then we can directly go ahead with computing the population mean and population standard deviation. When it comes to inferential statistics, we will not have the entire population data. We need to make inferences about the population from the given samples of the data. This is where the Confidence Intervals come into the picture.

Confidence Interval:
A confidence interval is an interval estimate with a range of values for a population statistic.

If a given random variable 'X' belongs to gaussian distribution, then
a) If population standard deviation is known,
The 95% confidence interval is [x_bar - (2*sigma)/sqrt(n), x_bar + (2*sigma)/sqrt(n)]
The 68% confidence interval is [x_bar - (sigma)/sqrt(n), x_bar + (sigma)/sqrt(n)]
For other % of CI like 90%, 80%, 70%, etc, we have to refer to the gausiaan distribution table.

b) If population standard deviation is not known, we then have to go for student's t-distribution for estimating the confidence interval.

c) If the given distribution is not gaussian, then we have to go for Bootstrapping technique.

Let x_bar represent the sample mean and mu represent the population mean. Now, if we repeat the sampling multiple times, each time, we get a different value of sample mean, x_bar. In 95% of the sampling experiments, mu will be between the endpoints of the C.I calculated using x_bar, but in 5% of the cases, it will not be. 95% C.I does NOT mean that mu lies in the interval with a probability of 95%.

that's why Instead of saying that there is a 95% chance that the population mean lies within the interval, it is correct to say that there is a 95% chance that the confidence interval you calculated contains the true population mean.

95% confidence in case of normal distribution, where 2 std-dev hold 95%information. given mean, std-dev we can give confidence interval using [mean-2*std-dev , mean + 2*std-dev] with 95% confidence.

Actually, what we told is instead of saying population mean of heights is 168.5, if we say it lies in the interval of [162.1, 174.9] with the 95% probability it will give much intuition thats what the confidence interval means. Later chapters we will discuss how to achieve this interval. Please continue to watch the next videos.

thumb rule means 68-95-97 which states that 68% of the data is within 1 standard deviation, 95% is within 2 standard deviation, 99.7% is within 3 standard deviations But C.I gives us a range of values in which the mean would lie with high probability because it is always not possible to consider whole data and find the mean, std values in real time many of the times you will be working with samples.
2. Please check out the comment thread: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/confidence-interval-c-i-introduction/#div-comment-40329
https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2859/confidence-interval-ci-introduction/2/module-2-data-science-exploratory-data-analysis-and-data-visualization#comment13137

When we are considering the average of a particular dataset, we are neglecting 50% of points which are below the average when trying to analyze the given dataset.
example: Let us consider the following example of salary of 10 people
[75,5,100,1000,25,60,500,22,44,56]
for this the mean is 188.7, so can I infer by saying that the average salary of this population is 188.7. Does this justify the amount of people way below the mean of the sample.
To over come this can I now say that the CI of the population is ~ [23.5,325] values higher are always safe, now with this the lower set of values are also close to the interval and hence it helps in understanding the range of the population we are dealing with and helps in giving more insights of the population.

Good example. But, we compute a C.I for a metric like mean or std-dev and not a population. So, we can say that "*95% of the salaries lie in [23.5,325]*" but NOT that "*C.I of the population is [23.5 and 325]*". As you rightly pointed out, stating that "*95% of the salaries lie in [23.5,325]*" is more informative than just stating the mean salary.

It is correct to say that there is a 95% chance that the confidence interval you calculated contains the true population mean. It is not quite correct to say that there is a 95% chance that the population mean lies within the interval. **What's the difference?** The population mean has one value. You don't know what it is (unless you are doing simulations) but it has one value. If you repeated the experiment, that value wouldn't change (and you still wouldn't know what it is). Therefore it isn't strictly correct to ask about the probability that the population mean lies within a certain range. In contrast, the confidence interval you compute depends on the data you happened to collect. If you repeated the experiment, your confidence interval would almost certainly be different. So it is OK to ask about the probability that the interval contains the population mean. What you can say is that if you perform this kind of experiment many times, the confidence intervals would not all be the same, you would expect 95% of them to contain the population mean, you would expect 5% of the confidence intervals to not include the population mean, and that you would never know whether the interval from a particular experiment contained the population mean or not.

Actually in case of normal distribution i.e., If we know that the population mean follows a normal distribution (as per CLT for any r.v with finite mean and std-dev) with mean=mu and std-dev=sigma, we know that 95% of the observations of this population mean would lie in the +/- 2 std-dev range from mean. But here If we have a single sample of points, applying CLT is not possible. It is best to use the sample std-dev as an approximation of the population std-dev to compute C.I. Note that this is a suboptimal choice to compute the C.I. It is better to use bootstrapping based approach to estimate the C.I as discussed in this next video. further reading: kindly, check out the link: https://www.mathsisfun.com/data/confidence-interval.html

for a 95% confidence interval, you want to know how many standard deviations away from the mean your point estimate is i.e., the "z-score" . Please go through the video:

https://www.youtube.com/watch?v=grodoLzThy4&feature=youtu.be

First of all, if we ae given the entire population data, then if we want to compute the population mean(μ), we can easily obtain it using the formula (sum of observations)/(number of observations). This is the exact point estimate when the entire population data is given. We need not go for estimting the interval estimates(ie., confidence intervals)

In case, if we do not have the entire population data, but we are provided only with a sample of

data, then we have to go for interval estimates because the sample mean could not be exactly equal to the population mean.
K% Confidence Interval says that the in K% of the experiments/cases, the population statistic(here it is mean) lies in this interval and in the remaining (100-K)% cases, this confidence interval doesn't contain the population statistic.

If we want to compute the 90% confidence interval for the population mean, and let this CI be [value1,value2], in this case, this confidence interval says that in 90% of the cases, the this confidence interval [value1,value2] contains the true population mean and in the remaining 10% of the cases, this interval doesn't contain the population mean value.

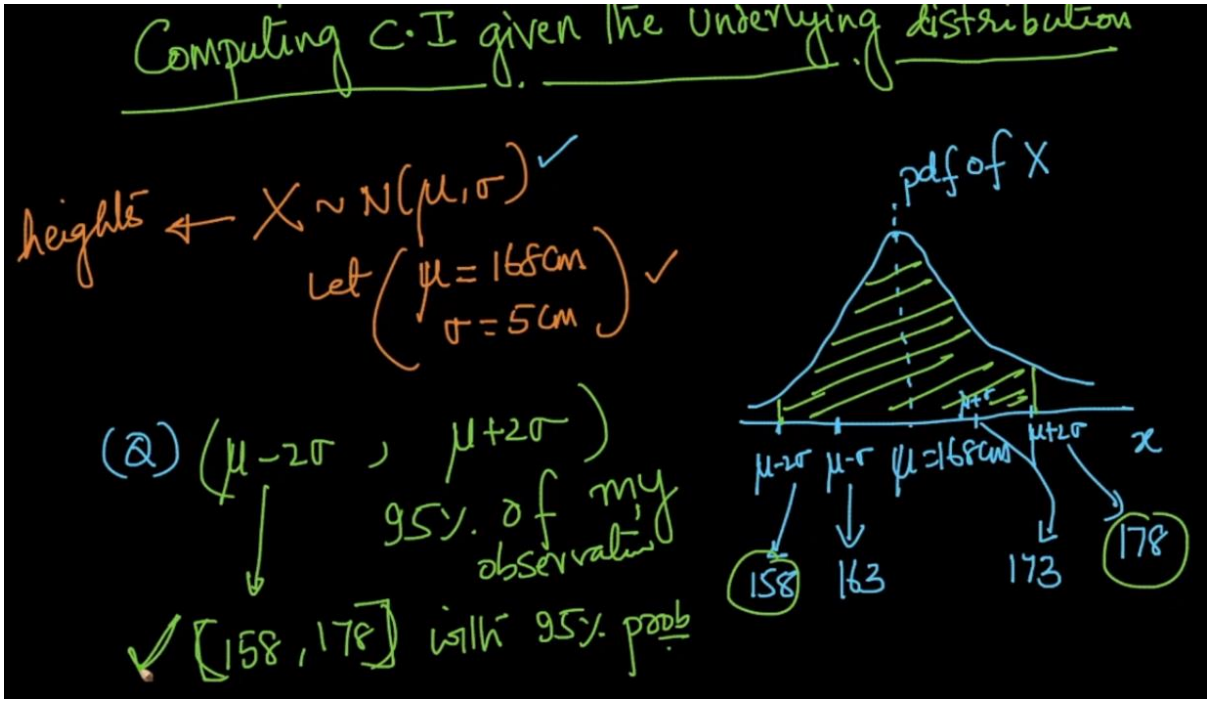
When you are working on inferential statistics, you have to make decisions based on the statistics obtained on the sample data because it is highlly difficult to gather the exact and entire population data. if we have the entire population data, then there is no need to go for inferential statistics at all. We could go directly for point estimates rather than going for interval estimates. As the entire population data is not available, we could not get the exact point estimates and so we are going forward with computing interval estimates and make decisions based on them.

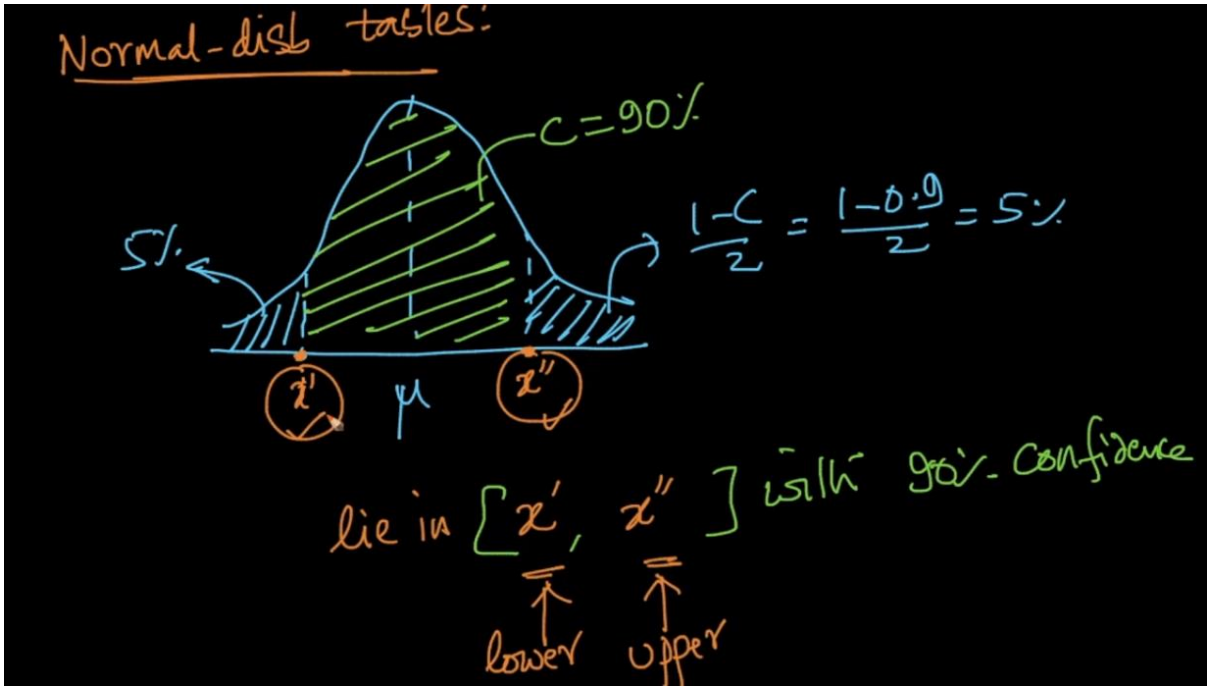
The 95% confidence interval defines a range of values that you can be 95% certain contains the population mean. It is correct to say that there is a 95% chance that the confidence interval you calculated contains the true population mean. If you want more confidence (99%) that an interval contains the true parameter, then the intervals will be wider. If you want to be 100.000% sure that an interval contains the true population, it has to contain every possible value so it'll be very wide. If you are willing to be only 50% sure that an interval contains the true value it'll be much narrower.

For more details please go through the link:
https://stats.stackexchange.com/questions/26450/why-does-a-95-confidence-interval-ci-not-imply-a-95-chance-of-containing-the

Computing confidence interval given the underlying distribution

confidence interval (**CI**) is a range of estimates for an unknown parameter, defined as an interval with a lower bound and an upper bound.





C.I for mean of a random variable

C.I for mean(μ) of a r.v

$X \sim F$ with pop-mean of μ & std-dev of σ

$\downarrow$ $\{x_1, x_2, \ldots x_{10}\}$ → sample of size = $n = 10$

✓ $\{180, 162, 158, 172, 168, 150, 171, 183, 165, 176\}$ - given this sample

(Q) What is the 95% C.I of μ.

Case 1: $\sigma = 5$ cm {we know pop-std-dev}

CLT: $\bar{x}$ = Sample mean = $\frac{1}{10}\sum_{i=1}^{10} x_i$ ($n=10$)

$\bar{x} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$ → CLT

$\bar{x}$: Sample-mean; μ: pop-mean; $\frac{\sigma}{\sqrt{n}}$: $\frac{\text{pop-std-dev}}{\sqrt{n}}$

$\left\{\mu \in \left[\bar{x} - \frac{2\sigma}{\sqrt{n}}, \bar{x} + \frac{2\sigma}{\sqrt{n}}\right]\right\}$ with 95% confidence

$\bar{x} - \frac{2\sigma}{\sqrt{n}}$: $\mu - 2\sigma$; $\bar{x} + \frac{2\sigma}{\sqrt{n}}$: $\mu + 2\sigma$

Case2: if we dont know σ (pop-std-dev)

Sample of n

t-distb

Student's t-distb

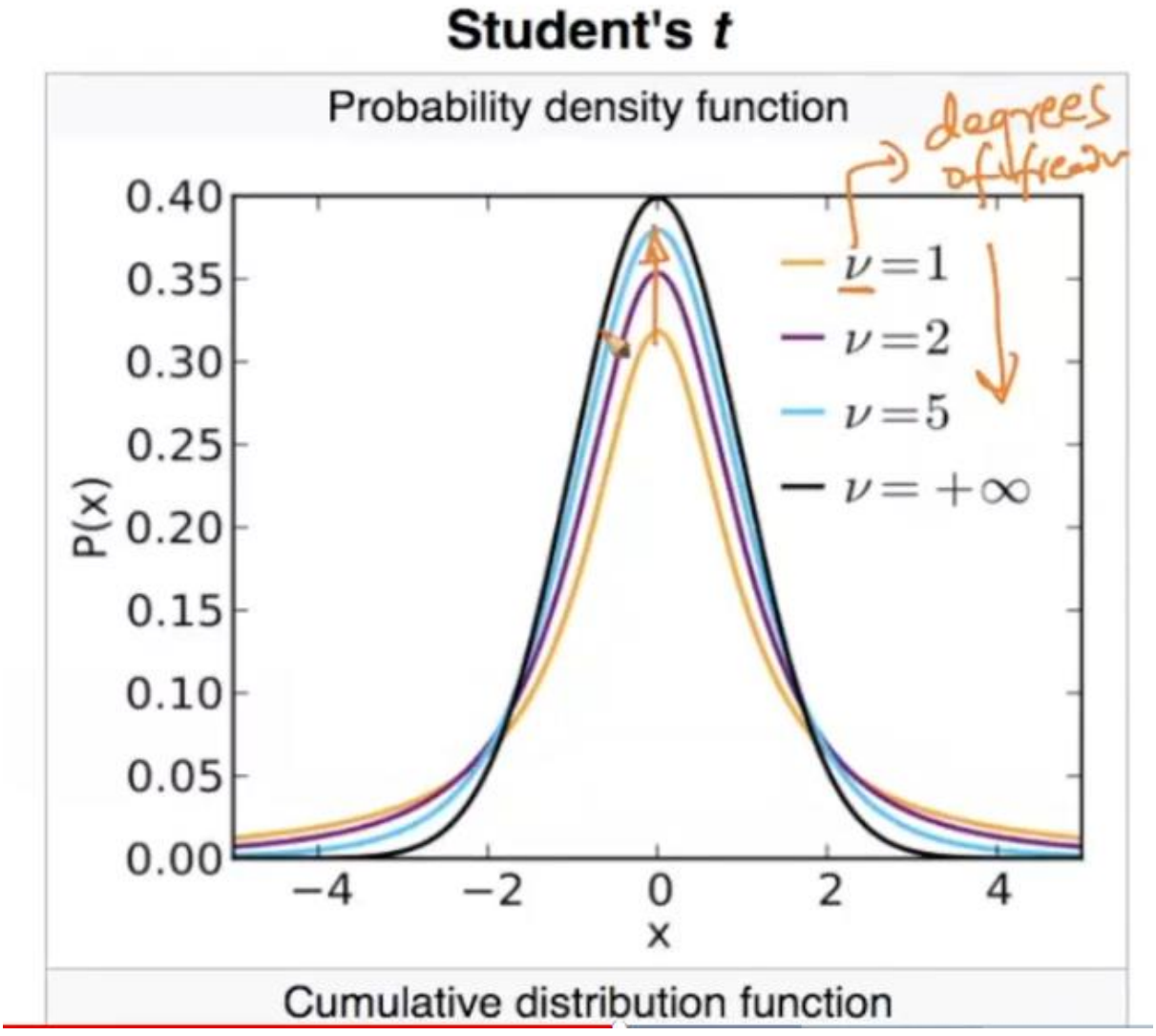
{ C.I of mean of a r.v when σ is unknown }

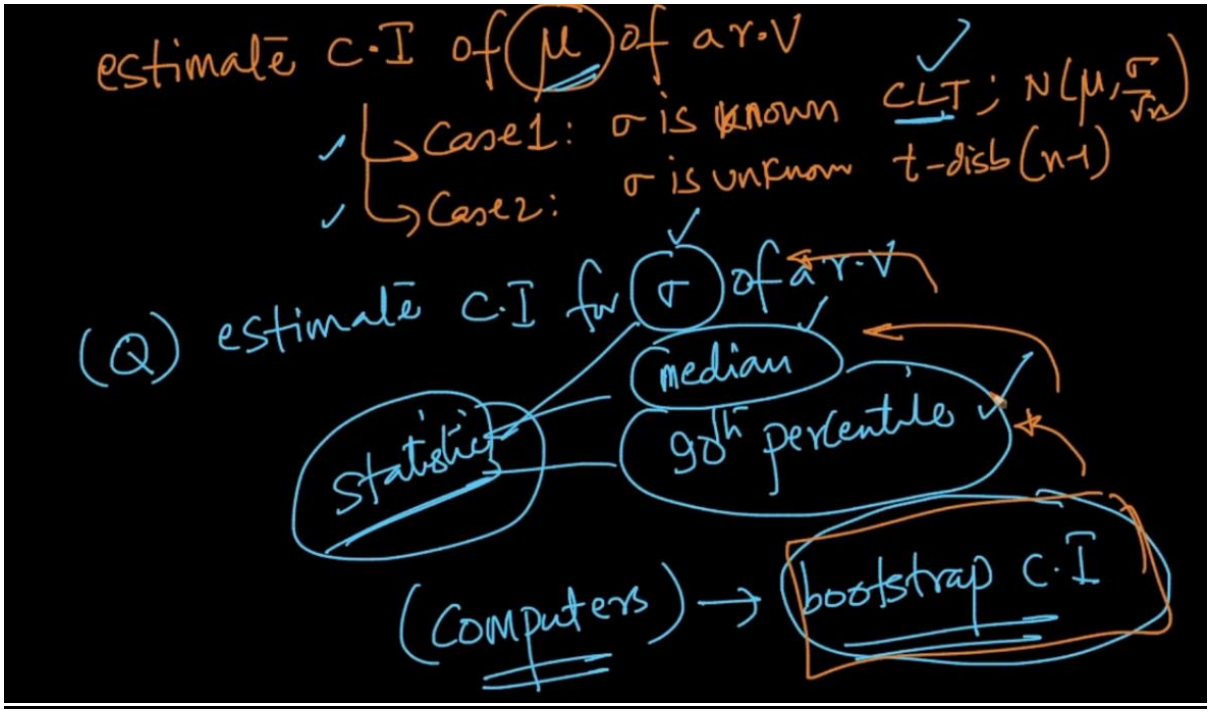
$\bar{x} \sim t(n-1)$

$\bar{x}$: Sample mean

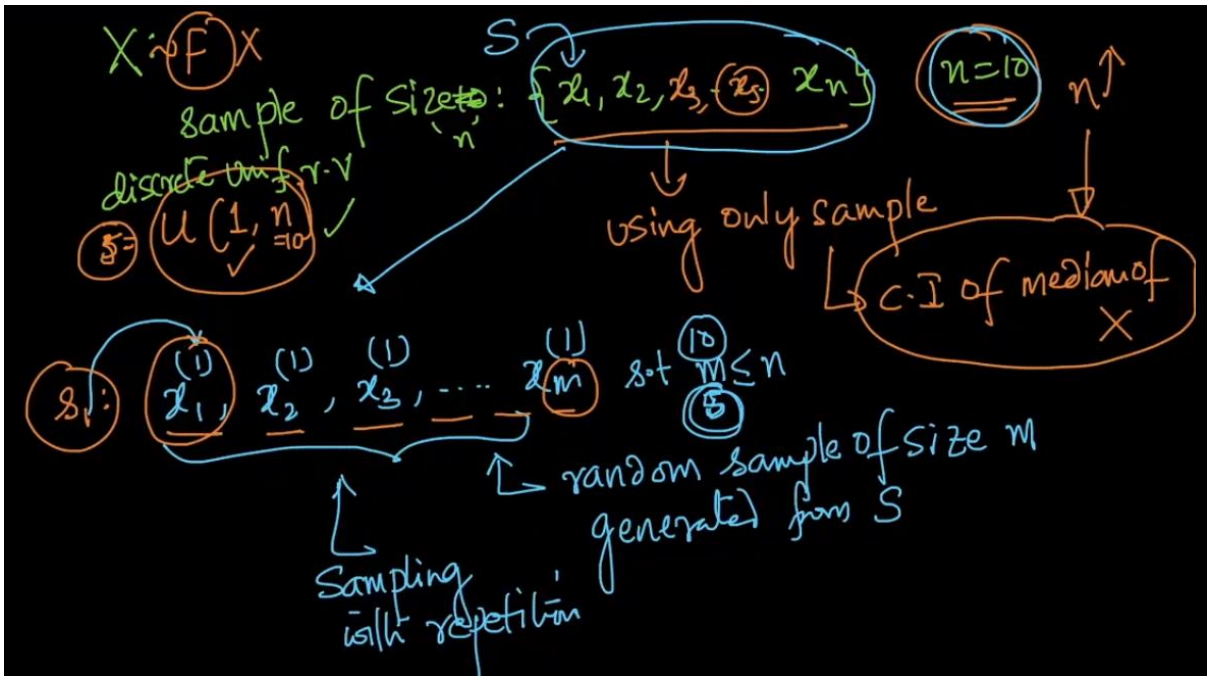
t : t-distb

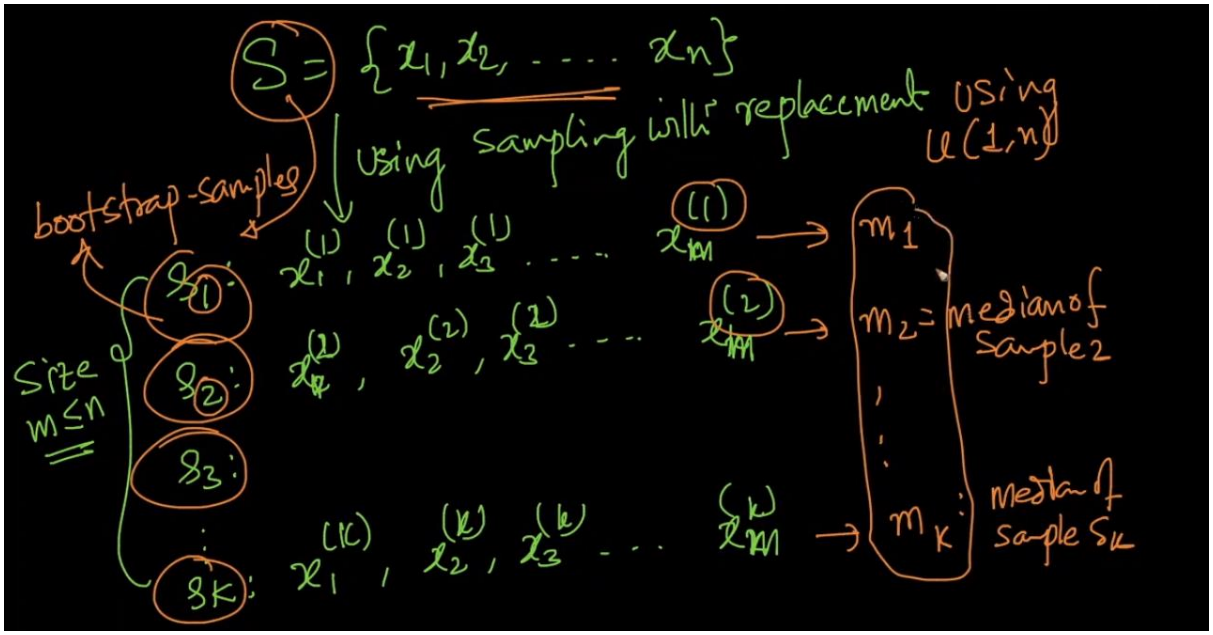
(n-1) : degrees of freedom

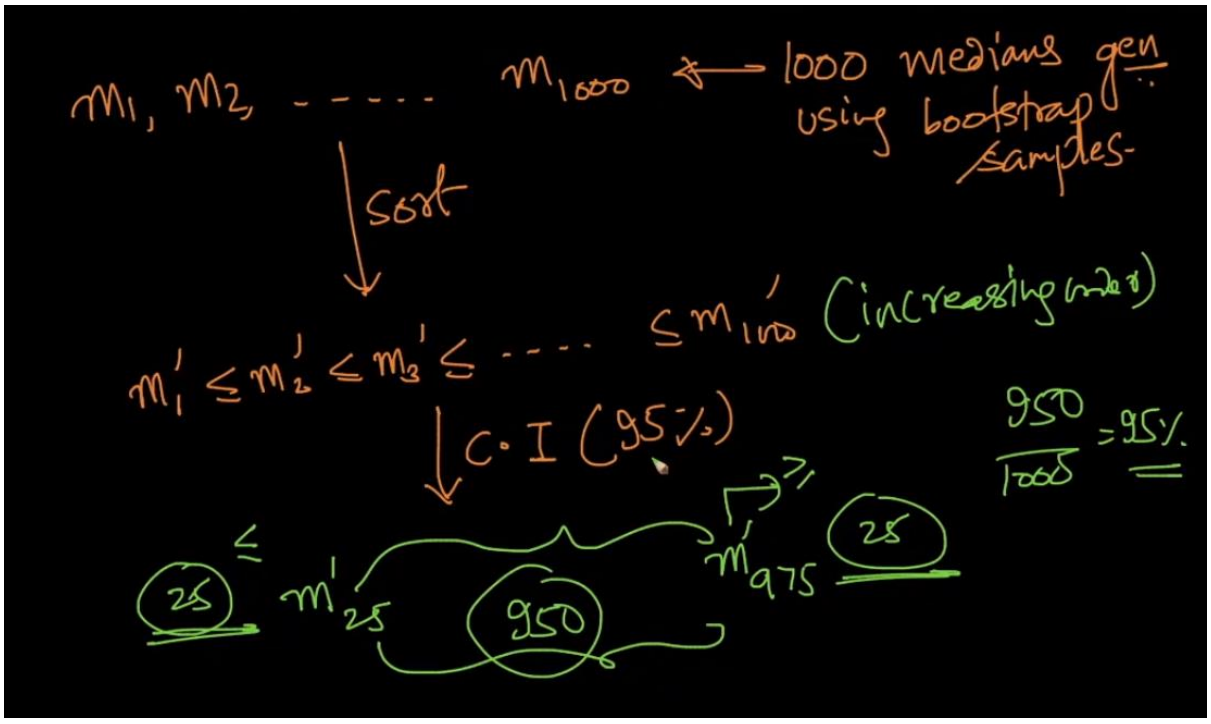


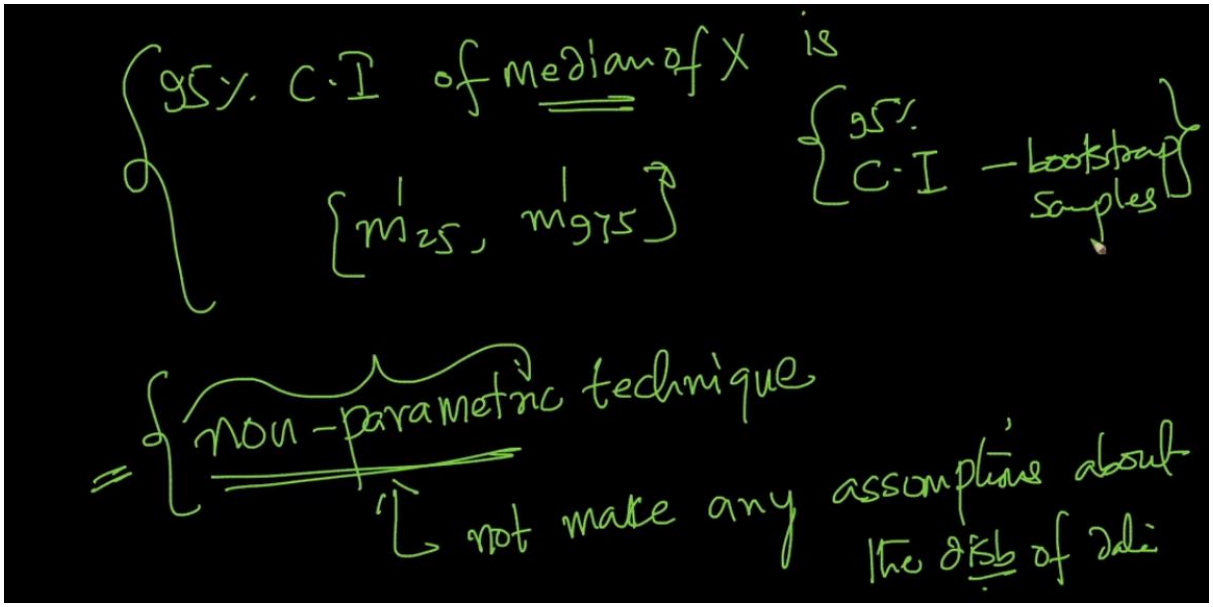


C.I. using Empirical Bootstrapping



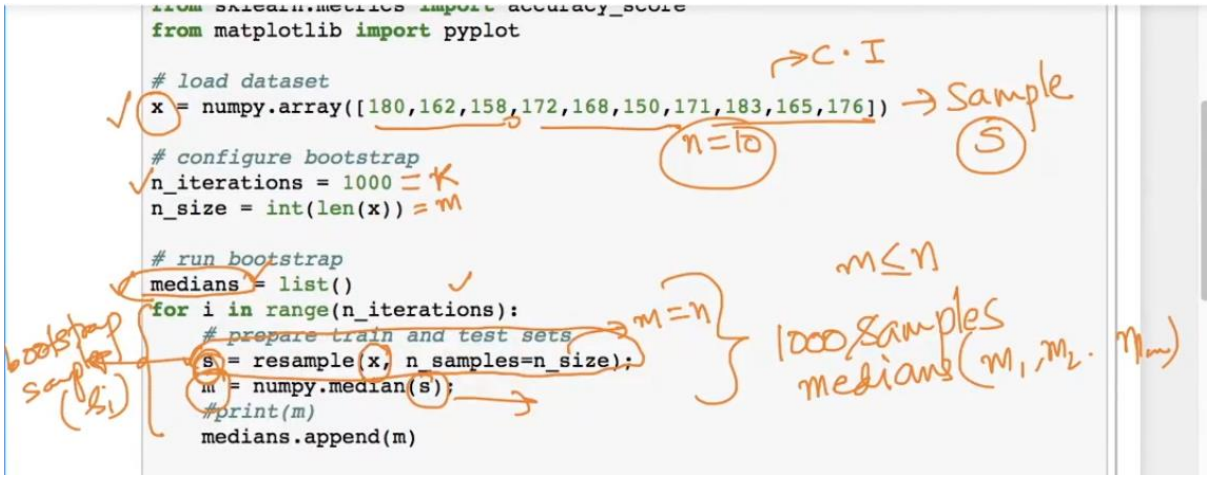

S = {x1, x2, ... xn}
Using sampling with replacement using U(1,n)
bootstrap-samples
Size m ≤ n
S1
S2
S3
Sk
median of sample 2
median of sample Sk


m1, m2, ... m1000
1000 medians gen using bootstrap samples
sort
m'1 ≤ m'2 ≤ m'3 ≤ .... ≤ m'1000 (increasing order)
C.I (95%)
950/1000 = 95%
25
25
950
m'25
m'975



https://stats.stackexchange.com/questions/26088/explaining-to-laypeople-why-bootstrapping-works

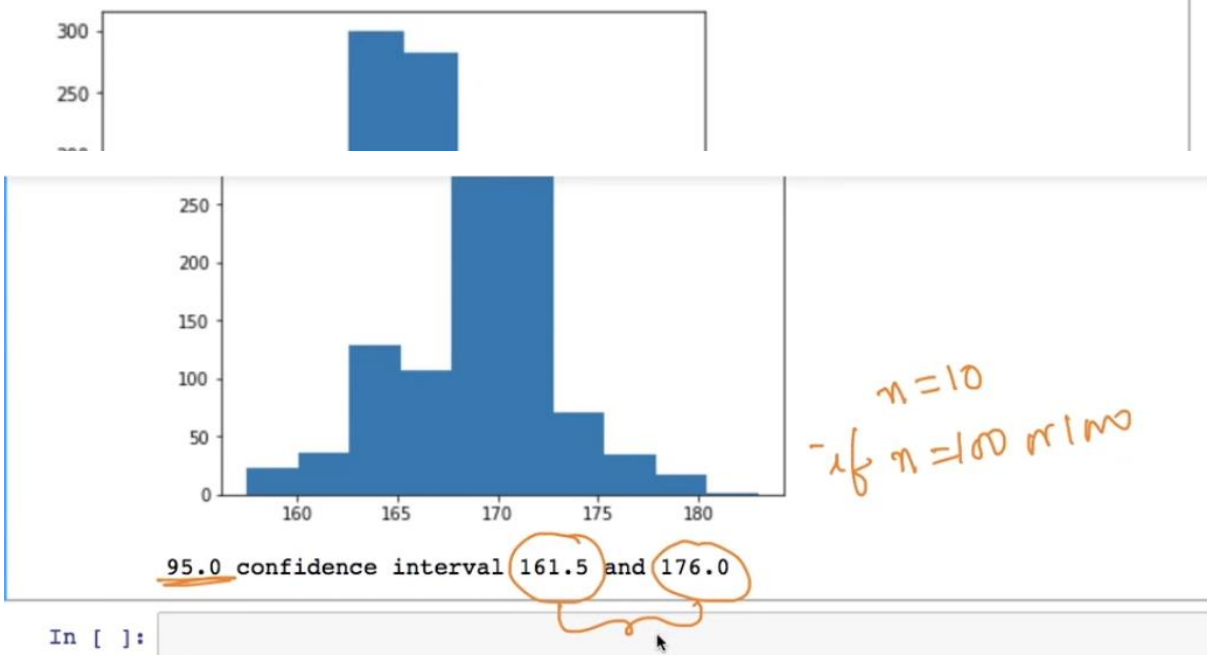
"Since the sample is the only information we have about the population, we take our sample itself as a model of population. In this case, resampling is not done to provide an estimate of the population distribution. Rather, resampling is done to provide an estimate of the sampling distribution of the sample statistic in question."



```
# plot scores
pyplot.hist(medians)
pyplot.show()

# confidence intervals
alpha = 0.95
p = ((1.0-alpha)/2.0) * 100
lower =  numpy.percentile(medians, p)

p = (alpha+((1.0-alpha)/2.0)) * 100
upper =  numpy.percentile(medians, p)
print('%.1f confidence interval %.1f and %.1f' % (alpha*100, lower, upper))
```



CLT also speaks in terms of confidence i.e how confident we can be that a sample mean will fall within some interval surrounding the population mean. yes more samples in bootstrap more is our confidence

Bootstrapping is used to obtain an interval estimate of a population statistic from a given sample. bootstrapping use samples to draw inferences about populations. To accomplish this goal, these procedures treat the single sample that a study obtains as only one of many random samples that the study could have collected.

Now, suppose an analyst repeats their study many times. In this situation, the mean will vary from sample to sample and form a distribution of sample means. Statisticians refer to this type of distribution as a sampling distribution. Sampling distributions are crucial because they place the value of your sample statistic into the broader context of many other possible values.

While performing a study many times is infeasible, both methods can estimate sampling distributions. Using the larger context that sampling distributions provide, these procedures can construct confidence intervals and perform hypothesis testing.

Central Limit Theorem

According to the central limit theorem, the mean of a sample of data will be closer to the mean of the overall population in question, as the sample size increases.

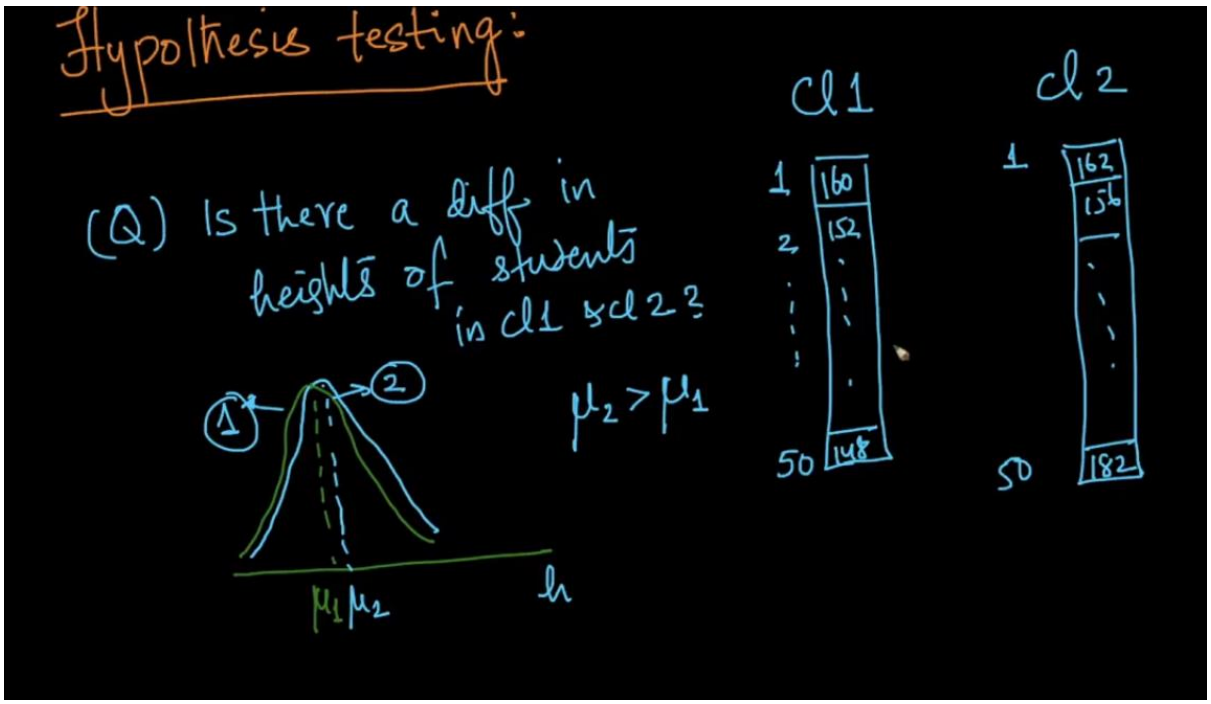
Average of the sample means and standard deviations will come closer to equaling the population mean and standard deviation as the sample size grows, which is extremely useful in accurately predicting the characteristics of populations.

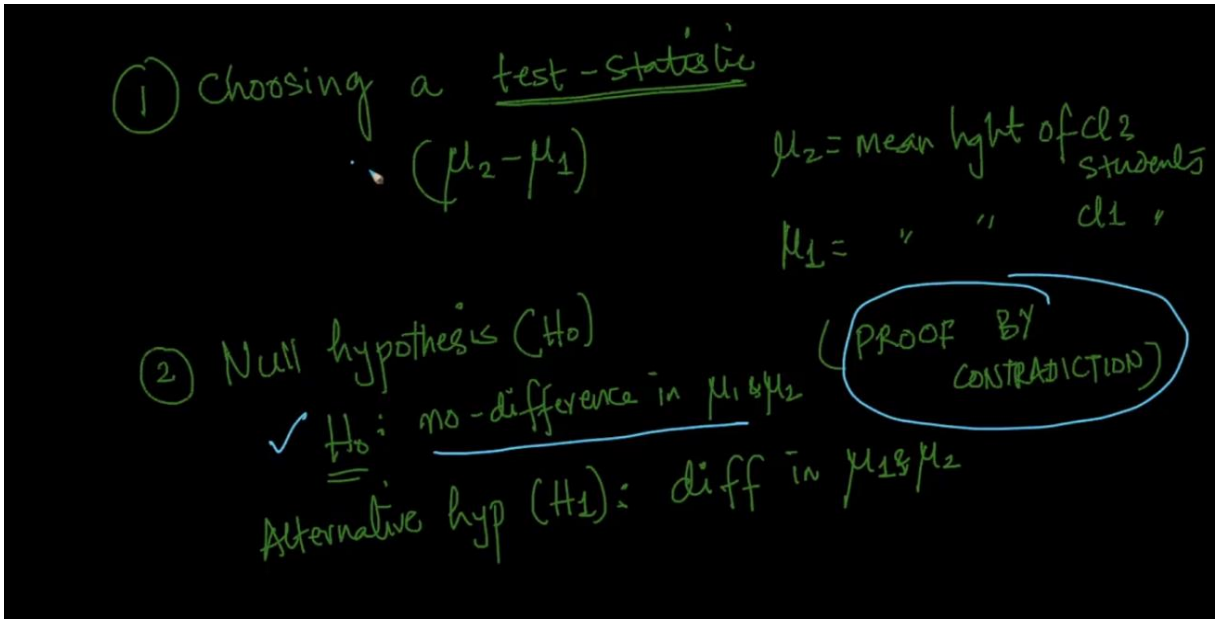
CLT does not work with any function f(x) on r.vs. It primarily works on the sum of r.vs. In a nutshell, the sum of a number of independent and identically distributed random variables with finite variances will tend to a normal distribution as the number of variables grows. Note that mean is a division of sum by a constant and hence, CLT works for the mean. If f(x) is an addition or addition followed by an operation with a constant, CLT will apply.

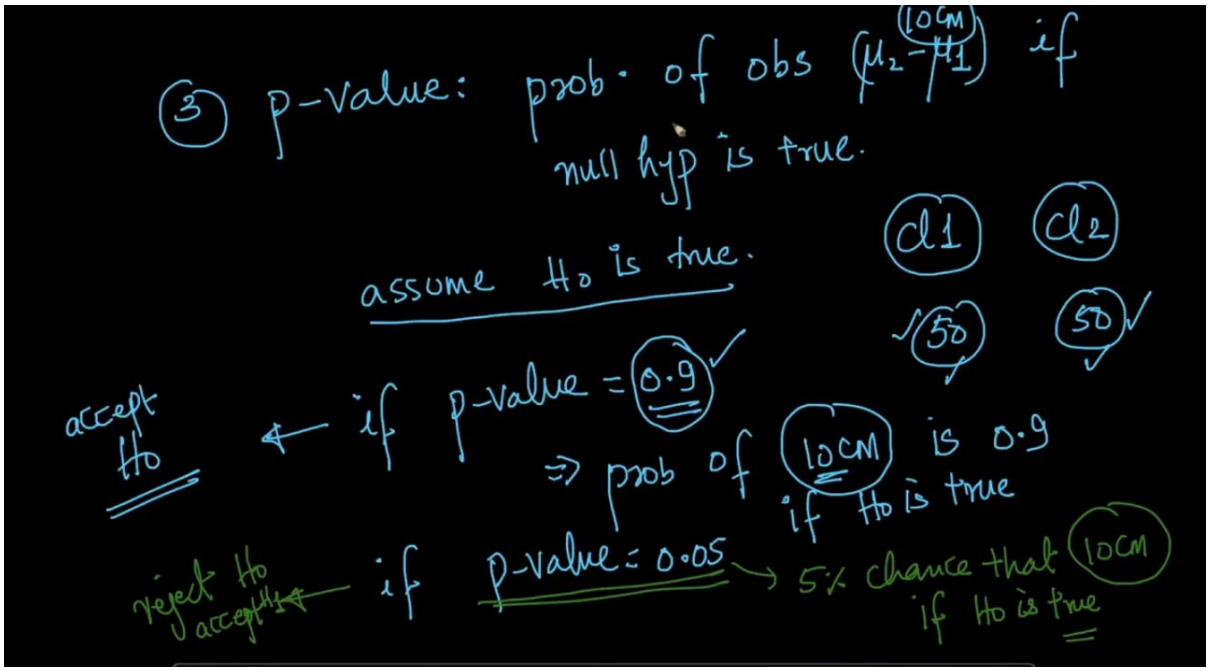
1. We can pick any fixed sample size for all samples. 30 is just a rule of thumb for minimum sample size. Larger the sample size, better is the sample-mean estimate and hence better is the population mean that we estimate using CLT.

   2. We typically choose fixed size samples while using CLT. So, each sample is of the same size. I haven't come across cases where each sample could be of different size. But, my knowledge of CLT is limited as I have not studied all of its many variations and tons of research papers covering many variations of CLT.

Hypothesis testing methodology, Null-hypothesis, p-value







if p-value=0.95, there are 95% of our random experiments have difference(mu2-mu1) as high as ground truth(i.e 10 cm) {it does not mean zero}. and we performed this random experiment under the condition that our H0 is true. (we achieved this by taking random 50-50 samples after combining). so since under H0 p-value is high we accept H0

"p = 0.95,i.e probability of getting difference 10cm is 95%" - p-value is not the probabilty of dfference , Please not that : p- value is the probability of obtaining a value of your test statistic that is at least as extreme as what we observed, under the assumption the null hypothesis is true. It is not the probability that the null hypothesis is true.

https://www.youtube.com/watch?v=2VE6AZPS6bI&feature=youtu.be&t=1s

Hypothesis Testing Intution with coin toss example

Hypothesis Testing: → confusing idea

example 1:

Task: Given a coin, determine if the coin is biased towards heads or not

basic probability

- ✓ biased towards heads :- $P(H) > 0.5$
- not-biased " " : $P(H) = 0.5$

---

design: expt: flip a coin (5) times and count # heads = (X) ← Test-Statistic

r.v

"perform" expt: f, f, f, f, f

↓ ↓ ↓ ↓ ↓

H H H H H

(X = 5) ← observation by expt

$P(X=5 \mid \text{coin is not biased towards heads}) = p(\text{obs} \mid H_0)$

obs

assumption

✓ Null-hypothesis (H_0)

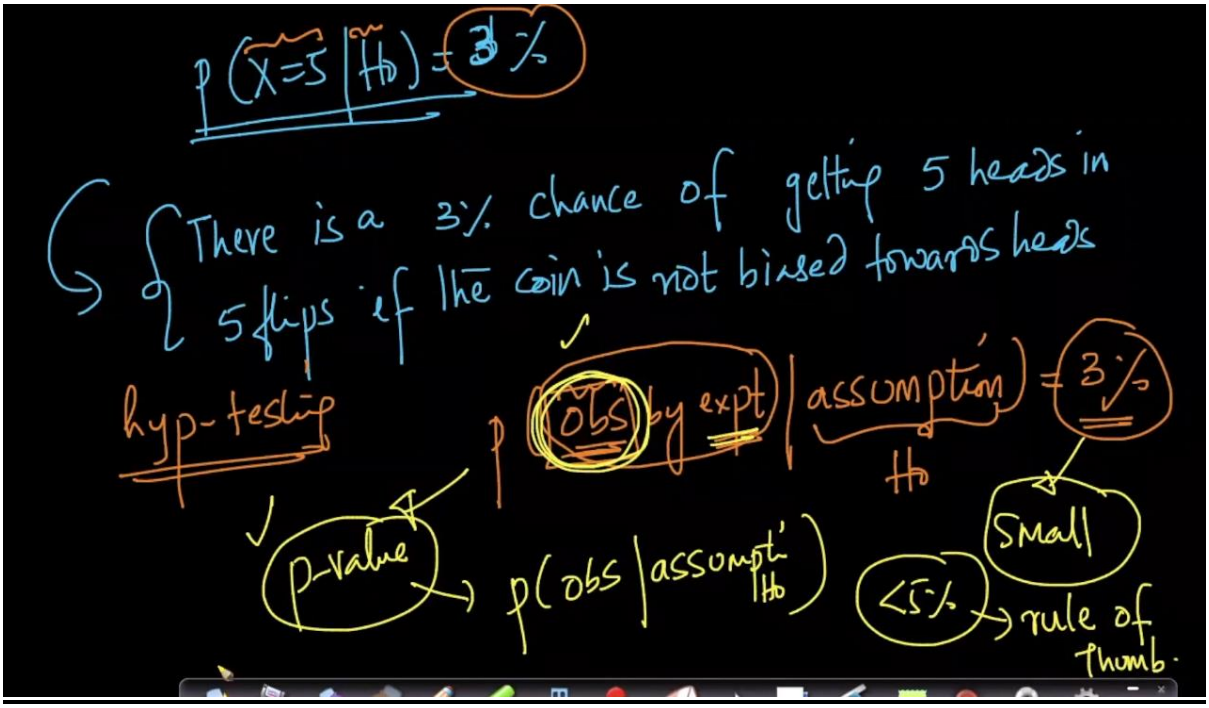
H_0 : coin is not biased towards heads

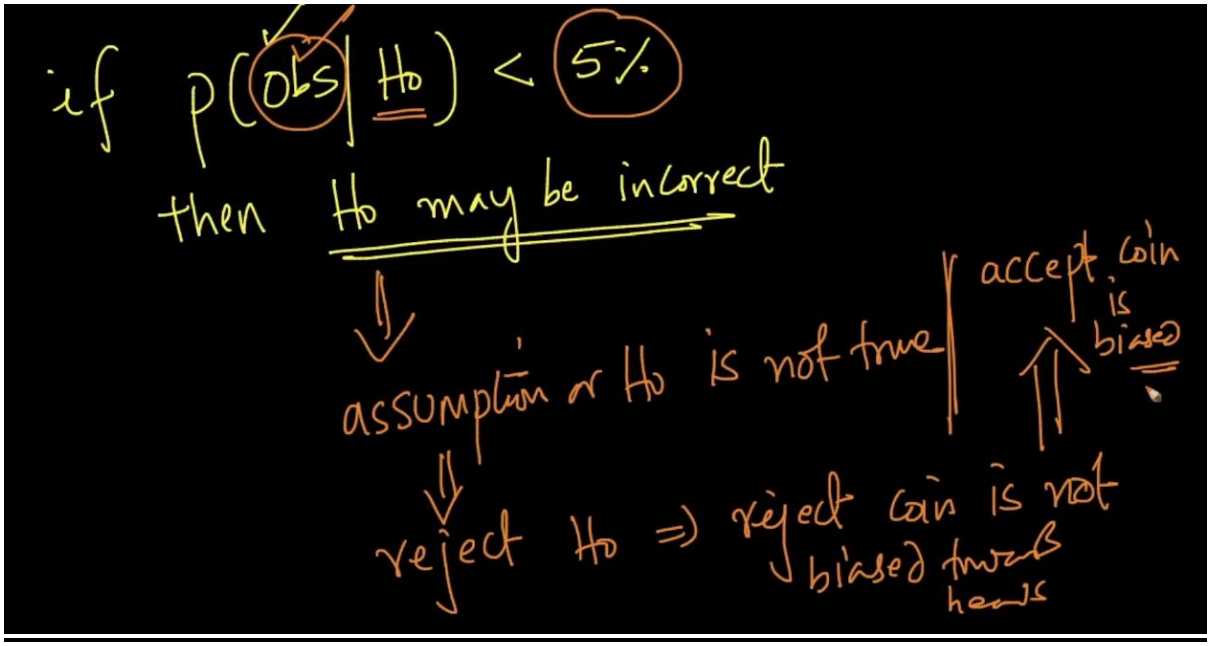
$P(X=5 \mid H_0) = 3\%$

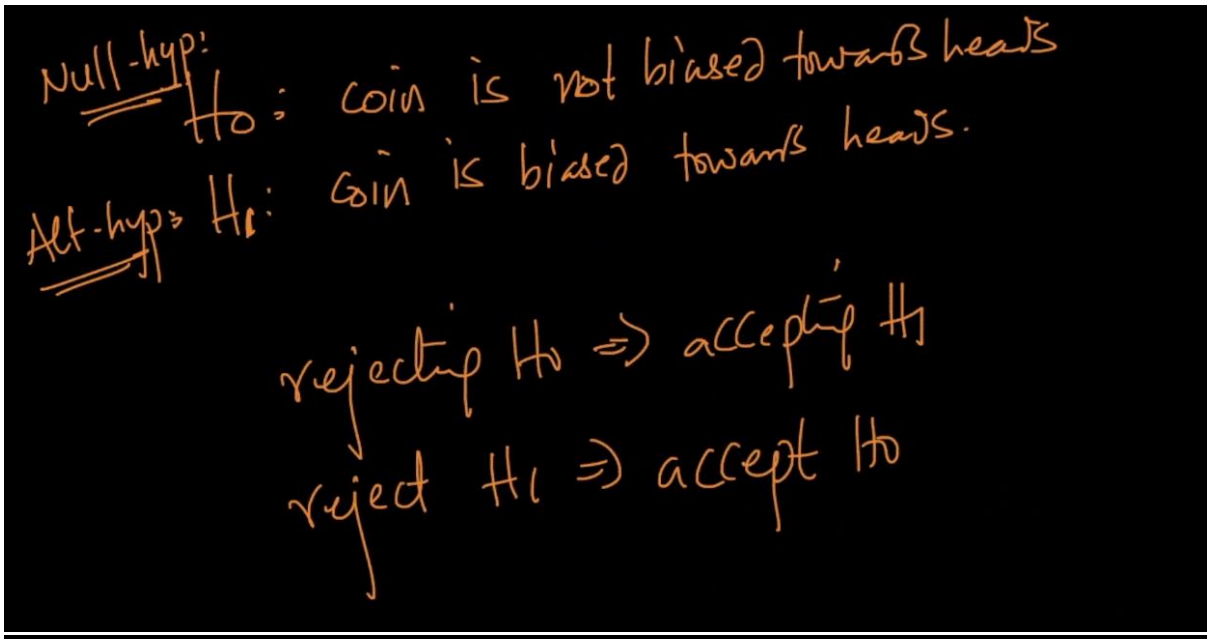
There is a 3% chance of getting 5 heads in 5 flips if the coin is not biased towards heads

hyp-testing

$P(\text{obs by expt} \mid \underbrace{\text{assumption}}_{H_0}) = 3\%$







Resampling and Permutation:

Step1- Calculate the max distance between both the distributions.(big delta)

Step2- Null hypothesis would be the there is no difference in both distribution.

Step3-Therefore in order to simulate this, we combine the data of both distributions into one and then randomly sample the points into two sets.

Step4- We calculate the max distance between both the sets.

Step5-We repeat Step 3 and 4 for some let's say 1000 times.(Small deltas)

Step6- We sort the obtained results and based on the position of big delta we either accept or reject the null hypothesis.

KS Test

There are extensions to KS-Test to Multivariate Distributions as referenced here. My preferred approach is as follows: A multivariate sample is nothing but a matrix of observations with each row an observation and each column a variable/dimension. Now, given two matrices of observations, we need to measure if they have the same disb or not. One way to work this out using ML(classification) as follows: 1. Give the data-points in the first matrix a label or y_i of 1 and the second matrix a class label of 0. 2. Now try various models to separate the classes. If the model performance of the model is bad, that implies that the two matrices are not separable which implies they have the same underlying disb. If the model performance is good, that implies that two matrices are from different disb and hence we are able to separate them.

1. p-value is dependent on D.

2. No, it is derived in the following video.

3. We can decide to reject Ho by comparing D and c(alpha)*sqrt((n+m)/nm) or by computing the p-value and comparing it with alpha. Both of them are equivalent.

Video reply: https://youtu.be/C04oZK6I5k4

p-value = P(test-statistic | H0) while significance level is suppose we reject H0 if p-value is less than 0.05 (this 0.05 is significance level) then here we have a risk of 5% where we make false conclusions.refer this

Hi Team, Could you please help confirming my understanding on below questions.

1. Is Alpha and p-value same?

2. In formula D> c(alpha)*sqrt(m+n/m*n) the right side c(alpha)*sqrt(m+n/m*n) will give a D value lets assume D1. Now this D1 is based on a selected alpha let it be alpha1=0.05. So can i say there is 5% probability that Max gap will be >= D1?

3. Now using the formula i believe we can also calculate an alpha for our D as D=c(alpha)*sqrt(m+n/m*n).

so can i say

i. when D > D1 => alpha < alpha1 : H0 Reject

ii. when D alpha > alpha1 : H0 Accept

iii. when D = D1 => alpha = alpha1 : H0 Accept

4. When we are using KS test using code it returns D and calculates alpha/p-value which will satisfy the equation D=c(alpha)*sqrt(m+n/m*n).

5. Is K-distribution for a particular N & M independent of empirical distribution function we are comparing i.e. is K-dist always same for particular N & M irrespective of what distribution we are comparing?

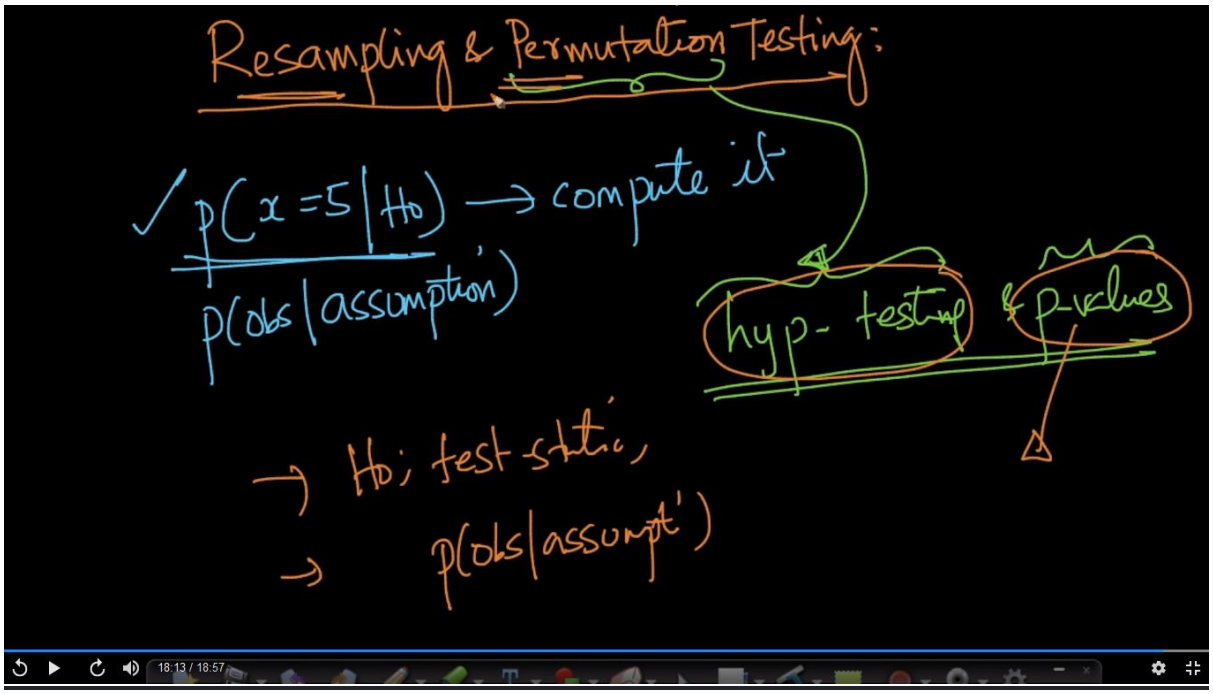
1. alpha is a threshold (or significance level) we choose to accept or reject H0 like 5%, 3% or 1%. We choose the value of alpha based on how rigorous we want to be when we reject H0. Once alpha is chosen, we compare the value of alpha with the p-value obtained using the data and if p-value > alpha we accept H0 else reject it. 2. No, the correct statement would be: "We reject the null-hyp that both distributions are same if D>D1 at a significance level of 5%.". 3. First, we pick an alpha based on our problem like 5%, 3% or 1% or 10%. ONce we pick an alpha, we compute to check if D>D1. If so, we reject H0 at alpha-significance level. 4. Yes, that is correct with scipy.stats.kstest 5. The Kolmogorov distribution is dependent only on the max-gap and not on the values of m and n as can be seen from it's CDF here.

Alpha is a pre-determined significance value like 5% or 3% or 1% that we use as a threshold on the p-value to determine if we accept H0 or reject it.

The p-value is the computed p(obs|H0) from the data.

If p-value > alpha, we accept H0, else we reject it.

Hypothesis Testing - Another example

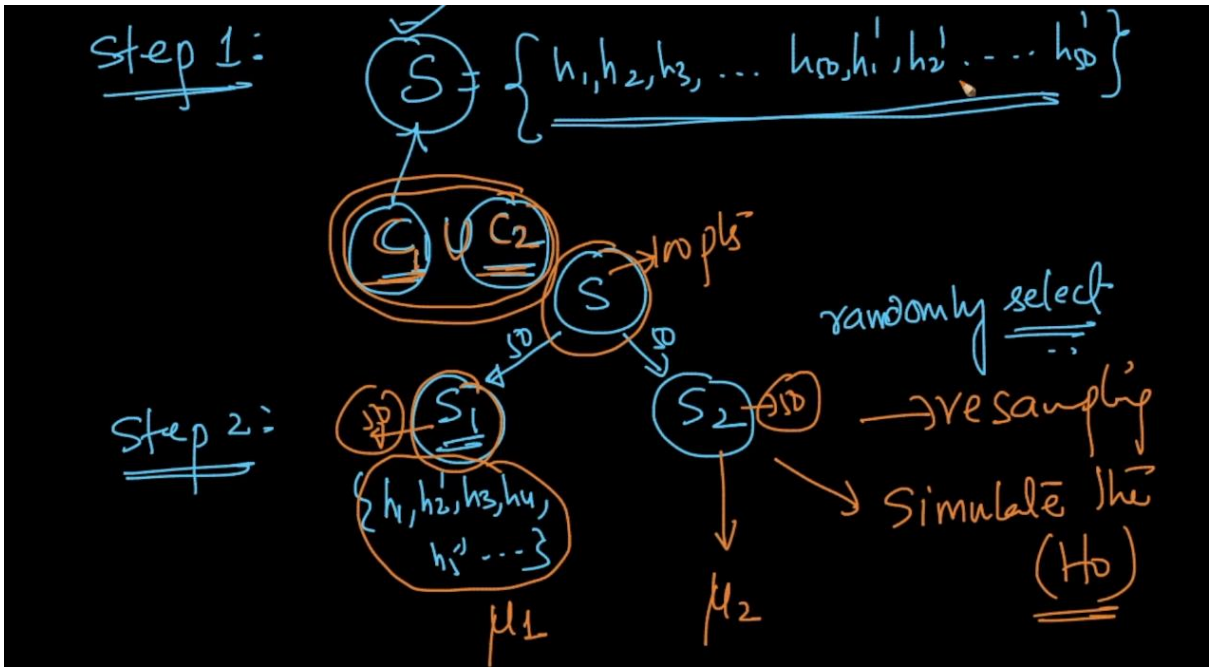


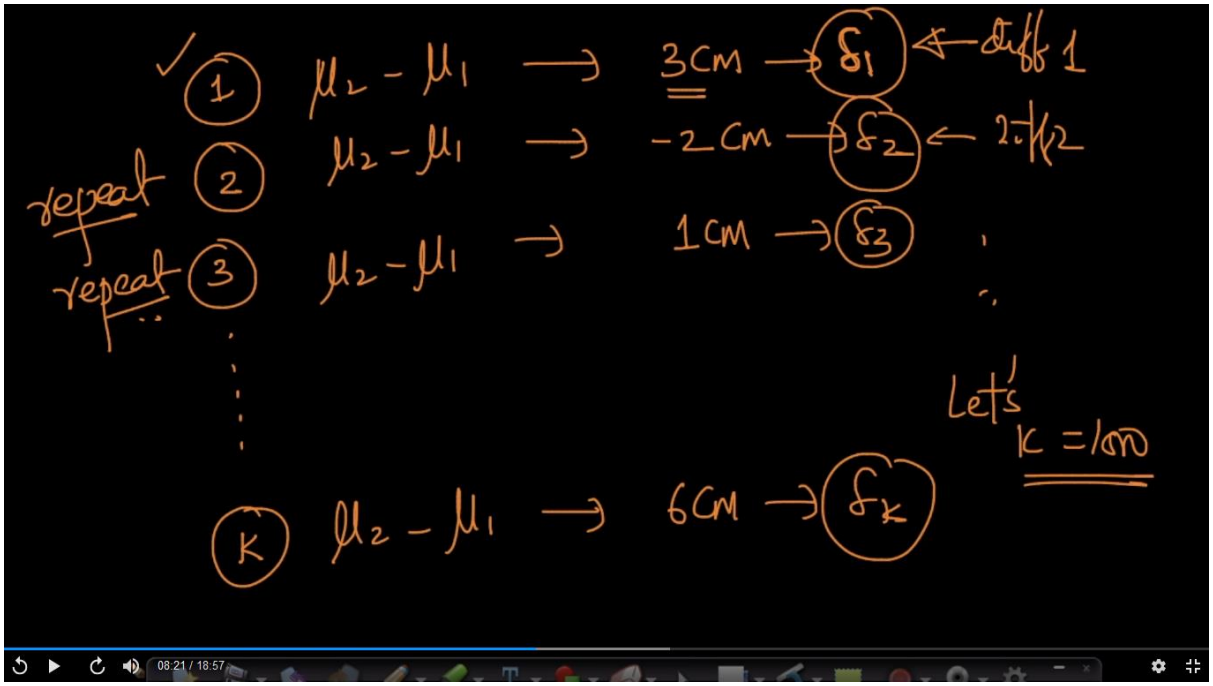
(1) $x = -\mu_1 + \mu_2$ ← diff in sample means with sample size of 50

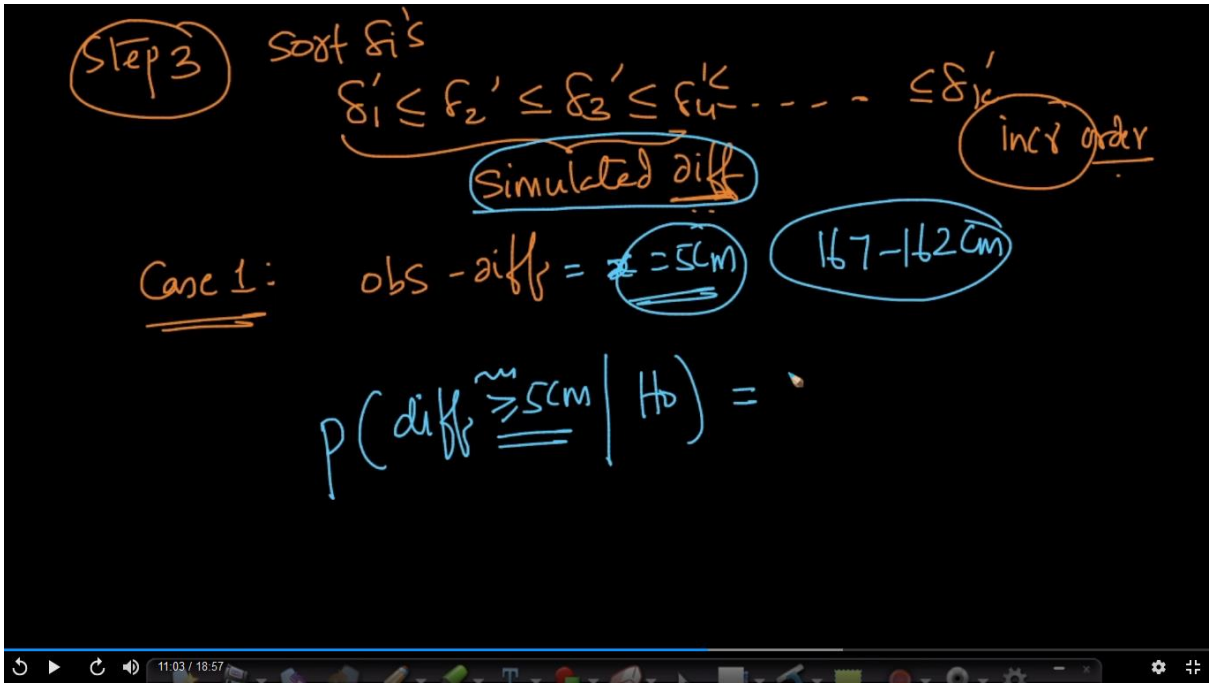
$x = 5cm$ ($-162 + 167$)

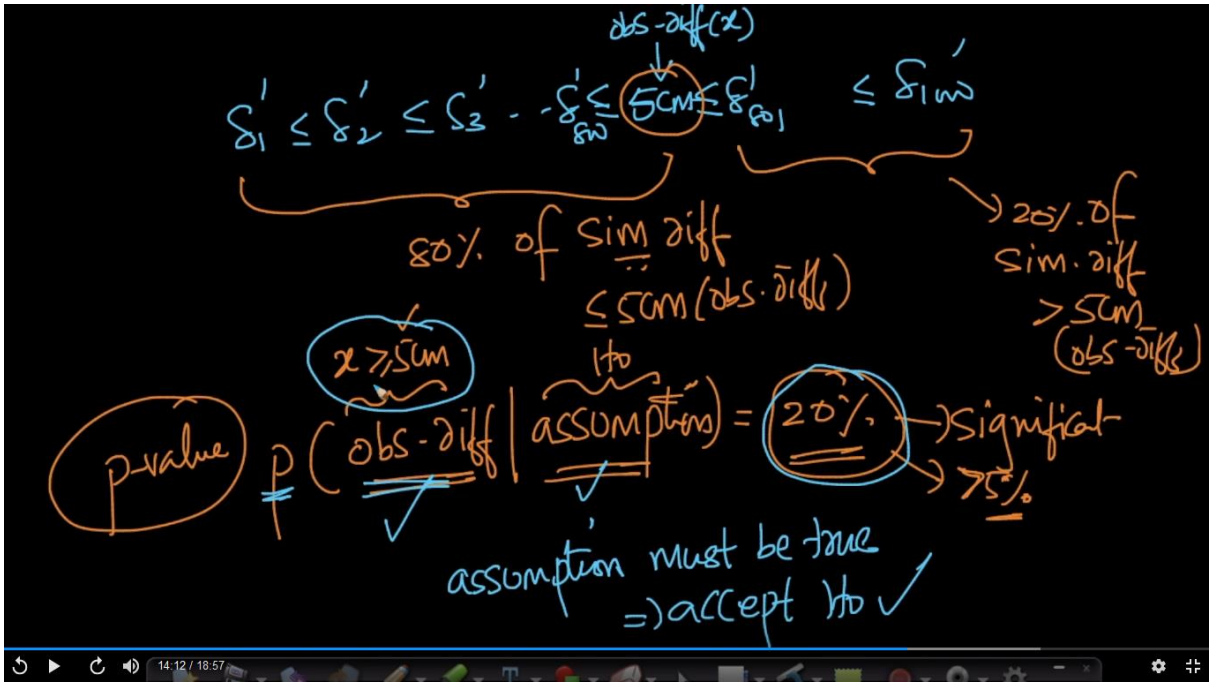
(2) H_0: no diff in population means.

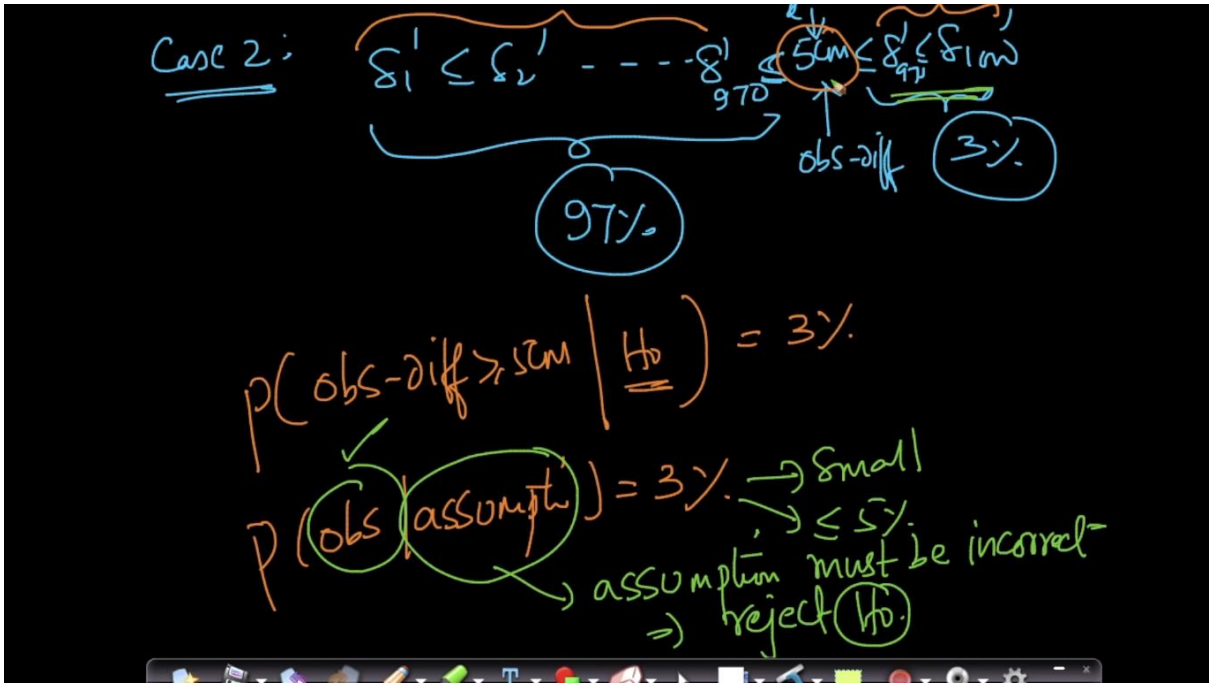
(3) $C_1 = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_{50} \end{bmatrix}$ $C_2 = \begin{bmatrix} h_1' \\ h_2' \\ \vdots \\ h_{50}' \end{bmatrix}$




1 $\mu_2 - \mu_1$ → 3cm → δ_1 ← diff 1
repeat 2 $\mu_2 - \mu_1$ → -2 cm → δ_2 ← 2nd
repeat 3 $\mu_2 - \mu_1$ → 1 cm → δ_3
Let's k = 10m
K $\mu_2 - \mu_1$ → 6cm → δ_k
08:21 / 18:57


Step 3 sort δ_i's
$\delta_1' \leq \delta_2' \leq \delta_3' \leq \delta_4' \cdots \leq \delta_k'$
incr order
Simulated diff
Case 1: obs - diff = 5cm
167 - 162 cm
$p(\text{diff} \geq 5\text{cm} \mid H_0) =$
11:03 / 18:57





In the video, P(x>=5 cm |H0) = 3%, which means the probability of observing the difference given the null hypothesis is 3%, which is very low. So the assumption we made must be incorrect. Hence we reject the null hypothesis.

P(x >= 5cm | H0) = 97% means 97% of the simulated mean differences is greater than or equal to 5cms. our ground truth about both the cities was mean2-mean1 = 5cm. since samples of cities has a mean difference of 5cm and we simulated experiment using the same samples of cities. observing simulated differences of 5cm or more is 97%. so we accept the null hypothesis that there is no difference in population mean. 3% of simulated mean differences has less than 5cm.

When we are computing the p-value, we are asking this following question: How likely is to observe a value of X higher than equal to 5 assuming H0 is true.

Now, if we take p(X=5|H0). This would be zero as X is a real-valued r.v and the probability of X taking an exact value of 5 is zero. For a continuous r.v, the probability of it taking an exact value (like 5) is zero.

Since a continuous distribution describes the probabilities of the possible values of a continuous random variable. A continuous random variable is a random variable with a set of possible values (known as the range) that is infinite and uncountable.

Probabilities of continuous random variables (X) are defined as the area under the curve of its PDF. Thus, only ranges of values can have a nonzero probability. The probability that a continuous random variable equals some value is always zero. If we used X<=5 which also contains X=0 which is part of the null hypothesis, we would not able to apply the hypothesis testing to accept or reject null-hypothesis.

if P(observation|assumption) is small then we reject null hypothesis because we already observed the observation and still we got small probability which means our assumption must be wrong.

if P(observation|assumption) is high (say 0.9) then our assumption is true but probability of difference(observation) is high which occurred due to error in sample and if we increase sample size then error will decrease.

https://medium.com/@matthewparker/common-machine-learning-resampling-methods-like-bootstrapping-and-permutation-testing-attempt-to-ddc4fbbda391

====

Deeper knowledge certainly helps understand more complex topics but having working and applied knowledge is good enough for most practical problem-solving. Knowledge of a breadth of techniques certainly is needed to work in data-science/ML.

1. A sample can be of any size in general in Stats. But in resampling for hypothesis testing, the sample has to be the same size as the original sample to simulate H0.

2. Resampling here refers to creating samples (S1 and S2) of the same size as the original samples to simulate H0. Resampling is similar to bootstrapping but not exactly the same as we don't sample with replacement in resampling for hypothesis testing.

3. In resampling for hypothesis testing, we sample without replacement to ensure that each of the original points belongs to exactly one subset: S1 or S2.

4. Nowadays, computer simulations are very popular for most statistical tests as they can be made to work with very few assumptions about the data. All we need is a decent amount of data and some code.

5. That's true. We can use CLT and come to conclusions using sampling distributions

6. Each of these tests has been designed in a pre-computing world where simulations were very hard to perform. Most of the tests like t-test make assumptions about the data and hence are limited to only certain types of data. With computer simulations, we can perform most of the tests these older methods perform using a computer and resampling.

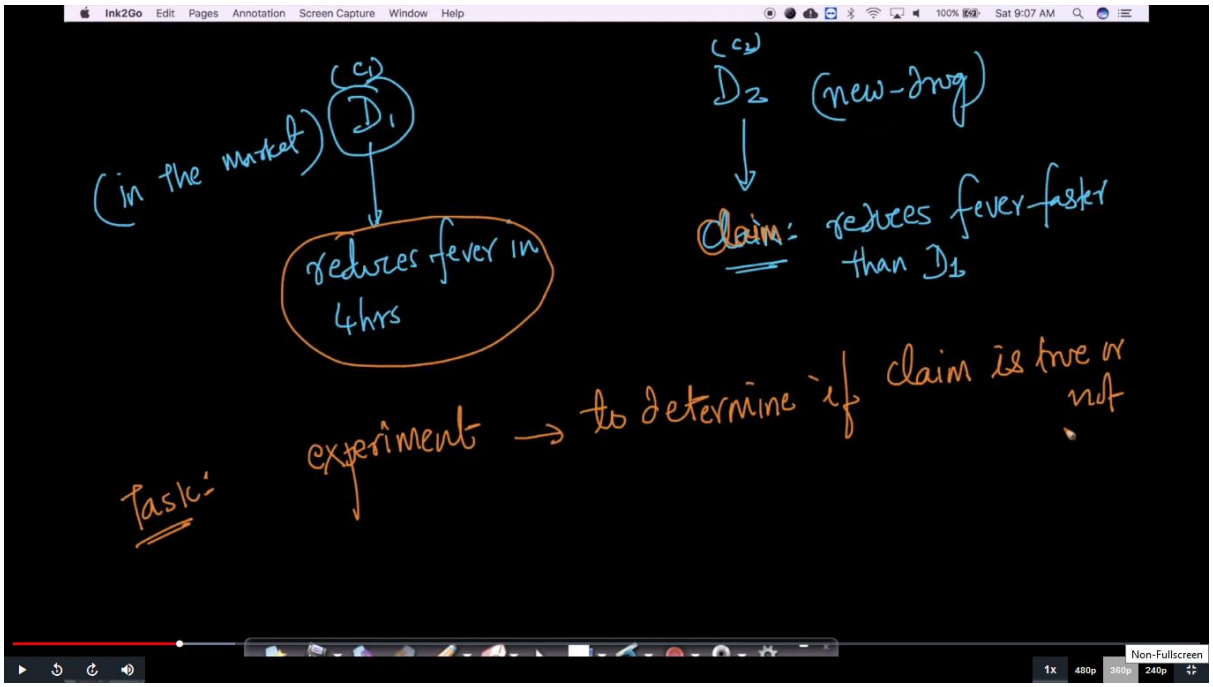
Hypothesis Testing

1. Most frequently asked questions about this topic: https://www.youtube.com/watch?v=2VE6AZPS6bI

2. How we setup hyp-testing as a proof by contradiction: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hypothesis-testing-testing-methodology-null-hypothesis-p-value/

3. Coin-toss example: https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hypothesis-testing-intution-with-coin-toss-example/

How to use Hyp testing



suppose alternate hypothesis is H1: drug D2 is better than D1 then D1-D2 returns a positive value while D2-D1 returns a negative value. so if we get a positive value and if the test statistic is D1-D2 then we can say H1 is D2 is better than D1

1. If drug D2 is better than D1:

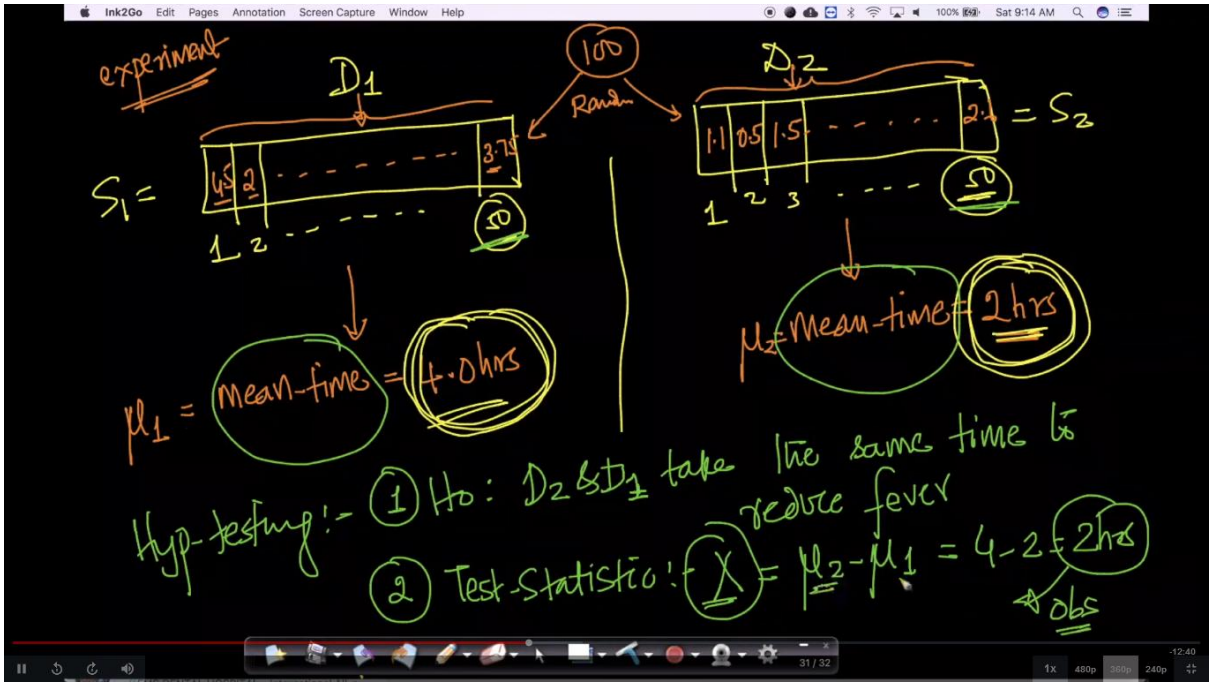
Test statistic --> u1-u2

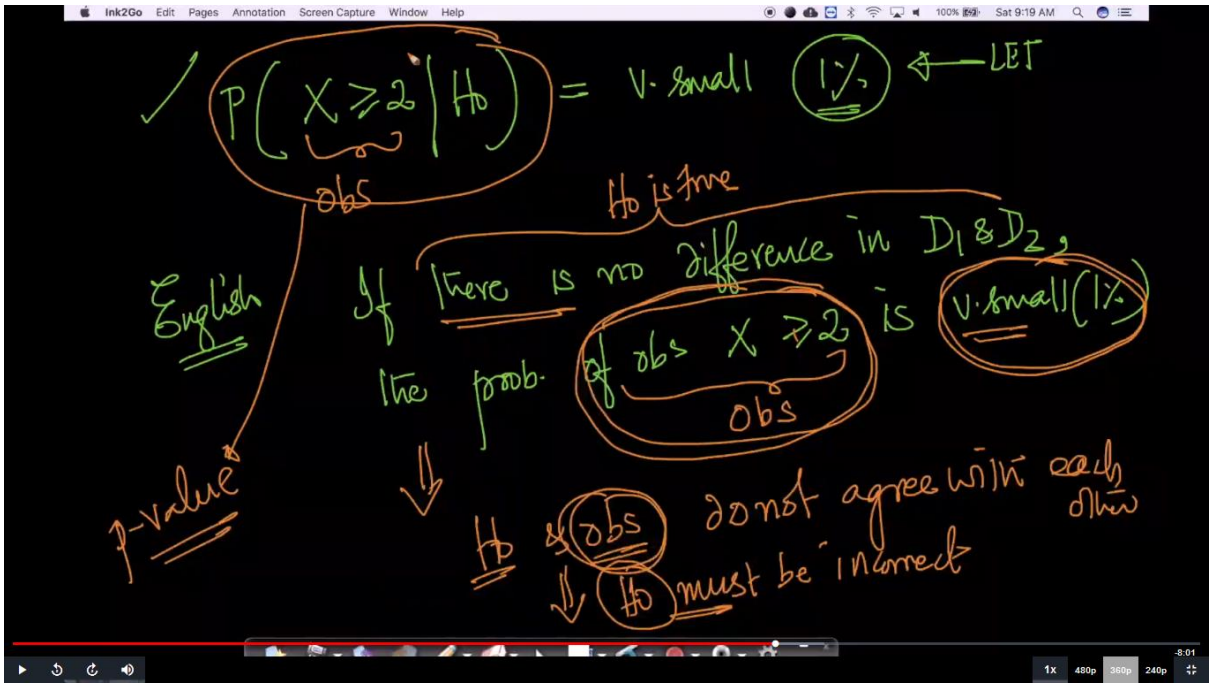
2.If D1 is better than D2:

--> u2-u1

3. or just say there is a difference between two drugs:

--> |u2-u1|





Questions:

1a.Its a very basic question:we only have 100 samples and we resample it again and again only from this 100 samples and we perform hypothesis testing and accept or reject it.My doubt is like,are these resampled samples sufficient enough to support/neglect such a big claim?

1b.May be we can increase the sample size but how much do we set for a sample size ?say for the drug example,the population is almost all(anybody can be the

patient for a normal fever medicine).In such a case how much do we set for sample size ?

1c.In general case like e-commerce what would be the appropriate sample sizes?

Ans 1a,1b - It is always beneficial to have more observations to start with. But, given only 100 samples, this is the best we can do. In some real-world critical hypothesis testing like in medicine, it is mandatory to have a minimum sample size of a few hundred to thousands. Regulators like FDA in the US have strict guidelines for sample sizes based on the severity of the illness that a medicine/drug would treat.

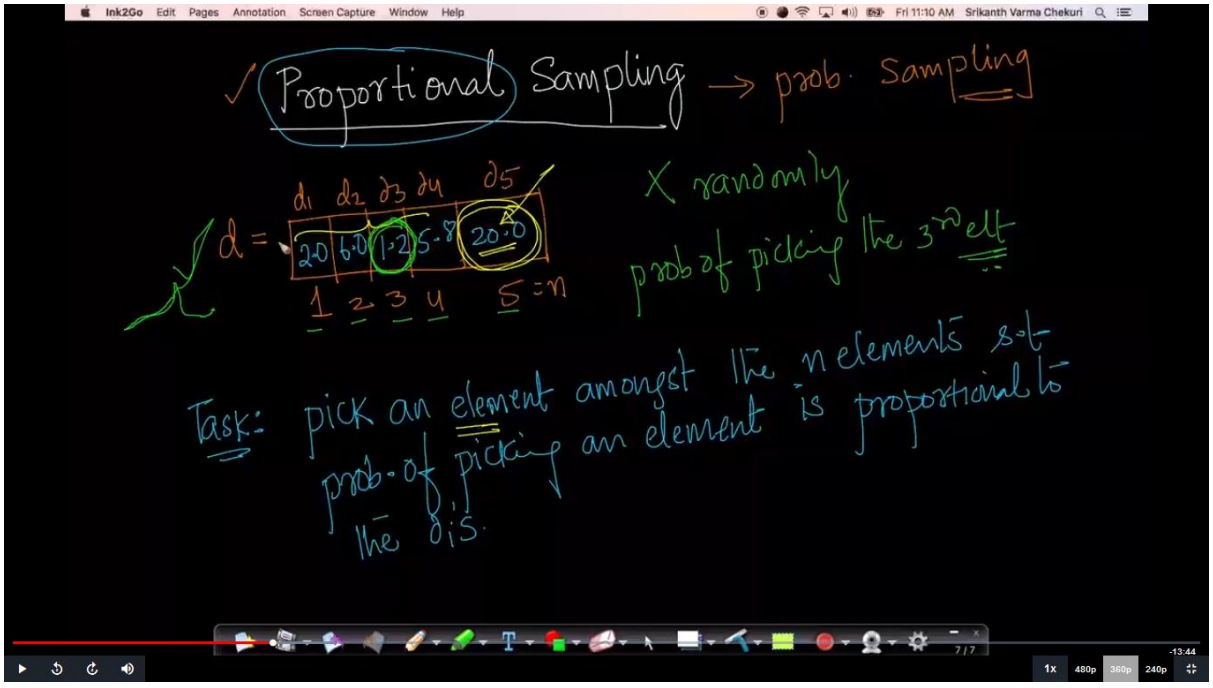
1c. In e-commerce since there is tons of data collected, it is easier to obtain very large sample sizes in hundreds of thousands and even millions.

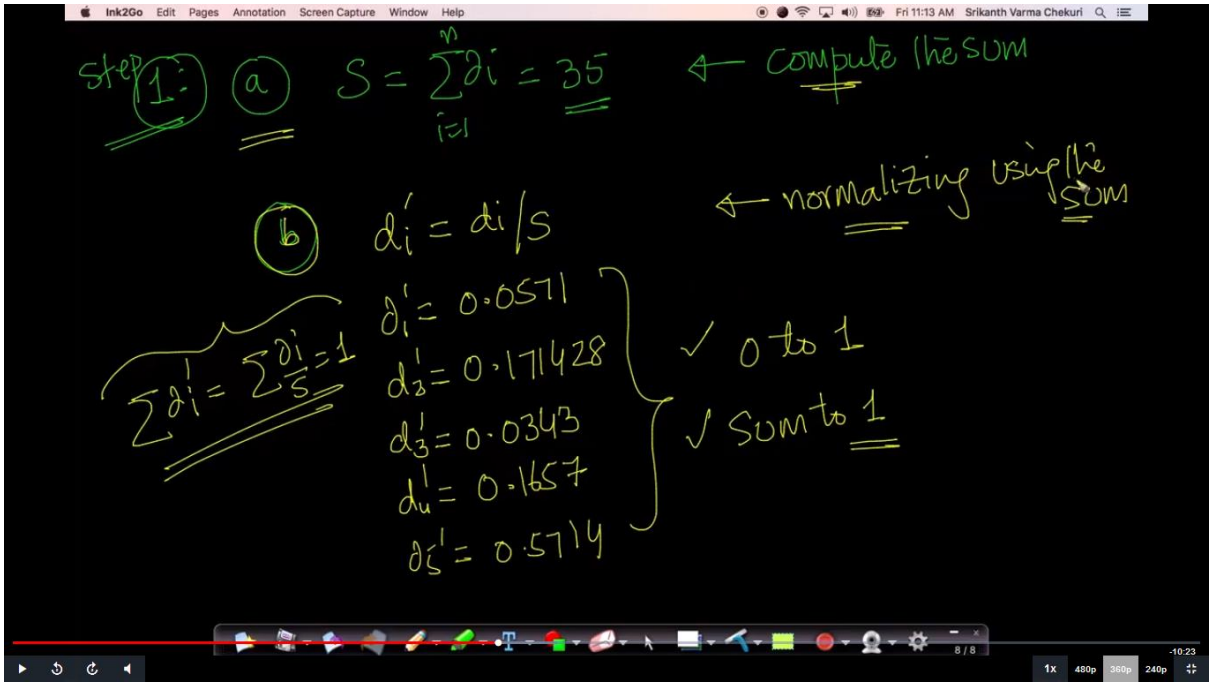
====

p value varying wildly means at design stage of experiment itself we have to carefully chose number of trails if required perform experiments to choose right number of trails for that particular experiment

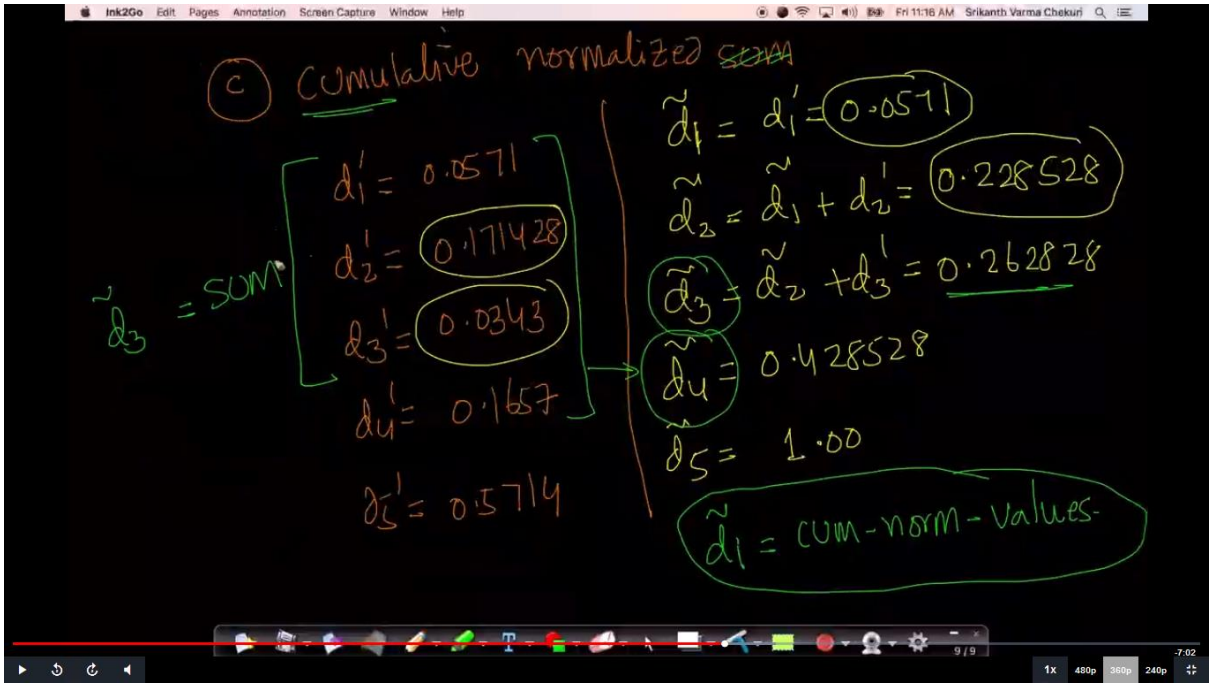
==

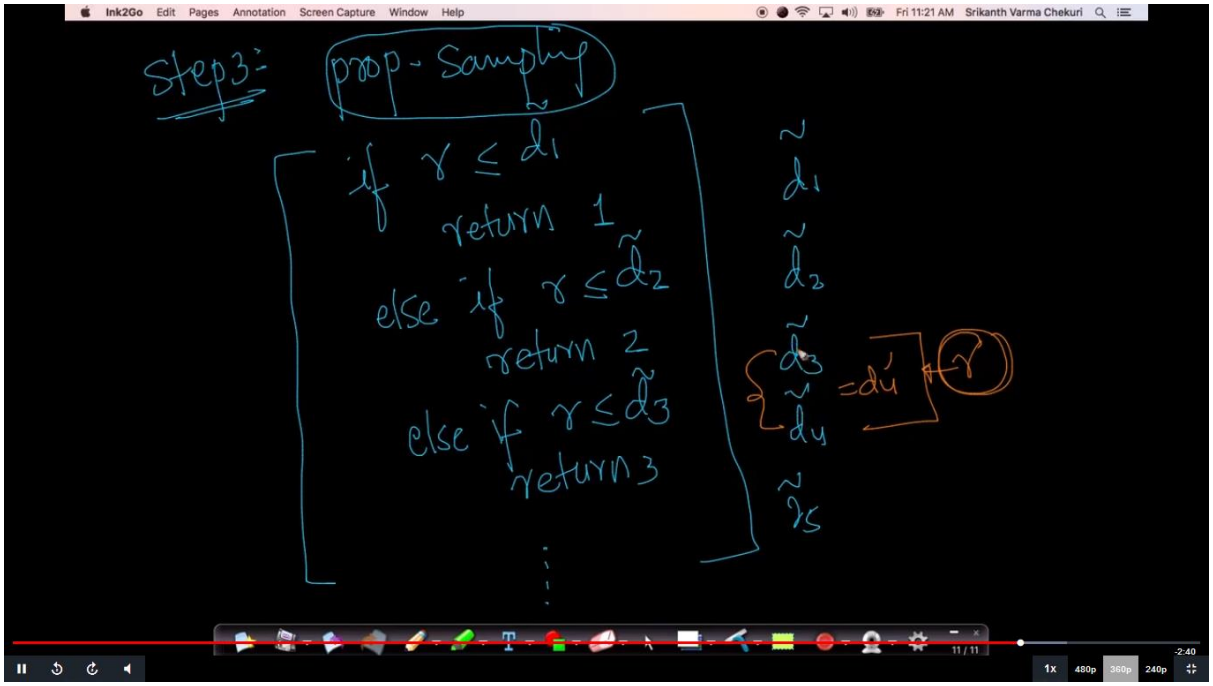
Proportional sampling





tildas can be found using cumulative sum from the di'. These values show probability of a value being picked up.





1. As discussed in video , the probability of getting 4th element is proportional to d4' (d4' = d4/S) which is proportional to d4 that's what our goal to start with proportional sampling.

2."Why we are doing r = U(0,1) ?" -Because by uniformly sampling between 0 and 1 , the probability of drawing point is same.

we are mainly talking about the gaps (di') and the higher the gap the more is the chance of that value being selected. Also the main focus here is the magnitude of the each value in the list, since the probability of a value being selected is calculated based on its magnitude. Here in the original list [2.0, 6.0, 1.2, 5.8, 20.0] we already got the gaps as:

d1' = 0.0571

d2' = 0.1714

d3' = 0.0343

d4' = 0.1657

d5' = 0.5714

This is the main thing to figure out the probability of getting each value.

P(d1) = 5.71%

P(d2) = 17.14%

P(d3) = 3.43%

P(d4) = 16.57%

P(d5) = 57.14%

So the normalized values using sum of all the values are enough to tell us the probability of that value being selected, i.e. step 1(a) and 1(b). Step 2 and 3 are to validate these probabilities. Hence for 57.14% values of r between 0.0 and 1.0 we will correctly get d5.

Coolest thing about this is since all these calculation are based on the magnitude of values in the list so the probability difference factors are proportional to value difference factors i.e.

d1 = 2.0,

d2 = 6.0 = (3) * d1

P(d1) = 5.714285714285714%

P(d2) = 17.14285714285714% = 3 * P(d1)

This applies for all values.

=======

http://methods.sagepub.com/Reference//encyc-of-research-design/n340.xml#:~:text=Proportional%20sampling%20is%20a%20method,sampling%20techniques%20to%20each%20subpopulation.

====

Revision Questions:

1. What is PDF?(https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2842/gaussiannormal-distribution-and-its-pdfprobability-density-function/2/module-2-data-science-exploratory-data-analysis-and-data-visualization)

2. What is CDF?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2843/cdfcumulative-distribution-function-of-gaussiannormal-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

3. explain about 1-std-dev, 2-std-dev, 3-std-dev range?

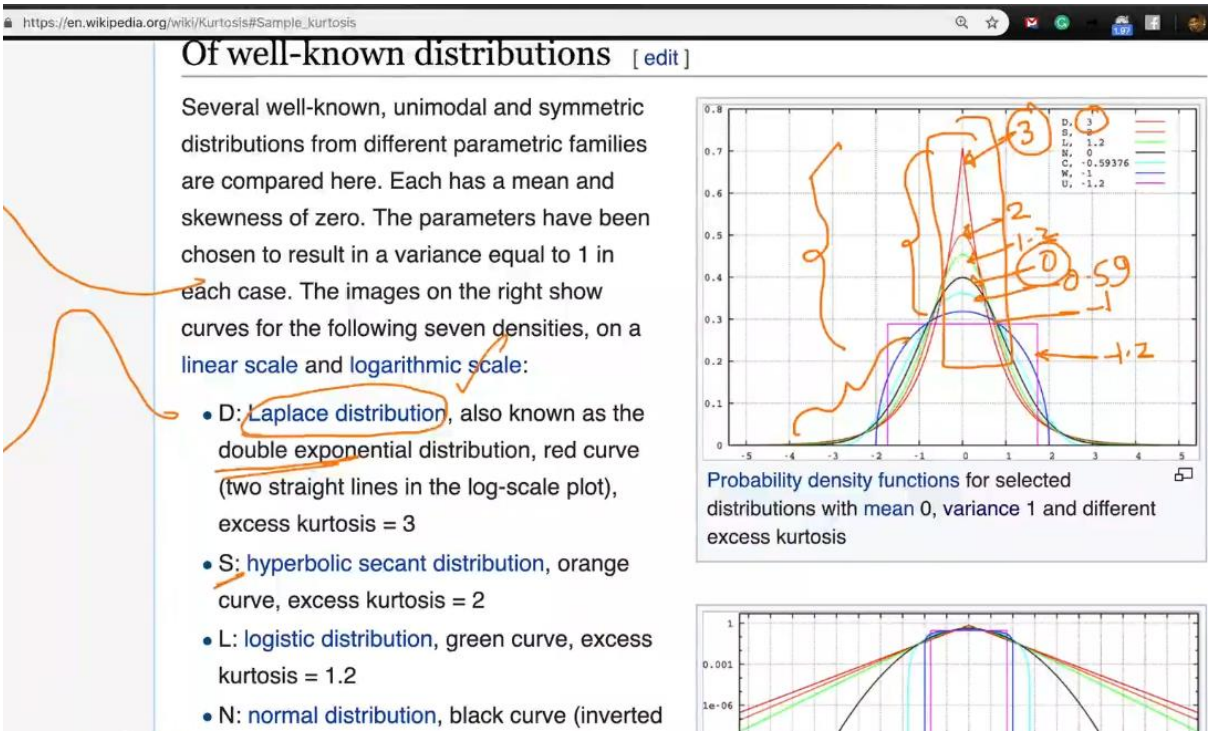
4. What is Symmetric distribution, Skewness and Kurtosis?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2844/symmetric-distribution-skewness-and-kurtosis/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

   Gaussian has Kurtosis = 3

   Excess Kurtosis = Kurtosis - 3 = How diff is the shape of distribution from Gaussian dist whose value is 3.

   It is a measure of Tailedness and not peakedness.

   Applications : Outliers in data set. Larger Kurtosis, more serious problems in dataset.



5. How to do Standard normal variate (z) and standardization?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2845/standard-normal-variate-z-and-standardization/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

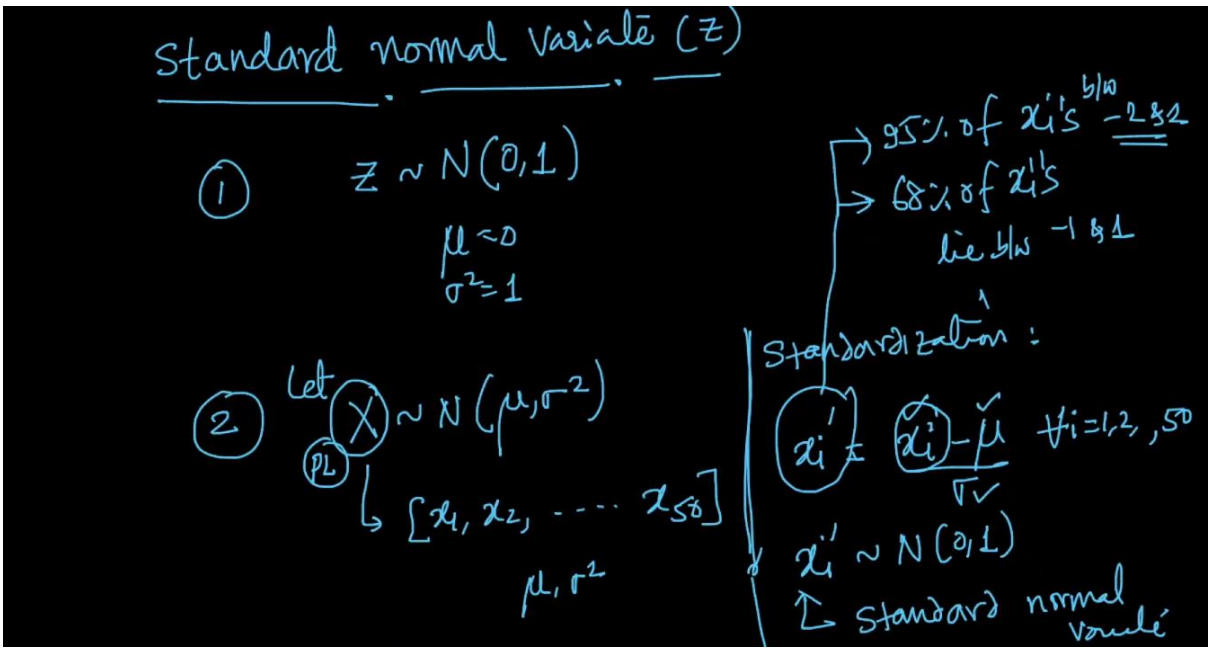
   X ~ N(mu,variance)

   say X has 50 observations [x1,x2,x3,.....,x50]

can apply transformation on each xi → xi' = (xi-mu)/s.d. sigma ⇒ subtract mean and divide by s.d. then can say that xi' ~ N(0,1) is gaussian distribution

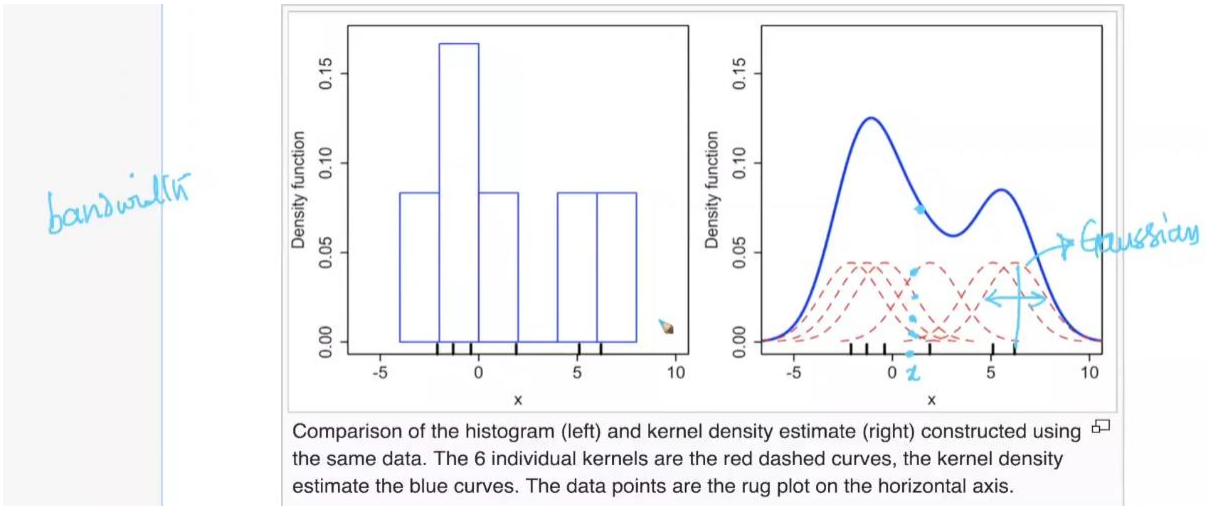
Why standardization?

It ensures that the resulting distribution after normalizing is Gaussian and Gaussian has properties of 68-95-97 rule, by which can surely say that 68.2% values lie between -sigma to +sigma and 95% values lie between [-2 sigma,+2 sigma] and 97% values lie between [-3sigma,+3sigma]



6. What is Kernel density estimation?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2846/kernel-density-estimation/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

   KDE - Kernel Density Estimation is used for smoothening Histogram and to get PDF from Histogram.



Comparison of the histogram (left) and kernel density estimate (right) constructed using the same data. The 6 individual kernels are the red dashed curves, the kernel density estimate the blue curves. The data points are the rug plot on the horizontal axis.

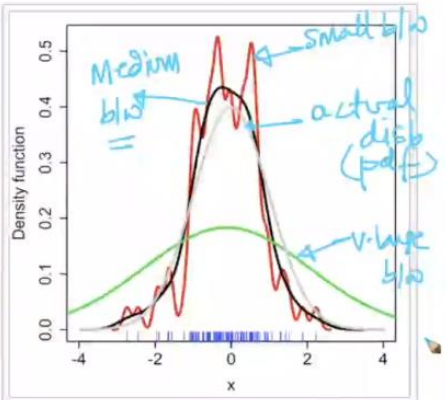
As the no of bins increases, at one point u will find a lot of kernel densities, as many points will be proximity to neighboring kernels and hence the height of the curve will be more. Variance in the case of kernels is called Bandwidth.

If I choose kernels which are very wide, I will get a green line curve - very large bandwidth. If we choose narrow kernels, we get a red curve (jagged).

Bandwidth selection [edit]

The bandwidth of the kernel is a free parameter which exhibits a strong influence on the resulting estimate. To illustrate its effect, we take a simulated random sample from the standard normal distribution (plotted at the blue spikes in the rug plot on the horizontal axis). The grey curve is the true density (a normal density with mean 0 and variance 1). In comparison, the red curve is *undersmoothed* since it contains too many spurious data artifacts arising from using a bandwidth $h = 0.05$, which is too small. The green curve is *oversmoothed* since using the bandwidth $h = 2$ obscures much of the underlying structure. The black curve with a bandwidth of $h = 0.337$ is considered to be optimally smoothed since its density estimate is close to the true density.

The most common optimality criterion used to select this parameter is the expected L_2 risk function, also termed the



Kernel density estimate (KDE) with different bandwidths of a random sample of 100 points from a standard normal distribution. Grey: true density (standard normal). Red: KDE with h=0.05. Black: KDE with h=0.337. Green: KDE with h=2.

7. Importance of Sampling distribution & Central Limit theoremhttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2847/sampling-distribution-central-limit-theorem/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
8. Importance of Q-Q Plot: Is a given random variable Gaussian distributed?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2848/q-q-plothow-to-test-if-a-random-variable-is-

normally-distributed-or-not/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

9. What is Uniform Distribution and random number generatorshttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2850/how-to-randomly-sample-data-points-uniform-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
10. What Discrete and Continuous Uniform distributions?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2849/discrete-and-continuous-uniform-distributions/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
11. How to randomly sample data points?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2850/how-to-randomly-sample-data-points-uniform-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
12. Explain about Bernoulli and Binomial distribution?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2851/bernoulli-and-binomial-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
13. What is Log-normal and power law distribution?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2852/log-normal-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

14. What is Power-law & Pareto distributions: PDF, exampleshttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2853/power-law-distribution/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
15. Explain about Box-Cox/Power transform?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2854/box-cox-transform/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
16. What is Co-variance?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2855/co-variance/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
17. Importance of Pearson Correlation Coefficient?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2856/pearson-correlation-coefficient/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
18. Importance Spearman Rank Correlation Coefficient?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2857/spearman-rank-correlation-coefficient/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

19. Correlation vs Causation?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2858/correlation-vs-causation/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
20. What is Confidence Intervals?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2859/confidence-interval-ci-introduction/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
21. Confidence Interval vs Point estimate?
22. Explain about Hypothesis testing?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2863/hypothesis-testing-methodology-null-hypothesis-p-value/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
23. Define Hypothesis Testing methodology, Null-hypothesis, test-statistic, p-value?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2863/hypothesis-testing-methodology-null-hypothesis-p-value/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
24. How to do K-S Test for similarity of two distributions?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2865/k-s-test-for-similarity-of-two-distributions/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

INTERVIEW QUESTIONS

1. What is a random variable?
2. What are the conditions for a function to be a probability mass function?(http://www.statisticshowto.com/probability-mass-function-pmf/)
3. What are the conditions for a function to be a probability density function ?(Covered in our videos)
4. What is conditional probability?
5. State the Chain rule of conditional probabilities?(https://en.wikipedia.org/wiki/Chain_rule_(probability))
6. What are the conditions for independence and conditional independence of two random variables?(https://math.stackexchange.com/questions/22407/independence-and-conditional-independence-between-random-variables)
7. What are expectation, variance and covariance?(Covered in our videos)
8. Compare covariance and independence?(https://stats.stackexchange.com/questions/12842/covariance-and-independence)
9. What is the covariance for a vector of random variables?(https://math.stackexchange.com/questions/2697376/find-the-covariance-matrix-of-a-vector-of-random-variables)
10. What is a Bernoulli distribution?

11. What is a normal distribution?
12. What is the central limit theorem?
13. Write the formula for Bayes rule?
14. If two random variables are related in a deterministic way, how are the PDFs related?
15. What is Kullback-Leibler (KL) divergence?
16. Can KL divergence be used as a distance measure?
17. What is Bayes' Theorem? How is it useful in a machine learning context?
18. Why is "Naive" Bayes naive?
19. What's a Fourier transform?
20. What is the difference between covariance and correlation?
21. Is it possible capture the correlation between continuous and categorical variable? If yes, how?
22. What is the Box-Cox transformation used for?
23. What does P-value signify about the statistical data?
24. 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?
25. How you can make data normal using Box-Cox transformation?
26. Explain about the box cox transformation in regression models.
27. What is the difference between skewed and uniform distribution?
28. What do you understand by Hypothesis in the content of Machine Learning?
29. How will you find the correlation between a categorical variable and a continuous variable ?
30. How to sample from a Normal Distribution with known mean and variance?

==========

NORMALIZAION: Data normalization is the process of rescaling one or more features to the range of 0 to 1. This means that the largest value for each feature is 1 and the smallest value is 0. Normalization is a good technique to use when you do not know the distribution of your data or when you know the distribution is not Gaussian (a bell curve). Normalization is useful when your data has varying scales and the algorithm you are using does not make assumptions about the distribution of your data

Standardization: Data standardization is the process of rescaling one or more features so that they have a mean value of 0 and a standard deviation of 1. Standardization assumes that your data has a Gaussian (bell curve) distribution. This does not strictly have to be true, but the technique is more effective if your feature distribution is Gaussian. Standardization is useful when your data has varying scales and the algorithm you are using does make assumptions about your data having a Gaussian distribution.

Column standardization is more effective when the underlying distribution is Gaussian with say mean=mu and std-dev=sigma. This is due to the fact that when you normalize such a feature, you get the new feature to be of N(0,1) distribution. This is an ideal distribution with lots of ML, Stats and Optimization techniques

assuming a Gaussian distribution to make the proofs work beautifully. You will be able to appreciate it better when you learn logistic regression and it's probabilistic interpretation (Gaussian Naive Bayes) later in this course.

That doesn't mean standardization is not good for no-gaussian distributed features. It still results in a new feature with a mean of 0 and variance of 1, but not N(0,1) distribution. In practice, we perform standardization irrespective of the underlying feature distribution. But the mathematical proofs are well suited when the distribution in Gaussian. That's all.

If the features follow different scaling, then it would be difficult for the model to converge faster and the computation time of the training increases and also it doesn't yield better results.

==========

Co-variance of a Data Matrix

>> This is also termed as Bessel's correction. Think of it this way. When we calculate the sample standard deviation from a sample of n values, we are using the sample mean already calculated from that same sample of n values. The calculated sample mean has already "used up" one of the "degrees of freedom of variability"(which is the mean itself) that is available in the sample. Only n-1 degrees of freedom of variability are left for the calculation of the sample standard deviation.

>> Take a look at this as well: http://vortex.ihrc.fiu.edu/MET4570/members/Lectures/Lect05/m10divideby_nminus1.pdf.

1. Take a look at this: https://stats.stackexchange.com/questions/349323/formula-for-sample-covariance-bessels-correction.

2. The covariance of two variables x and y is COV(x,y) = sum((xi - xm)(yi - ym))/(n-1) -> where xm and ym are the means of the respective variables. Take a look at this: http://ci.columbia.edu/ci/premba_test/c0331/s7/s7_5.html.

sij= fi(transpose)*fj / n-1

This video gives a clear representation of what is explained above

https://www.youtube.com/watch?v=9ONRMymR2Eg

In this blog they have explained how it actually works,kindly go through it

https://lazyprogrammer.me/covariance-matrix-divide-by-n-or-n-1/

https://math.stackexchange.com/questions/61251/intuitive-explanation-of-bessels-correction

****

MNIST Dataset

https://colah.github.io/posts/2014-10-Visualizing-MNIST/ -- Colah Blog - very good resource - visualizing in high dimensional space using PCA and t-SNE

Mnist is a very small dataset with very small images. So here we are able to get 784 dimensions which we can use for dimensionality reduction and all. But as you pointed out real images can be of very high resolution and if we try to convert them into vectors it will be huge vector and will be well beyond the computational capability.. For those kind of huge images we use other techniques to recognize and extract features like CNN which you will learn further in the chapters. Here we are just using 28*28 dimensional image , which when converted to vector is 784 dimensions only.

Whenever the size of the displayed image is larger than the actual resolution of the image , the distortion caused can be avoided using interpolation (It generally that reduces the visual distortion caused by the fractional zoom). If set to none , no interpolation is performed.

****

PCA article --

https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues/140579#140579

PCA reduces the dimensionality of data by seeking linear projections that can maximize the global variance of the projected data points. PCA thus can preserve the

maximum amount of the global information of a data set. (PCA is just a diagonal rotation of our initial covariance matrix and the eigenvectors represent a new axial system in the space spanned by our original data. We can directly explain what a particular PCA does.)

Global structure means preserving the global shape between all the datapoints like in Principal Component Analysis that are great at retaining global structure, because it looks at ways in which a dataset's variance is retained, globally, across the entire dataset. In other words local approaches seek to map nearby points on the manifold to nearby points in the low-dimensional representation. Global approaches on the other hand attempt to preserve geometry at all scales, i.e mapping nearby points to nearby points and far away points to far away points

PCA tries to preserve the global structure whereas T-SNe preserves the local neighborhood. like when we perform PCA on data which is circular in shape(non linear) then it loose lot of information when projected to lower dimensions. it also loose its circular shape while if we project linear shaped data to lower dimensions it tries to hold its original structure.

do we always apply PCA or t-SNE to dataset before applying any machine learning algorithm to dataset ?

Yes, we do apply PCA/T-SNE on a dataset before applying any ML algorithm. We apply PCA/T-SNE to reduce the dimensionality of the dataset either through preserving maximum variance or maximum neighborhood. After reducing the dimensionality, then we do build any ML model using this new dataset.

We use either T-SNE or PCA when we have to reduce the dimensionality of the given data matrix.

If we have to reduce the dimensionality keeping the maximum neighborhood, then we go for T-SNE.

If we have to reduce the dimensionality keeping the maximum variance, then we go for PCA.

Is t-SNE used now also??

still TSNE is the most widely used dimensionality reduction technique. We can also use Auto encoders for dimensionality reduction and this will be taught in the deep learning section in detail. For visualization, TSNE is preferred and for dimensionality reduction Auto encoders are preferred.

https://distill.pub/2016/misread-tsne/

Audio reply: https://soundcloud.com/applied-ai-course/tsne-in-modeling 1. Yes, t-SNE is stochastic in nature and hence the final embedding may change even with the same parameters. 2. At each iteration, we are trying to change the embedding points slightly in such a way that the newly embedded points better preserve the neighborhood than the previous configuration by solving an optimization problem. You will understand how optimizåtion problems are iteratively solved in the "solving optimization problems" chapter.

The perplexity is basically the effective number of neighbors for any point, and t-SNE works relatively well for any value between 5 and 50. Therefore, Larger perplexities will take more global structure into account, whereas smaller perplexities will make the embeddings more locally focused. \

Revision:

1. What is dimensionality reduction? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2878/what-is-dimensionality-reduction/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
2.

3. Explain Principal Component Analysis?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2889/geometric-intuition-of-pca/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
4.
5. Importance of PCA?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2888/why-learn-pca/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
6.
7. Limitations of PCA?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2894/limitations-of-pca/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
8.
9. What is t-SNE?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2898/what-is-t-sne/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
10.
11. What is Crowding problem?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2901/crowding-problem/2/module-2-data-science-exploratory-data-analysis-and-data-visualization
12.
13. How to apply t-SNE and interpret its output?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2902/how-to-apply-t-sne-and-interpret-its-output/2/module-2-data-science-exploratory-data-analysis-and-data-visualization

Interview Questions

1. 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.)(https://www.analyticsvidhya.com/blog/2016/09/40-interview-questions-asked-at-startups-in-machine-learning-data-science/)
2. Is rotation necessary in PCA? If yes, Why? https://google-interview-hacks.blogspot.com/2017/04/is-rotation-necessary-in-pca-if-yes-why.html
3. 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?(https://www.linkedin.com/pulse/questions-machine-learning-statistics-can-you-answer-saraswat/)

https://www.analyticsvidhya.com/blog/2017/03/questions-dimensionality-reduction-data-scientist/?utm_medium=social&utm_source=linkedin.com&utm_campaign=buffer

Suppose we are using dimensionality reduction as pre-processing technique, i.e, instead of using all the features, we reduce the data to k dimensions with PCA. And then use these PCA projections as our features. Which of the following statement is correct?

A. Higher ‘k’ means more regularization

B. Higher ‘k’ means less regularization

C. Can’t Say

Solution: (B)

Higher k would lead to less smoothening as we would be able to preserve more characteristics in data, hence less regularization.

I am not able to understand this question. What is regularization and how is it related to PCA ?

Audio reply: https://soundcloud.com/applied-ai-course/comment-pca-regularization/s-srZV3

Overfitting means the model performs very well on the training data but it is unable to perform well on unseen data. Regularization is a technique to reduce overfitting.

Please go through the remaining videos. You will understand it better after studying Logistic regression videos.

AMAZON FINE FOOD REVIEW DATA

While data cleaning, I found that book review data(approx 359 records) is mixed up with food review data.

so I came up with following code to delete such rows,

```
#########################################################################

def apply_mask_summary(filtered_data,regex_string):

mask = filtered_data.Summary.str.lower().str.contains(regex_string)

filtered_data.drop(filtered_data[mask].index, inplace=True)


def apply_mask_text(filtered_data,regex_string):

mask = filtered_data.Text.str.lower().str.contains(regex_string)

filtered_data.drop(filtered_data[mask].index, inplace=True)
```

apply_mask_summary(filtered_data,re.compile(r"\bbook\b"))

apply_mask_summary(filtered_data,re.compile(r"\bread\b"))

apply_mask_text(filtered_data,re.compile(r"\bbook\b"))

apply_mask_text(filtered_data,re.compile(r"\bread\b"))

apply_mask_summary(filtered_data,re.compile(r"\bbooks\b"))

apply_mask_summary(filtered_data,re.compile(r"\breads\b"))

apply_mask_text(filtered_data,re.compile(r"\bbooks\b")) //regex used bcs some words that are like 'books', 'booksandpens' etc.

apply_mask_text(filtered_data,re.compile(r"\breads\b"))

apply_mask_summary(filtered_data,re.compile(r"\breading\b"))

apply_mask_text(filtered_data,re.compile(r"\breading\b"))

##

Before removing book review data - (525814, 10)

After removing book review data - (506630, 10)

duplicate data is not removed so this number will further reduce.

we want reviews related to food items, not books. So, we can remove others.

****

Lemmatization vs Stemming

https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html

Stemming - crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes removal of derivational affixes.

***Lemmatization* usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the *lemma* . If confronted with the token *saw*, stemming might return just *s*, whereas lemmatization would attempt to return either *see* or *saw* depending on whether the use of the token was as a verb or a noun.**

Lemmatisation is the algorithmic process of determining the lemma of a word based on its intended meaning. Unlike stemming, lemmatization depends on correctly identifying the intended part of speech and meaning of a word in a sentence, as well as within the larger context surrounding that sentence, such as neighboring sentences or even an entire document.

Please go through the links for more details:

1. https://blog.bitext.com/what-is-the-difference-between-stemming-and-lemmatization /

2. https://stackoverflow.com/questions/1787110/what-is-the-true-difference-between-lemmatization-vs-stemming

In order to create the BoW firstly, we are required to remove the stop words. Then we process our Stemming process. Stemming is the algorithmic process where each words having the same sense of understanding are converted to its stem or root. Here root can be a word in itself or not. Example:

study , studying , studies all stemmed to studi (not study). Each word can be formed by adding the suffixes to the root.

We also perform Lemmatisation which is the algorithmic process of combining the inflected words into a common word called lemma. The Algorithm make use of the dictionary to guess down the lemma . Suppose we have words = {go , going , went} we get lemma as "go".

If you want to know how nltk does it let me give u a rough idea of it.

NLTK makes use of Parts of speech - Tagging , where parts of speech tagger take group of words as an input and it returns list of tuples. Tuple contains a word and corresponding parts of speech.This parts of speech tagging is done to make sense of context in the sentence and helps lemmatizer to choose the appropriate lemma.

Stemming is the process of reducing a word into its stem, i.e. its root form. The root form is not necessarily a word by itself, but it can be used to generate words by concatenating the right suffix.

For example, the words fish, fishes and fishing all stem into fish, which is a correct word. On the other side, the words study, studies and studying stems into studi and the words like cries and cry stems into cri which is not an English word.

but lemmatizer provides different lemma for both tokens study for studies and studying for studying. So when we need to make feature set to train machine, it would be great if lemmatization is preferred.

CountVectorizer() - Creates a vector where each index i.e.word contains value corresponding to the no of times it occurs in the document.

TfIdfVectorizer() - creates a vector where each index i.e.word contains value corresponding to the tfidf value of that word in the corpus.

To capture sequence information we use n_grams. For example, let us consider a review "This Cake is not tasty." If we take unigrams, most probably we get positive polarity for this review since the presence of positive word "tasty". So to get the correct prediction from the model we need to use sequence information in the review, which is "not tasty" by using bi_grams we can achieve this feature. In the same way, Unigrams features also useful for model performance. Since uni-grams capture, the complete information in the review whereas n-grams captures only sequence information.

==========

```
tf_idf_vect = TfidfVectorizer(ngram_range=(1,2), min_df=10)

tf_idf_vect.fit(preprocessed_reviews)

print("some sample features(unique words in the
corpus)",tf_idf_vect.get_feature_names()[0:10])

print('='*50)

final_tf_idf = tf_idf_vect.transform(preprocessed_reviews)
```

Here 'df' denotes document frequency and it generally takes float values.

Here we have 'min_df' and 'max_df' arguments

If min_df = 0.4, then it means it takes only those words that are present in minimum 40% of the reviews in the corpus.

If max_df = 0.8, then it means it takes only those words that are present in maximum 80% of the reviews in the corpus.

The sparse matrix that we obtained here (final_tf_idf) stores only the non zero values for memory efficiency.We don't need to explicitly apply any technique here. Internally tfidfvectorizer outputs the matrix in scipy sparse representation.

The shape attribute of a sparse matrix is exactly the same as the shape attribute of a dense (i.e. generic) matrix. Internally, sparse matrix stores only non-zero values.

TFIDF transformer vs TFIDF vectorizer

It starts with a corpus of raw texts. In order to get a sparse matrix of TF-IDF values,

1) First we need to tokenize

2) Count the tokens

3) Transform the raw counts to TF_IDF values

TFIDF vectorizer performs all these 3 steps whereas TFIDF transformer performs only Steps 2 and 3.

TfidfTransformer - Transform a count matrix to a normalized tf or tf-idf representation

TfidfVectorizer - Transform a collection of raw documents to a matrix of TF-IDF features.

i)What to do if we have class imbalance problem in NLP dataset.?

ii)how to solve class imbalance problem, in text data.? which technquie we can apply?

Please refer this blog : https://towardsdatascience.com/how-i-handled-imbalanced-text-data-ba9b757ab1d8#:~:text=The%20simplest%20way%20to%20fix,synthetic%20instances%20from%20minority%20class

We can go for oversampling of the text, undersampling of a text.

SMOTE creates synthetic samples, creating synthetic samples may be wrong in terms of the text so don't prefer it.

Another way is data augmentation. replace words by synonyms of the words for the imbalance class and get more data of that.

in Word2Vec, we are creating a vector for each word. Whereas in TF-IDF, we are creating a vector for each data point. The' mincount' in Word2Vec model indicates that the vecotrs have to be created oony for those words that have occurred >= 'mincount' number of times in the corpus.

Here those words for which the vectors have been created in Word2Vec model, we are using only the vectors of those words present in TF-IDF vectorizer as features and have vectors created in Word2Vec modell

---

TF-IDF and Wighted Word 2 vec for sentence conversion to vectors.

Audio reply: https://soundcloud.com/applied-ai-course/word2vec-intuition

Refer: https://projector.tensorflow.org/

Word2Vec code sample:https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/avg-word2vec-and-tfidf-word2vec-code-sample/

==========

CLASSIFICATION AND REGRESSION

Converting the text into a vector form comes under featurization of the data. You first have to perform all the text pre-processing like removal of punctuations, conversion to lowercase, stopword removal and then you need to convert the processed text into vector form using any one of the vectorization techniques like BOW/TF-IDF/Avg Word2Vec/TF-IDF Weighted Avg Word2Vec.

2) Now you have to perform feature scaling and then build a model like KNN or Logistic Regression or SVM, etc. This where classification happens through this model.

fit() and transform() methods are used in vectorization.

In vectorization, you first need to apply fit() on the train data, so that it creates vector with the features present in the train data and transform() has to be applied on both train and test data, in order to compute the scores for both train and test data.

In model building, we apply fit() on the train data, so that it trains the model using train data and then make predictions on the test data using predict().

3) Vectorization techniques like BOW/TF-IDF/Word2Vec do not need the labels as they are converting the text into the vector format, but not making any predictions on the data. Labels are to be passed through ML models during training phase, as the ML model needs to compute the hyperparameters and get trained.

In Vectorization phase, fit() method creates the features out of the corpus given as an input. transform() assigns the count/binary values or the TF-IDF scores to each of these components for every data points and finally the vector is obtained.

In training phase, the model we do not have transform() for ML models. We only have fit() and predict().

Here fit() learns the patterns in the given training dataset and obtains the final function that maps the input to the output.

Regression vs Classification

If the target variable consists of a set of unique values(Discrete RV - finite set of values) then it is a classification problem. If the target variable domain belongs to a range of real values(Continuous RV) - infinite set of values then it is a Regression problem.

there are many algorithms like decision trees, random forests which can perform classification as well as regression.

L1 norm L2 norm

https://medium.com/@montjoile/l0-norm-l1-norm-l2-norm-l-infinity-norm-7a7d18a4f40c

The L0 distance between (1,1) and (2,2) is 2 because neither dimension matches up. Imagine if the first and second dimensions represent username and password, respectively. If the L0 distance between a login attempt and the true credentials is 0, then the login is successful. If the distance is 1, then either the username or password is incorrect, but not both. Lastly, if the distance is 2, both username and password are not found in the database. Please refer here.

L0 norm means the total number of non-zero elements in a vector.

https://machinelearningmastery.com/vector-norms-machine-learning/

==========

We donot use **Crossvalidation** in real time. See we use CV for hyperparameter tuning to get the best parameter. After we get the best parameter of the model, we train the model on whole data using that data and in test time we just calculate the output based on that model. In the case of Knn, we find the best k using CV, then at testing time we use that best 'K' to test for the test query using the whole train dataset.

1) In training phase, all the data points are stored in the RAM. The train error is computed by making predictions on the train data again by computing the nearest neighbors from the train data.

2) The decision surface is a line/curve that separates the data points based on the predicted values. All we need to do it just draw a line/curve separating majority of the points correctly.

You're not performing the training and testing phases on the same data. Initially if a dataset is given, you first have to divide it into train and test data matrices. Append the train data labels to the train data matrix(now it becomes new train data) and the test data labels to the train data matrix(now it becomes new test data). Create a new column in both new train and test data and fill all the train data column values with 1 and for test data it has to be filled with 0. Build a model and then check if you're able to differentiate the train data points from the test data points. If we are able to differentiate, then it means the data has been changed completely and you need to retrain the model by increasing the train dataset. If they both are not able to be separated, then it means the distribution of the data has not been changed.

The above mentioned procedure is to check if both the train and the test data belong to same distribution.

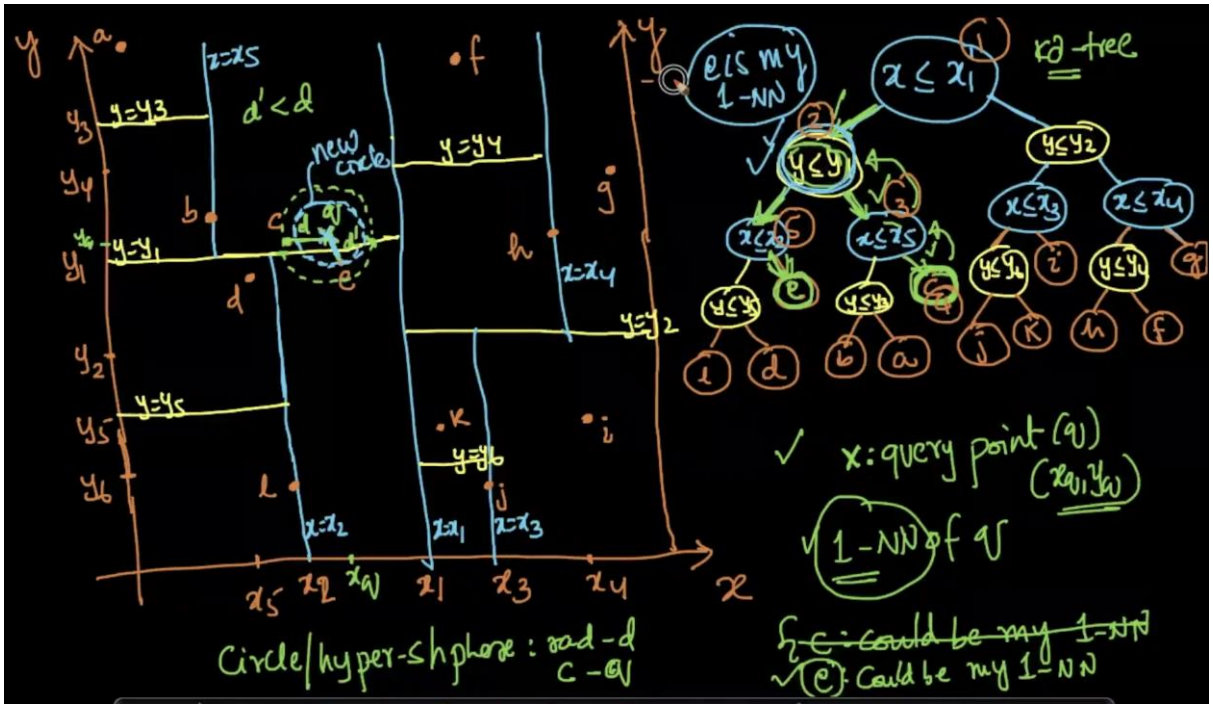
Regarding the train and test errors, you have to compute the train error on the train data and the test error on the test data using the same mode which you build using train data. (For diffferent values of 'K', you have to compute the train and cv errors)

==========

KD Tree is an extension to BST.

1. Pick x axis . Project points onto x-axis.Compute median. say call it x1. Split data using median.

2. Alternate between axis. first taken x then thake y.



LSH - Locality Sensitive Hashing

Steps for locality sensitive hashing:

1.First make ‘m’ hyperplanes to split into regions and create slices such that cluster of points lie in a particular slice and be called their neighbourhood.typically m = log(n)

2.Next for each point create a vector(also called hash function) by W1(transpose).point. if it is greater than 0 , it lies on the same side of that hyperplane else other side. Based on that create a vector of m size. For eg the vector can be [1,-1,-1] denoting point x lies on the same side of normal to hyperplane 1, opposite side to normal of hyperplane 2 and 3. Now this vector serves as a key to the hash table and all the points with the same key or vector representation will go in the same bucket as they have similar vector representation denoting they lie in the neighbourhood of each other.

3.Now it may happen that two points which are very close fall on different slices due to placing of hyperplane and hence not considered as nearest neighbour. To resolve this problem, create l hash tables. (l is typically small). In other words repeat step 2 l times thus creating l hash tables and m random planes l times. So when a query point comes, compute the hash function each of the ‘l’ times and get its neighbours from each bucket. Union them and find the nearest neighbours from the list. So basically in each ‘l’ iterations create m hyperplanes and hence region splitting will be different thus vector representation or hash function of the same query point will be different in each of the representations. Thus the hash table will be different as

points which lie on the same region in the previous iteration might lie in a different region in this iteration and vice versa due to different placement of hyperplanes.

Time complexity is O(m*d*l) for each query point. And for creating the hash table O(m*d*l*n) which is one time only.

Space complexity is O(n)

We can create our own hash function to hash the input values into buckets or we can use the predefined hash function we already have.

SKlearn implementation of LSH uses "GaussianRandomProjectionHash". you can check in this link.

Steps to compute projection of a point Xq on to the plane W(T)X=0 are correct-

1. first we will generate W using N(0,1)

2. Now as you explained in the session of linear Algebra that projection of vector a on to a vector b is d=a.b/||b||

where d is the distance of the projected point from origin

with the similar process we can compute projection of point Xq on to the Plane W(T)X=0 using below steps-

d=W.Xq/||W|| now as W is unit vector d= W.Xq

i.e d= W(T)*Xq

where d is nothing but the distance of projected point from origin

and W is the unit normal on plane W(T)X=0

LSH - Euc distance

vector w1 and you can compute the unit-vector in the same direction as w1 as w1/|| w1 || (or) w1/norm(w1)

break a plane into equi-sized buckets and compute the bucket index. This can be done as follows. Compute the projected point, then compute its distance from the origin to the project point and divide this distance by the bucket size to obtain the exact bucket-index.

Please understand that orthogonal vector is the way we represent a plane with. As plane can have infinite set of points, in order to define a plane, we need a orthogonal vector that is passing through the plane. A plane equation can be written as ax+by +cz=0 and here [a,b,c] are nothing but the orthogonal vector (Ex- 4x+2y+3z=0). For a point that is lying above this plane, ax+by+cz >0 , for a point lying below(below insense opposite to direction of normal vector) the plane ax+by+cz <0 and for all the

points that lie on this plane, ax+by+cz=0. Orthogonal vector is not a seperate component of plane. To represent a plane, we must need an orthogonal vector that is passing through this plane.

By bucket size, we mean the width of each segment/region on the plane. Yes, it is a predefined constant.

We can use the segment number/index to know which region of a plane a point belongs to.

Cell in hashtable is called a segment.

Each region consists of different number of planes passing through it so if we take only those number of planes into consideration, the chances for occurrence of duplicate keys are high.

Let us assume we are having 10 hyperplanes and for a given point 'x_1', which is in a region, we see 3 hyper planes passing through it. So here let the combination of the values associated with the hyperplane be +1,-1,-1 (Let's say associated with Planes P1,P4,P5)

Now we also have another point 'x_2' which is in a different region and it has planes P8,P9,P10 passing through it. So here let us assume we have the combination associated with the combination for this point is also +1,-1,-1. In this case, both 'x_1' and 'x_2' go into the same bucket though they're from different regions and having different set of hyper-planes passing through the regions in which they're present. Also the dictionary allows the key combination of +1,-1,-1 only once and these two points go into this bucket and gives us incorrect results. Hence we take all the hyper-planes into consideration while building the keys and taking ll the hyperplanes into consideration will categorize the points correctly into the buckets and hence we could get best results.

K-NN is used where??

It is often used in retail, as it is particularly sensitive to groupings. Items that are retuned / exchanged more frequently. Customer data entered manually more frequently that card swipes may indicated customer personal data theft. Fixed patterns of purchase are discernable say from a grocery store by customer demographics, ancillary product can go on sale to encourage purchases. This occured with middle age males visiting store on weekends, and store having beer sales on Thursday thru Sunday. Beer purchases increased.

In a particularly visible application, I believe Spotify is using it to make song recommendations: https://github.com/spotify/annoy

Revision Questions:

1. Explain about K-Nearest Neighbors?https://www.appliedaicourse.com/lecture/11/applied-machine-

learning-online-course/2927/k-nearest-neighbours-geometric-intuition-with-a-toy-example/3/module-3-foundations-of-natural-language-processing-and-machine-learning

K NN is a supervised algorithm (has labeled data for training the model) and is used for solving Regression and Classification problems. A **classification problem** has a discrete value as its output. For example, “likes pineapple on pizza” and “does not like pineapple on pizza” are discrete.

A **regression problem** has a real number (a number with a decimal point) as its output. For example, we could use the data to estimate someone’s weight given their height.

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.

The KNN Algorithm

1. Load the data
2. Initialize K to your chosen number of neighbors

3. For each example in the data
3.1 Calculate the distance between the query example and the current example from the data.
3.2 Add the distance and the index of the example to an ordered collection
4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances
5. Pick the first K entries from the sorted collection
6. Get the labels of the selected K entries
7. If regression, return the mean of the K labels
8. If classification, return the mode of the K labels

2. Failure cases of KNN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2928/failure-cases-of-knn/3/module-3-foundations-of-natural-language-processing-and-machine-learning

KNN's main disadvantage of becoming significantly slower as the volume of data increases makes it an impractical choice in environments where predictions need to be made rapidly.

Failure points -

1. If query point is far away from points in Dataset then not sure about the class of query point.
2. If +ve/-ve points are randomly spread i.e. jumbled then no useful info can be made.

An example of this is using the KNN algorithm in recommender systems, an application of KNN-search.

3. Define Distance measures: Euclidean(L2) , Manhattan(L1), Minkowski, Hamminghttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2929/distance-measures-euclideanl2-manhattanl1-minkowski-hamming/3/module-3-foundations-of-natural-language-processing-and-machine-learning

   Distance is between 2 points and Norm is for a vector from origin.

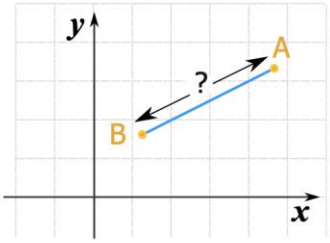
   Euclidean distance

Distance Between 2 Points

Quick Explanation

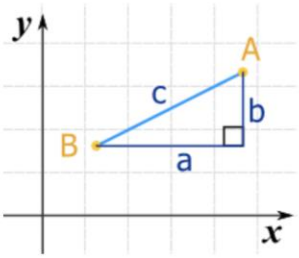
When we know the **horizontal** and **vertical** distances between two points we can calculate the straight line distance like this:

$$\text{distance} = \sqrt{a^2 + b^2}$$



Imagine you know the location of two points (A and B) like here.

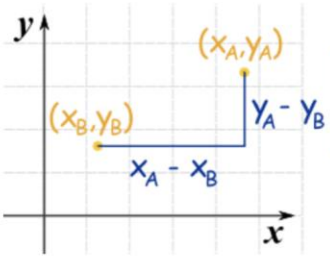
What is the distance between them?



We can run lines down from A, and along from B, to make a Right Angled Triangle.

And with a little help from Pythagoras we know that:

$$a^2 + b^2 = c^2$$



Now label the coordinates of points A and B.

x_A means the x-coordinate of point A

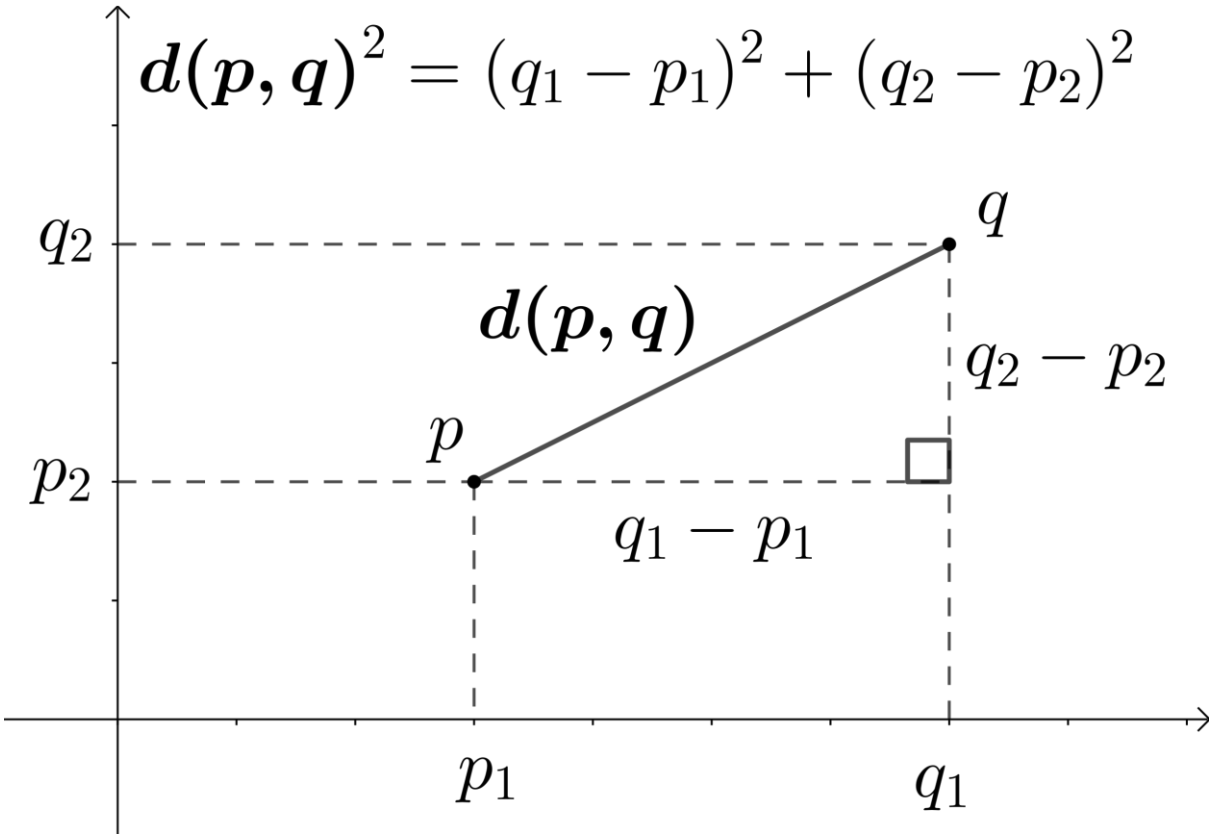
y_A means the y-coordinate of point A

The horizontal distance **a** is $(x_A - x_B)$

The vertical distance **b** is $(y_A - y_B)$

distance between xa,ya and xb,yb:

$$c = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

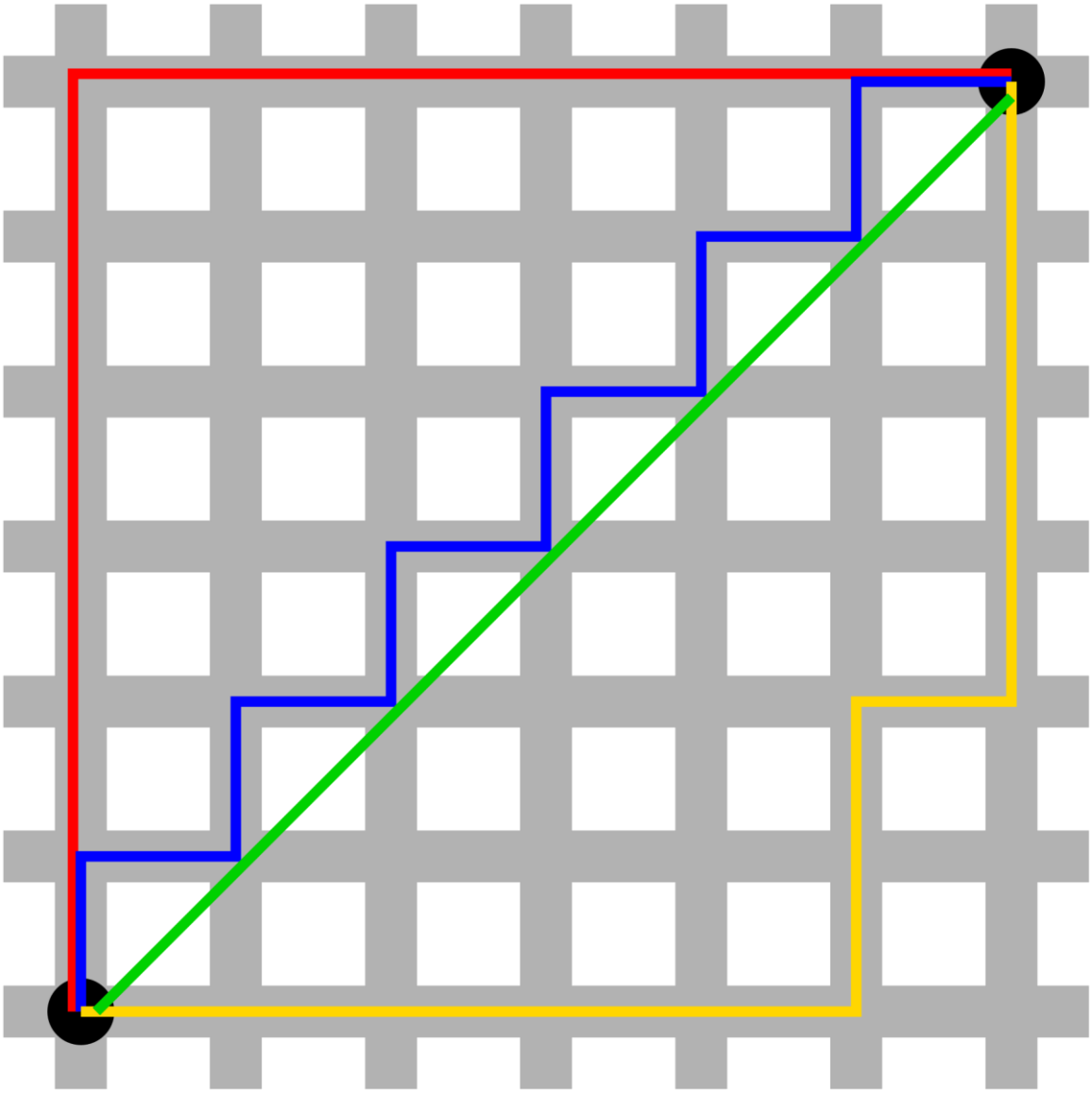


L2 Norm

L1 Norm

Manhattan distance : sum of absolute vector values.

$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|,$$



Taxicab geometry versus Euclidean distance: In taxicab geometry, the red, yellow, and blue paths all have the same shortest path length of 12. In Euclidean geometry, the green line has length 8.49 and is the unique shortest path.

In chess, the distance between squares on the chessboard for rooks is measured in taxicab distance; kings and queens use Chebyshev distance, and bishops use the taxicab distance (between squares of the same color) on the chessboard rotated 45 degrees, i.e., with its diagonals as coordinate axes. To reach from one square to another, only kings require the number of moves equal to their respective distance; rooks, queens and bishops require one or two moves (on an empty board, and assuming that the move is possible at all in the bishop's case).

Minkowski distance

The **Minkowski distance** or **Minkowski metric** is a metric in a normed vector space which can be considered as a generalization of both the Euclidean distance and the Manhattan distance.

$$D(X,Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{\frac{1}{p}}$$

$$X = (x_1, x_2, \ldots, x_n) \text{ and } Y = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n$$

Lp Norm

Hamming Distance

==========

Hamming distance is a metric for comparing two binary data strings. While comparing two binary strings of equal length, **Hamming distance** is the number of bit positions in which the two bits are different.

it measures the minimum number of *substitutions* required to change one string into the other, or the minimum number of *errors* that could have transformed one string into the other. In a more general context, the Hamming distance is one of several string metrics for measuring the edit distance between two sequences.

"**karolin**" and "**kathrin**" is 3.

"**karolin**" and "**kerstin**" is 3.

"**kathrin**" and "**kerstin**" is 4.

```
def hamming_distance(string1, string2):
      dist_counter = 0
      for n in range(len(string1)):
            if string1[n] != string2[n]:
                  dist_counter += 1
      return dist_counter
```

or

```
sum(xi != yi for xi, yi in zip(x, y))
```

It is used in telecommunication to count the number of flipped bits in a fixed-length binary word as an estimate of error, and therefore is sometimes called the **signal distance**.

However, for comparing strings of different lengths, or strings where not just substitutions but also insertions or deletions have to be expected, a more sophisticated metric like the Levenshtein distance is more appropriate.

4. What is Cosine Distance & Cosine Similarity?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2930/cosine-distance-cosine-similarity/3/module-3-foundations-of-natural-language-processing-and-machine-learning

   Cosine distance = 1- Cosine Similarity

   Cosine similarity = cos of angle between them

   cos 0 =1

   cos 90 = 0

   cos 180 = -1

   two vectors with the same orientation have a cosine similarity of 1, two vectors oriented at 90° relative to each other have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. The cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1]

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

   if ||A|| and ||B|| are unit vectors then similarity = cos theta = A.B

The resulting similarity ranges from −1 meaning exactly opposite, to 1 meaning exactly the same, with 0 indicating orthogonality or decorrelation, while in-between values indicate intermediate similarity or dissimilarity.

For text matching, the attribute vectors *A* and *B* are usually the term frequency vectors of the documents. Cosine similarity can be seen as a method of normalizing document length during comparison.

In the case of information retrieval, the cosine similarity of two documents will range from 0 to 1, since the term frequencies (using tf–idf weights) cannot be negative. The angle between two term frequency vectors cannot be greater than 90°.

5. How to measure the effectiveness of k-NN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2931/how-to-measure-the-effectiveness-of-k-nn/3/module-3-foundations-of-natural-language-processing-and-machine-learning

Split dataset Dn into Dtrain Dcv and Dtest. find nearest neighbors using Dtrain and find k using Dcv. Test on Dtest.

for each point in Dtest,

1. make xq = point
2. Use Dtrain to find k-nn and to predict yq

   if yq == ypt

   cnt +=1

   at end of loop, Accuracy = count of points correctly predicted/total points in Dtest.

6. Limitations of KNN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2933/knn-limitations/3/module-3-foundations-of-natural-language-processing-and-machine-learning

   because of large time and space complexity (O(nd)) - slower

7. How to handle Overfitting and Underfitting in KNN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2935/overfitting-and-underfitting/3/module-3-foundations-of-natural-language-processing-and-machine-learning

   k = 1 Overfitting - no mistakes - Low train error but high Error_CV

   k = very large close to n = Underfitting ⇒ Every point will be classified to majority class → High Train error , High CV error

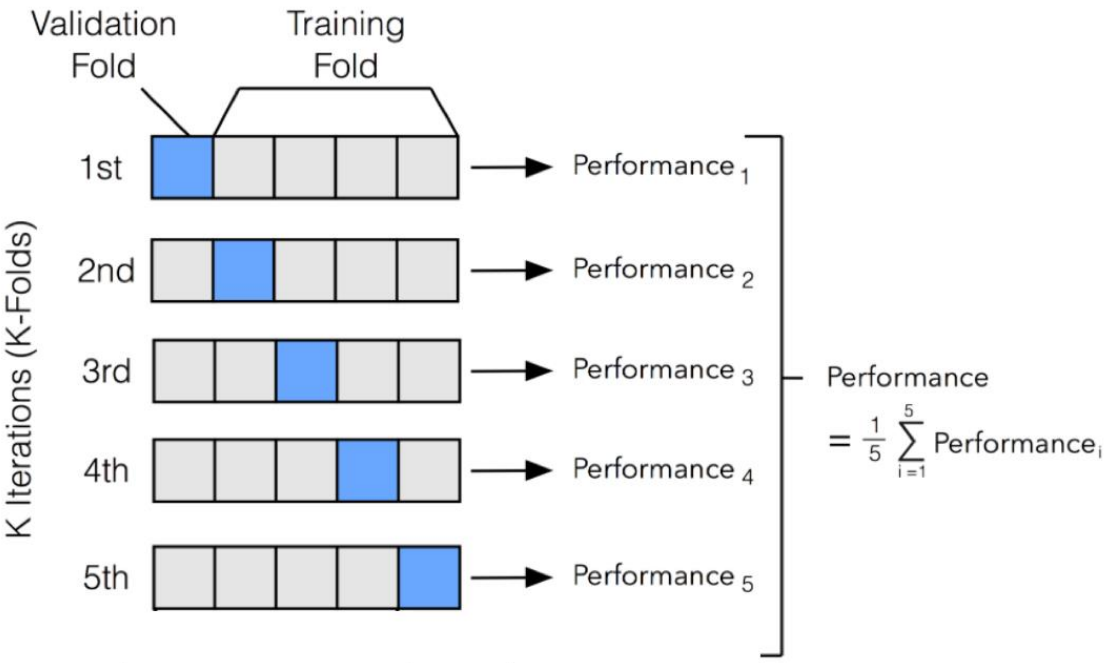
   Best fit → when training error is close to Validation error(Error_CV)

8. Need for Cross validation?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2936/need-for-cross-validation/3/module-3-foundations-of-natural-language-processing-and-machine-learning

   To determine k, Dtest is used however it should be unseen data. Hence Cross validation data is used to compute K and evaluate on Dtest.

9. What is K-fold cross validation?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2937/k-fold-cross-validation/3/module-3-foundations-of-natural-language-processing-and-machine-learning

k that refers to the number of groups that a given data sample is to be split into.



10. What is Time based splitting?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2940/time-based-splitting/3/module-3-foundations-of-natural-language-processing-and-machine-learning

11. Explain k-NN for regression?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2941/k-nn-for-regression/3/module-3-foundations-of-natural-language-processing-and-machine-learning

    Rather than majority voting, compute yq using Median/Mean of yi i=1 to k.

12. Weighted k-NN ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2942/weighted-k-nn/3/module-3-foundations-of-natural-language-processing-and-machine-learning

    Assign a weight = 1/distance from query point

    Sum weights of +ve and -ve points, who has greater sum, choose that class.

13. How to build a kd-tree.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2945/how-to-build-a-kd-tree/3/module-3-foundations-of-natural-language-processing-and-machine-learning

14. Find nearest neighbors using kd-treehttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2946/find-nearest-neighbours-using-kd-tree/3/module-3-foundations-of-natural-language-processing-and-machine-learning

15. What is Locality sensitive Hashing (LSH)?(
16. Hashing vs LSH?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2949/hashing-vs-lsh/3/module-3-foundations-of-natural-language-processing-and-machine-learning

    Nearby points goto same hash bucket.

    LSH is randomized algo that is it gives probabilistic output.

17. LSH for cosine similarity?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2950/lsh-for-cosine-similarity/3/module-3-foundations-of-natural-language-processing-and-machine-learning

    w = vector normal to hyperplane

    w = numpy.random.normal(0,1,d)

    d = dimensions, 0 = Mean, 1= variance

    for each slice, create m-dimensional vector where m is no of hyperplanes

    compute h(xq) as

Sign(w1T * xq)	Sign(w2T * xq)	Sign(w3T * xq)

    d.get(h(x)) will give the nearest neighbors.

    sign(w1T * x)

    wiT * x is +ve for all points which are in same direction as of wi.

    This h(x) will be key to hashtable and will store values of nearest neighbors in that slice.

18. LSH for euclidean distance?https://www.appliedaicourse.com/lecture/11/applied-machine-

learning-online-course/2951/lsh-for-euclidean-distance/3/module-3-foundations-of-natural-language-processing-and-machine-learning

Project the points on plane axis. Divide into regions say ‘a’ regions. Nearest neighbors will fall in same region.

h(x) will now have region in which point x lies.

INTERVIEW QUESTION ON K-NN

https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/

1. In k-means or kNN, we use euclidean distance to calculate the distance between nearest neighbours. Why not manhattan distance ?(https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/)

   In kNN algorithm we use minimum distance from a point to another and since euclidean distance gives the shortest or minimum distance between two points,that's we use euclidean distance instead of manhattan distance.

   When do we use MAnhattan?

   https://soundcloud.com/appliedai-course-185049385/why-not-manhattan-distance-in

   There is no fixed number after which we consider data as high-dimensional,but in general when the number of features are more than number of data points we consider the data as high dimensional data

   you can refer this link

   The Manhattan distance between two points on a grid is: The sum of the vertical and horizontal distances between them.

   like we are in the space right like in iris dataset shape is (150,4) we are in 4-dimensional space. where we donot have any blocks in between as we see in manhattan area of New york. so if we donot have any blocks we can directly find distance between the two while if we look in manhattan area we have to only move on the road to reach one point from other. refer this {consider lines as roads and squares as blocks while if we have blocks we have to move on roads{manhattan} while if no blocks we can move across square to reach our destination} typically in cases where dimensionality is large and euclidean distance fails we can use manhattan distance and when each feature is of different type we can use manhattan distance. refer this and this

2. How to test and know whether or not we have overfitting problem?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/how-to-determine-overfitting-and-underfitting/)

3. How is kNN different from k-means clustering?(https://stats.stackexchange.com/questions/56500/what-are-the-main-differences-between-k-means-and-k-nearest-neighbours)
4. Can you explain the difference between a Test Set and a Validation Set?(https://stackoverflow.com/questions/2976452/whats-is-the-difference-between-train-validation-and-test-set-in-neural-netwo)
5. How can you avoid overfitting in KNN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/how-to-determine-overfitting-and-underfitting/)

Which of the following machine learning algorithm can be used for imputing missing values of both categorical and continuous variables?

A) K-NN

B) Linear Regression

C) Logistic Regression

The answer is KNN because it makes predictions on the basis of the points in the neighborhood. Whereas other algorithms like Logistic Regression, sVm, etc work on the basis of creating a hyper-plane that separates the classes.

Building some insights based on the neighborhood points behaviour gives more valid and reasonable results. Hence the preferred algorithm here is KNN.

Imputing means filling. The data we get might have some missing values(NA). So the techniques that are used to fill those missing values with some other value are called imputation techniques.

Use D_train to train your model. To pick the best model with the best parameters you'd perform cross-validation. In a C(not K as we don't want to get confused with K in KNN) fold cross validation you'd subdivide the D_train into equal C subdivisions.

>> For example: if C = 5.

>> D_train = [D1, D2, D3, D4, D5] each of them equal split.

You'd train the model 5 times:

a. iteration1:

D_train = [D1,D2,D3,D4] and D_cross_validation = [D5], trained_model = M1, error = E1

b. iteration2:

D_train = [D1,D2,D3,D5] and D_cross_validation = [D4], trained_model = M2, error = E2

c. iteration3:

D_train = [D1,D2,D4,D5] and D_cross_validation = [D3], trained_model = M3, error = E3

d. iteration4:

D_train = [D1,D3,D4,D5] and D_cross_validation = [D2], trained_model = M4, error = E4

e. iteration5:

D_train = [D2,D3,D4,D5] and D_cross_validation = [D1], trained_model = M5, error = E5

Now this step is performed with every value of K(K in KNN). Say there are 10 K values then you'd have trained the model 10*5 times. For every value of K you have 5 cross-validation scores which you'd average. Plot this averaged error for each of the k-value. Choose the model with the least error. Apply D_test to this model(k value which returns the least accuracy).

>> If the train error is low and test error is high -> overfit

>> If the train error is very high and test error is very high -> underfit

So you'd have to find that balance in the train error and test error to call it an optimal model. Have a look at this blog as well :https://machinelearningmastery.com/k-fold-cross-validation/

Which distance to use when??

There are many other distance measures that can be used, such as Euclidean, manhattan and cosine distance. However, there is no 'best' distance, because the distance fundamentally depends on the form of data representation you choose, especially considering various structural representations.You can choose the best distance metric based on the properties of your data. If you are unsure, you can experiment with different distance metrics and different values of K together and see which mix results in the most accurate models.

Euclidean is a good distance measure to use if the input variables are similar in type (e.g. all measured widths and heights). Manhattan distance is a good measure to use if the input variables are not similar in type (such as age, gender, height, etc.). Due to the curse of dimensionality, we know that euclidean distance becomes a poor choice as the number of dimensions increases so use Cosine distance.

https://www.kdnuggets.com/2017/04/must-know-curse-dimensionality.html

https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/

Can you please explain how the answer to the 22nd question in the analyticsvidhya link provided is B.

I didn't quite get it.

Leave-one-out (LOO) CV is an extension of k'-fold CV wherein we select a k for k-NN and we keep just one point from the total data as a CV dataset and train the model with rest of the (n-1) points. Now, we measure the model's performance on the CV-dataset which contains just one point. Then, this gets repeated n times (where n= #points) wherein we leave one point each time in he CV dataset. Finally, we take the average performance of the k-NN model across these n-models.

The above gets repeated for multiple values of "k" and the best k is chosen like in any CV based hyper-parm tuning.

In the diagram provided in Q-22, the correct answer is B i.e., K=2 primarily because of the one negative points in between the +ve points. Just apply the above definition of leave-one-out-CV and you would be able to get lowest LOO-accuracy for k=2. Just try various values of k and get the LOO-CV accuracy for each of them. It needs some patience to try out various k values and computing accuracy for each of the n-iterations for each k.

What would be the time taken by 1-NN if there are N(Very large) observations in test data?.

Please explain the answer.

Time taken depends on the dimension of the test data as for each data point it will be same so O(d)

For concrete understanding kindly refer this thread

https://stats.stackexchange.com/questions/219655/k-nn-computational-complexity

---

If dataset has both continuous and categorical data,

You can go to a Euclidean distance or cosine distance.

If you want to check which distance metric is giving better distances, plot a histogram of pairwise distances of each point to another point for every distance metric if you want to check and then use the metric which histogram is not near to uniform distribution.

LOF - Local Outlier factor

It is a technique for outlier detection. LOF is not only for a particular cluster. It basically tells us about the outlier by looking at the density of the neighbourhood. this you will learn in future videos. Hence we can use LOF to remove outliers from the dataset.

Box-plots are for univariate outlier detection while LOF are for multivariate outlier detection. refer this

https://stackoverflow.com/questions/10238357/finding-the-outlier-points-from-matplotlib-boxplot

Till now we have learnt following techniques:

1) Boxplots

2) Interquartile Range (IQR) if point lies less than Q1-1.5IQR or more than Q2+1.5IQR then it is outlier.

3) This method LOF

We can find out the outliers using IQR or techniques like Local Outlier Factor, RANSAC, etc.

IQR is used to find out outliers among the scalars whereas techniques like LOF, RANSAC are used to find out the outliers among the vectors.

Local Outlier takes huge time as well as huge memory. So in practise, we can still use LOF, but if there is no need of low latency systems and memory consumption doesn't matter. But in general, we need to build low latency systems and these systems need to consume low amount of memory. Hence on larg scale projects, it isn't recommended to use LOF. You'll learn about techniques like RANSAC in future chapters, which is used for outlier removal.

========

we set a threshold on LOF value and discard those points that have a large value. We cannot check for errors here as no one told us which points are outliers. If they did, we could directly build a classification model itself. So, when we have to decide on the threshold, we have to use some of these methods:

1. Use manual validation and domain knowledge to determine which of the points flagged as outliers are actually outliers. We can choose our threshold now to ensure that our model mostly flags outliers only.

2. If you have lots of data and if you are ok throwing out x% of your data, you can choose a threshold that discard x% of data.

3. You can use elbow/knee method as we saw in determining "k" in k-nn to remove extreme outliers. Here, we can sort and plot LOF values on y-axis for each data-point represented on x-axis. You can now choose to use the elbow/knee point to discard outliers.

suggest you to pick up any simple data set from kaggle and work out this.try https://www.kaggle.com/mlg-ulb/creditcardfraud

how can we determine the appropriate K for LOF??

According to the original paper "The LOF values fluctuate wildy for k<10, with points in a uniform distribution sometimes showing up as outliers, so they recommend at least min(k)=10". Also you need to consider the value of k to be less than or equivalent to the cluster sizes in general. This is explained in this statement: "Secondly, the minimum k-value serves as a minimum size for something to be considered a "cluster", so that points can be outliers relative to that cluster. If k=15, and you have a group of 12 points and a point p, each point in the group will include p in its nearest neighbors, and p will include those points, leading them to have very similar LOFs."

https://stats.stackexchange.com/questions/138675/choosing-a-k-value-for-local-outlier-factor-lof-detection-analysis

"The authors of the paper recommend choosing a minimum k and a maximum k, and for each point, taking the maximum LOF value over each k in that range. They offer several guidelines for choosing the bounds."

Does this mean the following:

I select a range of range say k=[10,12,14,16,20]

For each k i calculate the LOF of each point

Then for a point i select the LOF value which is highest.

Which means for some values K=10 gives highest LOF and for others some other k

Then i use elbow method to plot the LOF and remove point which are above a threshold value.

In your elbow method plot, in one axis all the k values will be there and on the other axis LOFs and then you choose the right K. We plot the 'K' values on the X-axis and the LOF values on the Y-axis. At certain point, we'll see a sudden rise in the LOF value. That value of 'K' which gives sudden change in the shape is considered as the threshold. They have recommended the value of k to be at least k=10 as the values fluctuate for k.

leaf size parameter : since in KD-Tree we try to construct tree, leaf_size=30 indicates don't further split the node if that node contains 30 data points. with leaf_size we are restricting ourself's from constructing deep trees which consume a lot of memory and lot of retrieval time.

COLUMN STANDARDIZATION

feature transformation such that Mean = 0 Std Dev = 1

a’ = a-mu1/sigma1

b’ = b-mu2/sigma2

To standardize the query point, training data mean and standard deviation are used.

standardscalar.transform(query_point)

use standardscalar.fit() it saves all the means and variance of all features.

if our classification technique is scale dependent then, we first apply the column standardiation on the Dtrain to train the model on the tranformed dataset and then for each query point from Dcv, Dtest or Real Query Point in production apply standarization using the mean and standard deviation of feature(s)/ dimension(s) from Dtrain.

Interpretability

models which gives explanation to the output class labels in classification problems are called interpretable models and others are called Black box models.

https://christophm.github.io/interpretable-ml-book/interpretability-importance.html

FEATURE SELECTION

1. PCA is used exclusively to reduce the number of dimensions but not from the existing features. It creates all new features and if we want to reduce the dimensionality from **a** to **b** where (b<a) then among all the **a** features given as an input to the dataset, those top **b** features are selected which could preserve maximum variance of the data. We could not use PCA of Forward Feature Selection/Backward Elimination as **PCA belongs to the category of Feature Engineering Techniques and Forward Feature Selection/Backward Elimination belong to the category of Feature Selection Techniques.**

2. . In real world there can be cases when we have very high dimensional data and following that approach will be very time consuming but in some cases with very low dimension we can follow this approach and the result will certainly be better.

There are many feature selection methods available and chi square test is one of them where by calculating the Chi square scores for all the features, we can rank the features by the chi square scores, then choose the top ranked features for model training. Chi square comes under filter method of feature selection while forward/backward feature selection belongs to wrapper method of feature selection. We recommend you please go through this nice link to know about it in more detail: https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/

Feature engineering vs Feature Selection

feature engineering is to use domain knowledge to extract new features from raw data while feature selection is out of available many choices of selecting features which features have to be selected. refer this:click here

Lets make a note of difference between pca and forward or backward selection

1.forward or backward selection comes under feature selection techniques where you pick some subset of features from the original set

2.pca comes a dimensionality reduction technique where it create some complex features from the existing set of features and ,and then discards the less important ones

3.If the data is not linearly separable, then you have to try applying transforms to the existing features and check if you could separate the classes. If it works, then that would be fine. Otherwise, you need to go with creating a completely new set of features that could preserve either the maximum variance or the neighborhood,in this case we use pca because pca creates complex set of features from existing ones

4)In case, if you are asked not to create a new set of features or if you find the data already to be linearly separable and are looking to get rid of the curse of dimensionality, then you have to go for Feature selection.

when to use feature selection/extraction and when to use PCA/Tsne?

Since both refer to a reduction in the dimensionality of data, just that extraction reduces to a set of original dimensions and PCA/Tsne reduces them to a completely new set of features. But both of them are eventually surviving the same purpose.

Ans : First of all, you have to check if you are able to separate the data points linearly. If yes, then you can directly go ahead with building the model.

If the data is not linearly separable, then you have to try applying transforms (Transformations like squaring, applying sine() or log() functions etc) to the existing features and check if you could separate the classes. If it works, then that would be fine. Otherwise, you need to go with creating a completely new set of features that could preserve either the maximum variance or the neighborhood.

In case, if you are asked not to create a new set of features or if you find the data already to be linearly separable and are looking to get rid of the curse of dimensionality, then you have to go for Feature selection.

PCA and t-SNE both can perform dimensionality reduction. We can use PCA based features in modelling. But, t-SNE is only used for visualization. Because, tsne doesn't preserve distances or densities well. When we map our data from high dimensions to low dimension, we need to preserve the properties as well. In PCA, we preserve the variance of the data or the information loosely. So, we can use this for analysis. Tsne doesn't preserve distances or density. So, the data transformed to low dimension using tsne can't be used for analysis.

You dont have to do feature engineering while using deep learning because it automatically gets the feature extraction according to the problem at hand.

Keras: Feature extraction on large datasets with Deep Learning

https://www.pyimagesearch.com/2019/05/27/keras-feature-extraction-on-large-datasets-with-deep-learning/

How to check whether a data is linearly separable or not?

Visualizing the data is useful in such cases. Another option is training a linear classifiers and checking if you can get zero errors. Then your dataset is linearly separable. Otherwise, it's nonlinearly separable.

We can use **pair plot** to check if the data is linearly seperable or not . Please refer this : https://towardsdatascience.com/data-visualization-for-machine-learning-and-data-science-a45178970be7

Q8)Handling missing values by imputation?

Ans. In any dataset there may be chance that some entries are blank or filled with 'NaN'(not a number) or 'NULL' or -1 and we call them missing values.These values may be miss due to data corruption,collection error etc.And Handling missing data is important as many machine learning algorithms do not support data with missing values.So imputation is a process by which we try to replace missing values.SO we can apply imputation by taking mean,median or mode of non-missing values of the feature in which data is missing.

There are following ways to implement imputation:-

1)By taking mean,median or mode of non-missing values:For example let say a data has three features f1,f2 and f3.And there are two values which are missing in column of f3.So to replace these missing values we can take mean,median or mode of the feature f3.So let say we take mode of the f3 and it is m1(let),so now we will fill this m1 to all the missing entries in feature f3. And let say if the feature f3 contains text or categorical data then we replace the non missing values by filling them with most frequent text occur in non-missing values.

2)By taking mean,median or mode of non-missing values on the basis of class to which the missing value belong:There is another way of imputation which is used when there is dataset which includes classes.In this method we take mean,median or mode of only those non-missing values to which they should belong to the same class as the missing value is belonging. For example,let take same example with dataset of three features f1,f2 and f3.And let there is a single value is missed for the

feature f2 and all other values corresponding to f1,f3 and y(class) are present.Now suppose class belonging to this missing value is C1.So to replace this single value we will take mean,median or mode of only those non-missing values which have class as C1.After taking mean,median or mode we will fill the obtained value into the missing entry.

3)By creating new missing value feature:In this method we will build another set of features and fill them with 0's for non-missing values and 1's for missing values.let say we have a one missing value in f1 and two missing values in f2.Then we will make another set of features f1(dash),f2(dash) and f3(dash).And in these new features we will 0's to the corresponding non-missing values and fill 1's to the missing values.

4)Model based imputation:This method is based on K-NN.So if we have 100 missing values in f3 feature,then we will take f1,f2 as normal features and we will consider f3 as y.In this new f3 i.e y we will take all 100 missing values as test set and all non-missing values as training set.So now we have two parts of y-training set and test set.Now we will use these non--missing values from y and all the corresponding values from f1 and f2 as for training the model.Once we trained the model,we test our data points by using the model and finally get those missing values.

Irreducible error

A very clear understanding you can get from Robert Tibshirani's book called Introduction to Statistical Learning, first chapter. But let me explain you with an intuitive example. Let's say that you want to determine If the stock prices for a company would go up or down. Now you collected a lot of information about the company say last 5 days' price, any govt contract, a new deal with overseas partners etc. and then use this information to model it. But you are not God. You might have missed some unknown factor that governs the stock price of the company. That unknown feature is not even present in the dataset itself. Hence, no optimisation algorithm and no evaluation metric can trace it down. No matter how much you control the bias-variance tradeoff, you would never account for the error. But in reality, the stock price is also dependent on that unknown variable too. But because you have never considered that unknown variable, you don't even know how to reduce the error with respect to that variable. Hence this can be one example of Irreducible Error.

BIAS VARIANCE TRADE OFF

https://youtu.be/EuBBz3bI-aA

Bias-variance is more of mathematical concept and foundation used by lot of folks in research area. Over-fitting and under-fitting are used in practical aspects of machine learning by many ML practitioners.

Nothing can be done about irreducible error which is data dependent , acts as an upper bound on the accuracy of our prediction for Y . This bound is almost always

unknown in practice. It means whatever amount of irreducible error is present that could not be minimized In order to improve the model performance, the two parameters we could minimize as much as possible are the bias and the variance.

We just check whether the model has more bias or variance on the basis of train and cross validation errors through the process of Cross Validation.

If train error is low, but CV error is high, then the model overfits. ⇒ k=1

If both the train and CV errors are high, then the model underfits. ⇒ k=n

If CV error is low, then the model fits well.

If the training error is low, it means the model is trying the fit such that every point in the training data is classified correctly. There are also chances for outliers/noise to be present in the train data that could affect the model and making the model much more complex so that it fits every point correctly. Due to this complexity, when the model is allowed to make predictions on the unseen data(CV here), then it makes a huge number of misclassifications. This situation where the model performs well on the training data, but not on the CV data is called Overfitting.

If both the training and the CV errors are high, then it means the model is trying to build some random decision surface that is not much bothered about the proper classification of the points. This situation where the model doesn't care about the proper classification of the points and making a huge number of mistakes on both training and CV data is called Underfitting.

Revision Questions:

1. What is Imbalanced and balanced dataset. https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2958/imbalanced-vs-balanced-dataset/3/module-3-foundations-of-natural-language-processing-and-machine-learning
2. Define Multi-class classification? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2959/multi-class-classification/3/module-3-foundations-of-natural-language-processing-and-machine-learning
3. Explain Impact of Outliers? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2962/impact-of-outliers/3/module-3-foundations-of-natural-language-processing-and-machine-learning
4. What is Local Outlier Factor? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2963/local-outlier-factor-simple-solution-mean-distance-to-

knn/3/module-3-foundations-of-natural-language-processing-and-machine-learning

5. What is k-distance (A), N(A)
   https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2964/k-distance/3/module-3-foundations-of-natural-language-processing-and-machine-learning
6. Define reachability-distance(A, B)?
   https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2965/reachability-distanceab/3/module-3-foundations-of-natural-language-processing-and-machine-learning
7. What is Local-reachability-density(A)?
   https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2966/local-reachability-densitya/3/module-3-foundations-of-natural-language-processing-and-machine-learning
8. Define LOF(A)?
   https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2967/local-outlier-factora/3/module-3-foundations-of-natural-language-processing-and-machine-learning
9. Impact of Scale & Column standardization?
   https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2968/impact-of-scale-column-standardization/3/module-3-foundations-of-natural-language-processing-and-machine-learning
10. What is Interpretability?
    https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2969/interpretability/3/module-3-foundations-of-natural-language-processing-and-machine-learning
11. Handling categorical and numerical features?
    https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2971/handling-categorical-and-numerical-features/3/module-3-foundations-of-natural-language-processing-and-machine-learning
12. Handling missing values by imputation?
    https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2977/handling-missing-values-by-imputation/3/module-3-foundations-of-natural-language-processing-and-machine-learning
13. Bias-Variance tradeoff?
    https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2973/bias-variance-tradeoff/3/module-3-foundations-of-natural-language-processing-and-machine-learning

Precision & Recall

Precision is - out of the points the model predicted to be +ve, how many are actually positive. Model predicted +ve : TP+ FP

TP - actually positive

Precision = TP/TP+FP

Recall is basically out of total positives how many positives are you able to classify as positives.

Recall = TP/ P = TP/TP+FN

So lets say we were classifying some as negatives because of threshold , now as we lower the threshold those points which we were earlier classifying as negatives can now be classified as positives hence we can change the recall.

Usually, when we have probabilistic outputs we take anything greater than 0.5 as class 1 else class zero. But what we are suggesting here is take a different threshold, say anything greater than 0.3 is class 1 else it is class zero.

RoC and AuC

AUC doesn't depend on the actual values of the probability scores. It only depends on the ordering of the predicted probability scores. The shape of the ROC curve gets disturbed when we do not follow the ordering. Hence it is recommended for the data points to be sorted

best_thershold=thersholds[np.argmax(tpr * (1-fpr))]

TNR = TN/(TN+FP)

1. False positive - the test says you are pregnant when you aren't(as the threshold increases your false positive count decreases - most would be classified as not pregnant).

2. If FP decreases then TNR increases due to the inverse relation with TNR.

Consider this example

```
import numpy as np
from sklearn.metrics import roc_curve
y = np.array([0, 0, 1, 1])
# remember you need to have scores as probabilities
# scores = model.predict_proba(y)
scores = np.array([0.1, 0.4, 0.35, 0.8])
fpr, tpr, thresholds = roc_curve(y, scores)
print("tpr =",tpr)
print("fpr =", fpr)
print("thresholds = ",thresholds)
```

the output of the above code snippet

```
tpr = [0.5 0.5 1.  1. ]
```

```
fpr = [0.  0.5 0.5 1. ]
```

```
thresholds = [0.8  0.4  0.35 0.1 ]
```

The ROC_AUC will be calculated by keeping multiple thresholds on probability scores we are predicting

> The threshold values can be simply determined in a way similar to grid search

Conside the given example

For threshould value: 0.8

index	scores	Predicted(Threshould(>=0.8))	Original
1	0.1	0	0
2	0.4	0	0
3	0.35	0	1
1	0.8	1	1

	True Positive	True Negetive
Predicted Positive	1	0
Predicted Negetive	1	2

TPR = 1/(1+1) = 0.5
FPR = 0/(0+2) = 0

https://i.imgur.com/K6fSdWd.png

For threshould value: 0.4

index	scores	Predicted(Threshould(>=0.4))	Original
1	0.1	0	0
2	0.4	1	0
3	0.35	0	1
1	0.8	1	1

	True Positive	True Negetive
Predicted Positive	1	1
Predicted Negetive	1	1

TPR = 1/(1+1) = 0.5
FPR = 1/(1+1) = 0.5

https://i.imgur.com/7yiXmK7.png

For threshould value: 0.35

index	scores	Predicted(Threshould(>=0.35))	Original
1	0.1	0	0
2	0.4	1	0
3	0.35	1	1
1	0.8	1	1

	True Positive	True Negetive
Predicted Positive	2	1
Predicted Negetive	0	1

TPR = 2/(2+0) = 1
FPR = 1/(1+1) = 0.5

https://i.imgur.com/HEdDBAg.png For multiple values thresholds, you will get different set of (tpr, fpr) values, which will help you plot the ROC curve

We compute different TPRs and FPRs with different probability estimates as thresholds. But the ideal threshold would be the one which gives the maximum value of it's corresponding TPR*(1-FPR) value.

Idealy we want TPR ===> a very high value

FPR ===> a very low value

& since these 2 metrics range values are [0,1], so maximizing TPR*(1-FPR) is equivalent of maximizing TPR and Minimizing FPR. It's just mathematical way of representing 2 numbers, instead of dealing with 2 numbers.

If UniqueProba is in Ascending Order, We get Negative Values for AUC Else Positive

```
In [15]: TPRScores, FPRScores, AUC = AUC_Score(ascending=True)

         # To AUC Score Value
         print("AUC Score Value: ",AUC)

         100%|██████████| 9982/9982 [00:00<00:00, 28614.31it/s]

         AUC Score Value:  -0.48830149999999994
```

```
In [16]: TPRScores, FPRScores, AUC = AUC_Score(ascending=False)

         # To AUC Score Value
         print("AUC Score Value: ",AUC)

         100%|██████████| 9982/9982 [00:00<00:00, 18034.90it/s]

         AUC Score Value:  0.48830149999999994
```

Log-Loss

The log-likelihood function is merely the logarithm of the likelihood function. The log loss is defined "as the negative log-likelihood of the true labels given a probabilistic classifier's predictions"

We have to minimize the maximum liklihood function so we place negative sign.

The log loss is derived by assuming the outcomes are coming from bernoulli distribution.So, the y values are taken to be 1 or 0. You can even see this in the definition of log loss. -ylog(y^) - (1-y)log(1-y^) . Only if y takes 1 or 0, we choose between the two terms ylog(y^) and (1-y)log(1-y^). But you can plug in 1 or -1, but you won't get any useful result.

both log loss and auc score are trying to measure the model's discrimination power of positive and negative class ..

Log loss is small when the probabilities are more toward the ideals i.e. 1 and 0 .. a perfect model will output 1 for positive class and zero for negative class.

Similarly, auc score is using varying thresholds to get the auc metric which ultimately is proportional to P(x1>x2) when x1 belongs to positive class and x2 belongs to negative class.

This way if the log loss is high, this will mean that the model doesn't have a very good contrast between the class probabilities.

And for auc, if auc is low or close to 0.5 per se, this will mean the same thing !

If you care for a class which is smaller in number independent of the fact whether it is positive or negative, go for ROC-AUC score and If you care for absolute probabilistic difference, go with log-loss. Check out this excellent blog to see the comparison between through with an example:

https://medium.com/usf-msds/choosing-the-right-metric-for-evaluating-machine-learning-models-part-2-86d5649a5428

In auc , we are not determining right threshold- here we are keeping multiple thresholds based on unique probabilities scores so if you have n unique probabilities then for that we are having n pairs of tpr,fpr values that we are plotting and finally getting the auc score and we expect AUC to be >0.5 for any reasonable model.

Using AUC, we can select the best threshold for the model.

But when we calculate ROC_AUC_Score in sklearn, it internally selects many thresholds and checks which one is giving the optimal results and only returns that result. So, we can get to know how the model is performing too.

Among all the probability estimates, we have to sort them first and set the first one as a threshold. All the probability estimates greater than this threshold should be labelled as 1 and the others as 0. Now we should compare these labelled values with the actual values and check how many points are misclassified.,..

??????Repeat the same process for the 2nd, 3rd, 4th.. values and compute the total number of misclassified points. Whichever probability estimate gives least number of mis-classifications, that is considered as the best threshold.

To explain why the ROC and PR curves tell a different story, recall that the PR curve focuses on the minority class, whereas the ROC curve covers both classes.

If we use a threshold of 0.5 and use the logistic regression model to make a prediction for all examples in the test set, we see that it predicts class 0 or the majority class in all cases. This can be confirmed by using the fit model to predict crisp class labels, that will use the default threshold of 0.5. The distribution of predicted class labels can then be summarized.

```
...
# predict class labels
yhat = model.predict(testX)
# summarize the distribution of class labels
print(Counter(yhat))
```

We can then create a histogram of the predicted probabilities of the positive class to confirm that the mass of predicted probabilities is below 0.5, and therefore are mapped to class 0.

```
...
# create a histogram of the predicted probabilities
pyplot.hist(pos_probs, bins=100)
pyplot.show()
```

Tying this together, the complete example is listed below.

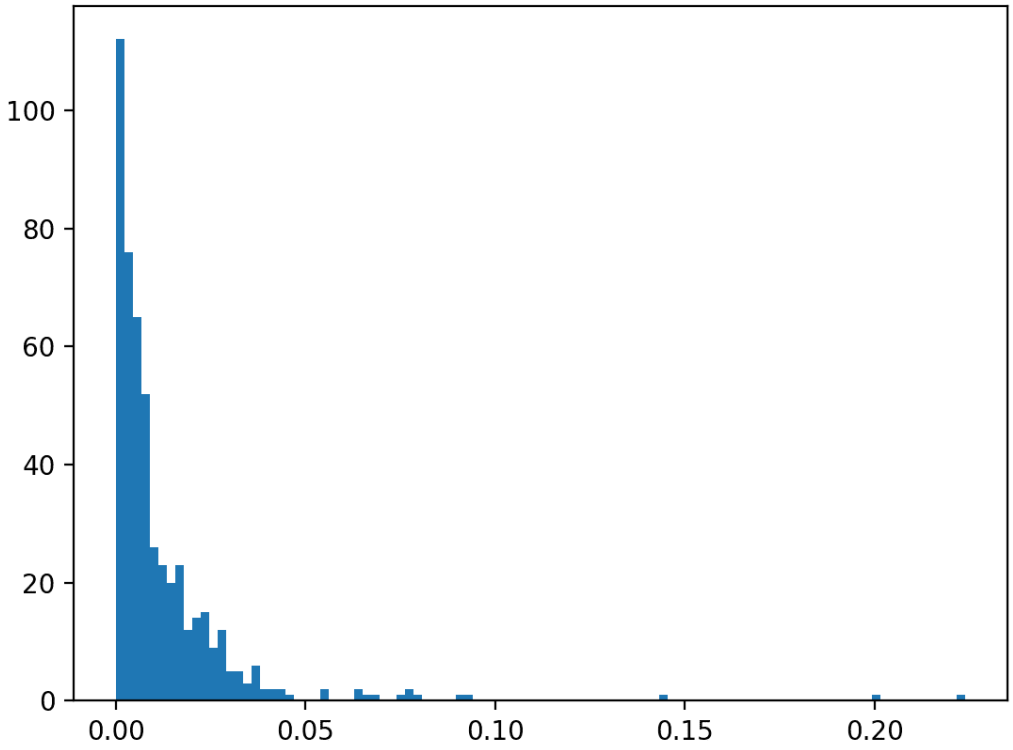
```
# summarize the distribution of predicted probabilities
from collections import Counter
from matplotlib import pyplot
from sklearn.datasets import make_classification
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
# generate 2 class dataset
X, y = make_classification(n_samples=1000, n_classes=2, weights=[0.99, 0.01], random_state=1)
# split into train/test sets with same class ratio
trainX, testX, trainy, testy = train_test_split(X, y, test_size=0.5, random_state=2, stratify=y)
# fit a model
model = LogisticRegression(solver='lbfgs')
```

```
model.fit(trainX, trainy)
# predict probabilities
yhat = model.predict_proba(testX)
# retrieve just the probabilities for the positive class
pos_probs = yhat[:, 1]
# predict class labels
yhat = model.predict(testX)
# summarize the distribution of class labels
print(Counter(yhat))
# create a histogram of the predicted probabilities
pyplot.hist(pos_probs, bins=100)
pyplot.show()
```

Running the example first summarizes the distribution of predicted class labels. As we expected, the majority class (class 0) is predicted for all examples in the test set.

```
Counter({0: 500})
```

A histogram plot of the predicted probabilities for class 1 is also created, showing the center of mass (most predicted probabilities) is less than 0.5 and in fact is generally close to zero.



Histogram of Logistic Regression Predicted Probabilities for Class 1 for Imbalanced Classification

This means, unless the probability threshold is carefully chosen, any skillful nuance in the predictions made by the model will be lost. Selecting thresholds used to interpret predicted probabilities as crisp class labels is an important topic

R2 - Coefficient of Determination

in ss_total we use the simple mean model where yq for every query point in dtest is y' i.e. mean of (lets say height) of all points in Dtest that is basically a measure of how a data set varies around a central number (like the mean) or you can say ss_total tells you how much variation there is in the dependent variable. One major use is in finding the coefficient of determination (R2). The coefficient of determination is a ratio of the explained sum of squares to the total sum of squares.

The problem with R2 is it's value gets increased even if unimportant features are present in the model. Due to this, even the random models also can have higher R2 scores.

R2 is generally used in Regression models and AUC is used in binary classification problems.

MAD (MEAN ABSOLUTE DEVIATION)

Before trying it out on any machine learning model, we can take yi^ = median of yi ,

now calculate error i and then calculate MAD. This can be set as a benchmark with which MAD of other models can be compared with.

For metrics such as log loss, MSE or MAE that doesn't have an upper limit. In order to set an upper limit, so that we can get to know how well our machine learning models are performing, we either take random values of yi or take mean or median and find the metric for these values. This concept is discussed in detail in case studies. Please refer this.

The main reason for taking MAD into consideration when compared to the standard deviation is that it doesn't get affected easily by the outliers. MAD gets affected by the outliers only if the median gets affected by the outliers. The median gets affected only if more than 50% of the points in our data are outliers.

The reason why MAD is still not being used in realtime is that all the algorithms are designed and implemented taking mean and SD into consideration. If we want to use MAD, we have to implement it manually.

Revision Questions Performance measurement of models

1. What is Accuracy ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2978/accuracy/3/module-3-foundations-of-natural-language-processing-and-machine-learning
2.
3. Explain about Confusion matrix, TPR, FPR, FNR, TNR?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2979/confusion-matrix-tpr-fpr-fnr-tnr/3/module-3-foundations-of-natural-language-processing-and-machine-learning
4.
5. What do you understand about Precision & recall, F1-score? How would you use it?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2980/precision-and-recall-f1-score/3/module-3-foundations-of-natural-language-processing-and-machine-learning
6.
7. What is the ROC Curve and what is AUC (a.k.a. AUROC)?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2981/receiver-operating-characteristic-curve-roc-curve-and-auc/3/module-3-foundations-of-natural-language-processing-and-machine-learning
8. What is Log-loss and how it helps to improve performance?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2982/log-loss/3/module-3-foundations-of-natural-language-processing-and-machine-learning
9.
10. Explain about R-Squared/ Coefficient of determinationhttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2983/r-squaredcoefficient-of-determination/3/module-3-foundations-of-natural-language-processing-and-machine-learning
11.

12. Explain about Median absolute deviation (MAD) ?Importance of MAD?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2984/median-absolute-deviation-mad/3/module-3-foundations-of-natural-language-processing-and-machine-learning
13.
14. Define Distribution of errors?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2985/distribution-of-errors/3/module-3-foundations-of-natural-language-processing-and-machine-learning

Questions

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

Model performance

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

Decision based on revenue prediction of a company, where if we predicted revenue is going to increase (FP) in near future however in actual it is not the decision company might have taken on prediction basis can impact company reputation

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

Generally medical test of any serious disease where ensuring no patient is wrongly classified if they are suffering with serious disease.

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

when we're classifying flowers into, say, two classes - Versicolor and Virginica, so here whether Versicolor is wrongly classified as Virginica or vice-versa both are equally bad, hence equal importance is given to FP and FN.

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

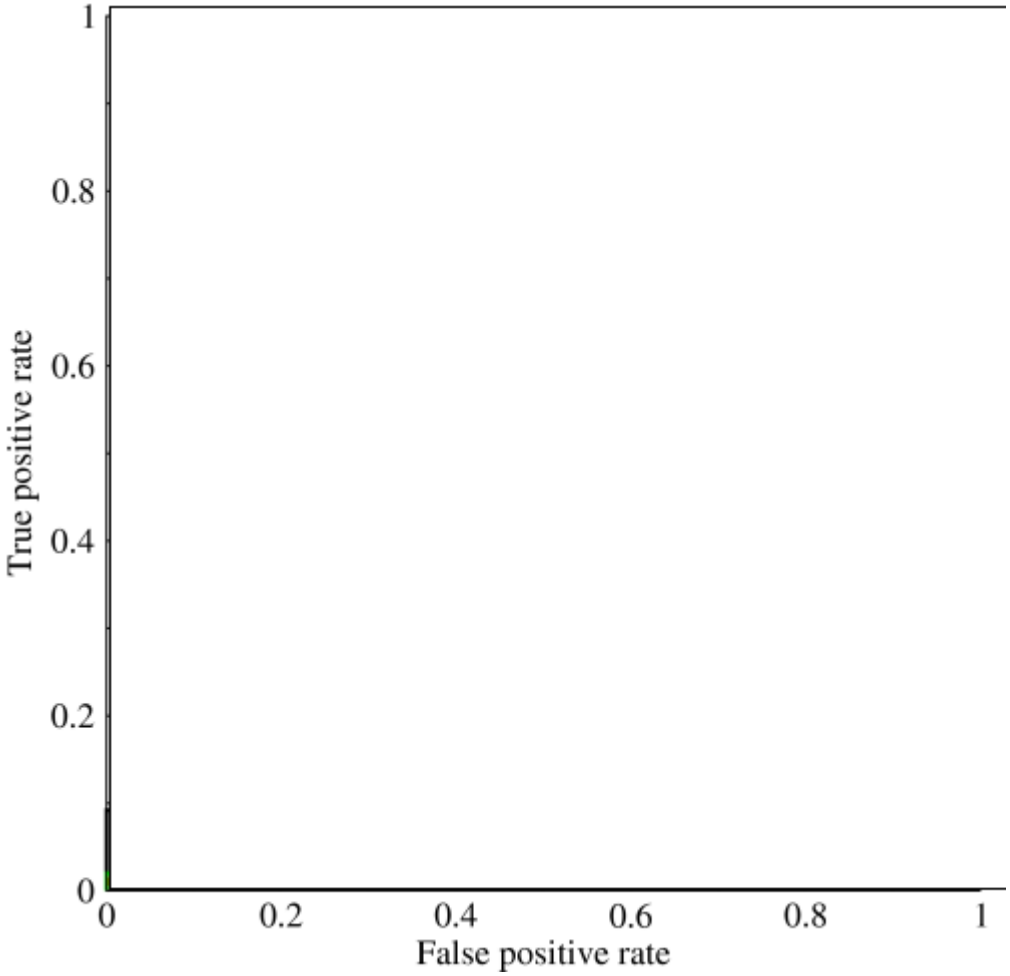
ROC in case of binary classification else confusion matrix

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

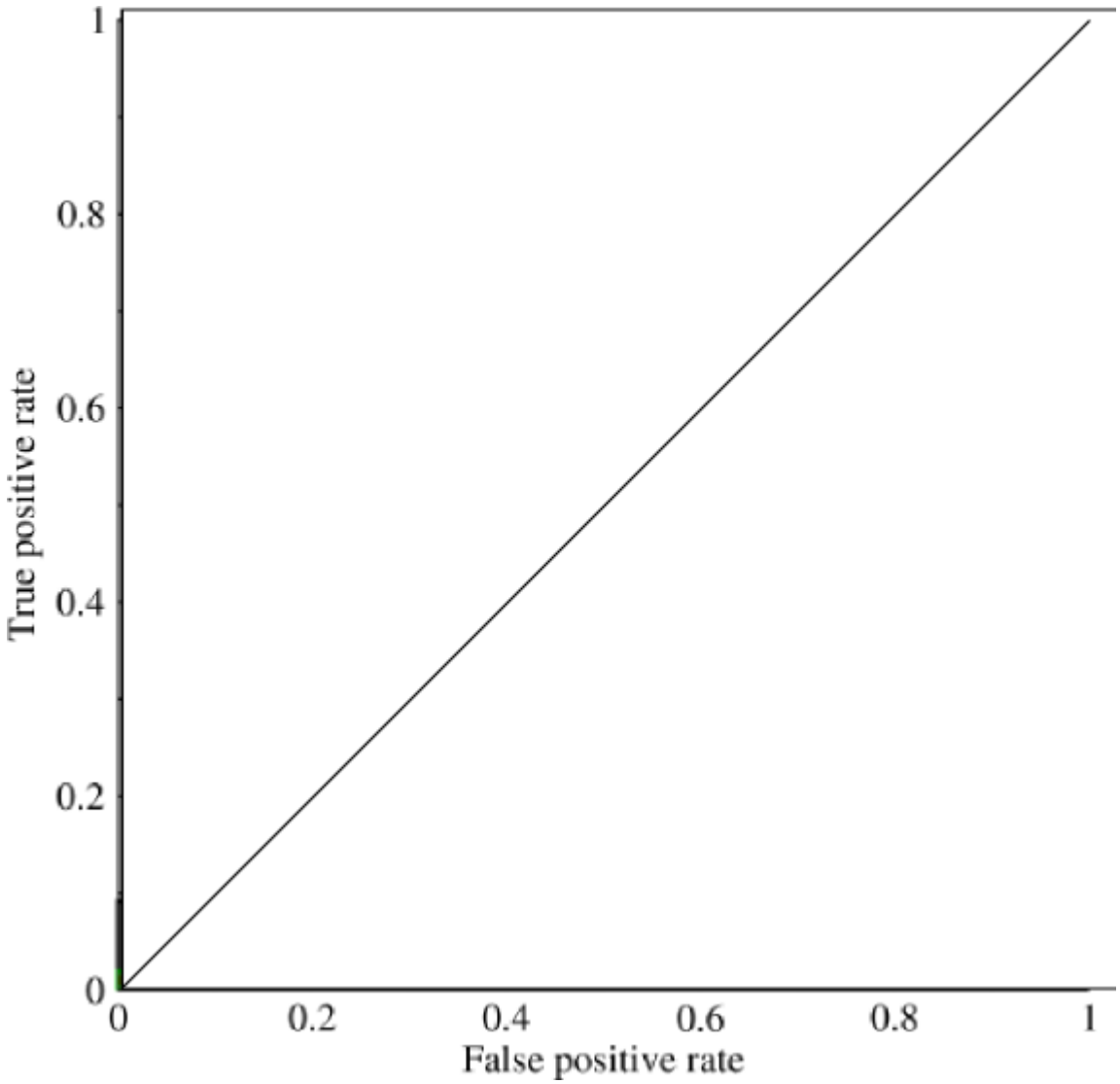
https://datascience.stackexchange.com/questions/806/advantages-of-auc-vs-standard-accuracy

Really great question, and one that I find that most people don't really understand on an intuitive level. AUC is in fact often preferred over accuracy for binary classification for a number of different reasons. First though, let's talk about exactly what AUC is. Honestly, for being one of the most widely used efficacy metrics, it's surprisingly obtuse to figure out exactly how AUC works.

`AUC` stands for `Area Under the Curve`, which curve you ask? Well, that would be the `ROC` curve. `ROC` stands for Receiver Operating Characteristic, which is actually slightly non-intuitive. The implicit goal of `AUC` is to deal with situations where you have a very skewed sample distribution, and don't want to overfit to a single class.
A great example is in spam detection. Generally, spam datasets are STRONGLY biased towards ham, or not-spam. If your data set is 90% ham, you can get a pretty damn good accuracy by just saying that every single email is ham, which is obviously something that indicates a non-ideal classifier. Let's start with a couple of metrics that are a little more useful for us, specifically the true positive rate (`TPR`) and the false positive rate (`FPR`):



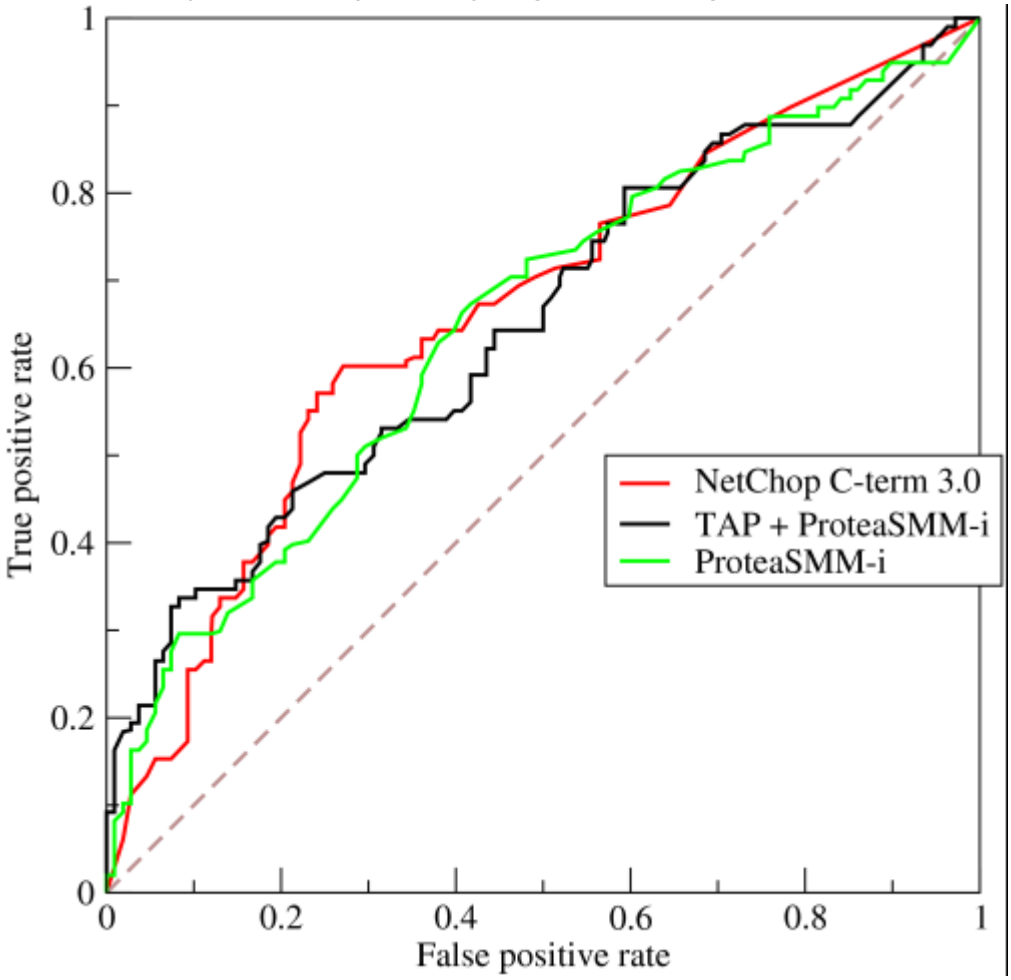
Now in this graph, `TPR` is specifically the ratio of true positive to all positives, and `FPR` is the ratio of false positives to all negatives. (Keep in mind, this is only for binary classification.) On a graph like this, it should be pretty straightforward to figure out that a prediction of all 0's or all 1's will result in the points of `(0,0)` and `(1,1)` respectively. If you draw a line through these lines you get something like this:



Which looks basically like a diagonal line (it is), and by some easy geometry, you can see that the `AUC` of such a model would be `0.5` (height and base are both 1). Similarly, if you predict a random assortment of 0's and 1's, let's say 90% 1's, you could get the point `(0.9, 0.9)`, which again falls along that diagonal line.

Now comes the interesting part. What if we weren't only predicting 0's and 1's? What if instead, we wanted to say that, theoretically we were going to set a cutoff, above which every result was a 1, and below which every result were a 0. This would mean that at the extremes you get the original situation where you have all 0's and all 1's (at a cutoff of 0 and 1 respectively), but also a series of intermediate states that fall within the `1x1` graph

that contains your ROC. In practice you get something like this:



So basically, what you're actually getting when you do an AUC over accuracy is something that will strongly discourage people going for models that are representative, but not discriminative, as this will only actually select for models that achieve false positive and true positive rates that are significantly above random chance, which is not guaranteed for accuracy.

-

AUC and accuracy are fairly different things. AUC applies to binary classifiers that have some notion of a decision threshold internally. For example logistic regression returns positive/negative depending on whether the logistic function is greater/smaller than a threshold, usually 0.5 by default. When you choose your threshold, you have a classifier. You have to choose one.

For a given choice of threshold, you can compute accuracy, which is the proportion of true positives and negatives in the whole data set.

AUC measures how true positive rate (recall) and false positive rate trade off, so in that sense it is already measuring something else. More importantly, AUC is not a function of threshold. It is an evaluation of the classifier as threshold varies over all possible values. It is in a sense a broader metric, testing the quality of the internal value that the classifier generates and then compares to a threshold. It is not testing the quality of a particular choice of threshold.

AUC has a different interpretation, and that is that it's also the probability that a randomly chosen positive example is ranked above a randomly chosen negative example, according to the classifier's internal value for the examples.
AUC is computable even if you have an algorithm that only produces a ranking on examples. AUC is not computable if you truly only have a black-box classifier, and not one with an internal threshold. These would usually dictate which of the two is even available to a problem at hand.
AUC is, I think, a more comprehensive measure, although applicable in fewer situations. It's not strictly better than accuracy; it's different. It depends in part on whether you care more about true positives, false negatives, etc.
F-measure is more like accuracy in the sense that it's a function of a classifier and its threshold setting. But it measures precision vs recall (true positive rate), which is not the same as either above.
On an imbalanced dataset using the majority run as a classifier will lead to high accuracy what will make it a misleading measure.

AUC: You should use it when you ultimately care about ranking predictions and not necessarily about outputting well-calibrated probabilities. You should not use it when your data is heavily imbalanced.

F1 is better where you care more about the positive class.

Metrics are totally problem-dependent.

Conditional Probability
https://www.youtube.com/watch?v=JGeTcRfKgBo
1.Random variables represent an event quantitatively. Since X and Y are two random variables they must also represent two events. Two events A and B are independent if and only if their joint probability equals the product of their probabilities: P(A and B) = P(A)P(B) Independence can be seen as a special kind of conditional independence, since probability can be seen as a kind of conditional probability given no events.
Check out the wikipedia link for more details.

2. "Is this when they are not correlated?" Not always if two random variables have a covariance of 0 they must be independent, is not true. In probability theory and statistics, two real-valued random variables, X,Y, are said to be uncorrelated if their covariance is zero. A set of two or more random variables is called uncorrelated if each pair of them are uncorrelated and there is no linear relationship between them. If X and Y are independent, with finite second moments, then they are uncorrelated. However, not all uncorrelated variables are independent. For example, if X is a continuous random variable uniformly distributed on [?1, 1] and Y = X2, then X and Y are uncorrelated even though X determines Y and a particular value of Y can be produced by only one or two values of X.
===== >> =====

understand the concepts like independence and correlation which we use in conditional probability.
1.independence implies uncorrelated
which means if two random variables are independent then they are definitely uncorrelated

2. Uncorrelated doesn't imply independence
Which means if two random variables are uncorrelated then they 'need not be' independent.
As we know that Correlation measures linear association between two given variables and it has no obligation to detect any other form of association else.
So those two variables might be associated in several other non-linear ways and correlation could not distinguish from the independent cases.
As a very didactic, artificial and non-realistic example, one can consider X such that P(X=x)=1/3 for x= - 1,0,1 and Y=X2. Notice that they are not only associated, but one is a function of the other. Nonetheless, their correlation is 0, for their association is orthogonal to the association that correlation can detect.
https://www.themathcitadel.com/uncorrelated-and-independent-related-but-not-equivalent/

Why Naive Bayes work comparatively better than Logistic on smaller data set?
Naive Bayes might do better than others for you because your dataset is very small and NB does not overfit as much as others. Naive Bayes is a generative model. It models p(x,y)
It is well known that when you have very little data that a generative model can beat a discriminative model like logistic regression. This paper by Andrew Ng has some good comparisons:
https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf

Also, go through this SO:
https://stackoverflow.com/questions/19129141/naive-bayes-and-logistic-regression-error-rate

Naive Bayes

Bayes Theorem named after the Reverend Thomas Bayes, describes the probability of an event, based on prior knowledge of conditions that might be related to the event.[2] For example, if the risk of developing health problems is known to increase with age, Bayes' theorem allows the risk to an individual of a known age to be assessed more accurately (by conditioning it on their age) than simply assuming that the individual is typical of the population as a whole.

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

P(A|B) : Posterior probability : Conditional prob : Probability of occurring event A given that B is true
P(B|A) : Likelihood of A given fixed B : Prob of occurring B given A is true
P(A) : Prior Prob
A and B are independent events
The Monty Hall problem is a very famous problem that has a counter-intuitive solution by using Bayes Theorem (https://brilliant.org/wiki/monty-hall-problem/).

Naive Bayes Classifier

https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/

Because it's Naive. it assumes the features are independent, the probabilities are incorrect if this assumption is not correct. eg assume you are predicting mortality based on smoking and drinking. NB may well identify people who smoke and drink as higher risk (=probability), just because there is a correlation between smoking and drinking. Suggest you setup a contingency table for eg smoking, drinking, dying and calcuate Naive bayes. probabilities vs true probabilities

Naive in Naive Bayes Classifier

If we didn't make the naive assumption then the algorithm would be totally complicated. Imagine you are tossing a coin n times and assume your tosses are not independent. Then what is the probability of P(HHHH) ? If it were independent, this would simply be P(H)*P(H)*P(H)*P(H). Now, we need to apply the chain rule if the events are not independen i.e,

P(HHHH) = P(H)*P(H/H)*P(H/HH)*P(H/HHH) . Now, we are having many terms. If the coins were independent it is enough to know about P(H) and P(T). With this we can calculate the P(HHHH) or probability of heads in any number of tosses easily b just multiplying. So, here we need to just store the P(H) and P(T) values. But when it is not independent, we need to store P(HH) , P(HHH) , P(H), P(HHHH) and note here we need to have 24 values to be stored . Why ? For two tosses, there are 4 possibilities and for n tosses there are 2n possibilities and for all 2n possibilities we need to compute the probability of that event and store it . It consumes more time and space. In computer science, we call it as a exponential algorithm and it is pretty bad. Just calculate 2100 and see yourselves. This is for binary outcomes. Imagine applying the same argument for a real valued feature. It would be worse than before. So, it is better to assume conditional independence (conditional independence just means given the Y, the X's are independent) to simplify the calculations and storage. Not all features are conditionally independent. But for applications like spam detection it works good. Even Yahoo and Microsoft deployed their spam detector using a variation of Naive Bayes algorithm.

==========

Why is naive Bayes a descent classifier but a bad estimator?

***The short answer**: The Naive Bayes classifier produces competitive classification accuracy, but pretty inaccurate data point-class label association probability estimates.*

***The longer answer**: When classifying a given data point, the Naive Bayes classifier first calculates the probabilities with which it believes the data points belong to each possible class label. It produces a classification by selecting the class label associated with the largest probability. Very often the largest probability is desirably associated with the correct class label. However, beyond the (important) close correlation between the largest probability and the correct class label, the probabilities are not correlated with classification confidence. The largest probability is frequently close to 1, with other probabilities close to 0. This indicates extremely high classification confidence, much more than empirically demonstrated actual classification accuracy justifies.*

***The reason**: The Naive Bayes classifier's conditional independence assumption, namely the assumption that features are independent of one another when conditioned upon class labels, is rarely accurate. Features often depend on one another non-trivial amounts, meaning multiple features often contain similar signals. However, the Naive Bayes classifier's conditional independence assumption results in its treatment of features as distinct signals each of which should independently contribute additional confidence to the classifier's prediction. This phenomenon amplifies the contribution of signals to the ultimate classification confidence. The Naive Bayes classifier nonetheless produces competitive classification accuracy because the extent to which the independence assumption favors different class labels roughly evens out on average.*

==========

https://towardsdatascience.com/multinomial-naive-bayes-classifier-for-text-analysis-python-8dd6825ece67

==========

Difference between P(A "intersection" B) and P(A|B)

The P(A and B) refers to the probability of both events occurring while P(A given B) refer to the individual probability of A occurring given that B has occurred if P(A and B)= P(A)*P(B|A) = P(B) * P(A|B)

i.e, P(A and B) is the probability of both events happening simultaneously.

P(A | B) is the probability of the subset of above events in which B has already occurred.

For example:Suppose that there's a town with 1000 people where everyone is either a Democrat or a Republican, and the demographics break down like this:

300 Republican men

200 Republican women

200 Democratic men

300 Democratic women

Let A be the event that a person is a Democrat, and let B be the event that a person is a woman. P(A?B) is the probability that a randomly chosen person is a Democratic woman, and is equal to 30%. P(A|B) is the probability that a randomly chosen woman is a Democrat, and is equal to 60%.

===

PRACTICE PROBLEMS FOR CONDITIONAL PROB

Let's consider this example of a conditional probability problem requiring Bayes' Theorem:

1% of OBU students are philosophy majors. 90% of OBU philosophy majors are accepted into their preferred graduate program. 30% of OBU non-philosophy majors are accepted into their preferred graduate program. Jane is an OBU student that was accepted into her preferred graduate program. What is the probability that she is a philosophy major?

Let, P = “An OBU student is a philosophy major” and A = “An OBU student was accepted into her preferred graduate program.”

The first step is to determine the conditional probability that the problem is asking us to solve. The first part is generally easy, just look for the question. In this case, “What is the probability that she is a philosophy major?” To find the given, look for the one thing that is known for certain; it won’t be a probability or percentage. We know that she was accepted into her preferred graduate program. So, we want to know the probability that Jane is a philosophy major given that she was accepted into her preferred graduate program, or Pr(P|A)Pr(P|A). Once that is determined, then simply write out the formula:

$Pr(P|A)=Pr(P)\times Pr(A|P)/\ Pr(P)\times Pr(A|P)+Pr(\neg P)\times Pr(A|\neg P)$

Now, we have to find the numbers to plug into the formula. Many, if not most, of the problems, are stated in terms of percentages. The probability of A given B is a function of the percentage of B’s that are A’s. That is, if A’s comprise half of the B’s, then Pr(A|B)=0.5Pr(A|B)=0.5

So,

$Pr(A|P)=0.9$
$Pr(A|\neg P)=0.3$
$Pr(P)=.01$
Now you can put the values into the above formula and solve it. You will learn by practicing more questions.

https://www.math.colostate.edu/~adams/teaching/math151win2012/m151midterm_solutions.pdf
solved using the integrated probability concept.
https://www.quora.com/If-a-point-is-inside-a-circle-then-what-is-the-probability-that-it-will-be-near-the-centre-of-a-circle

4. Four players Harry, Ron, Hermione and Ginny are playing a card game. A deck of 52 cards are dealt out equally. If Hermione and Ginny have a total of 8 spades among them, what is the probability that Harry has 3 of the remaining 5 spades?
(a) 0.669 (b) 0.339
(c) 0.331 (d) 0.661
ANS 4 players and 52 cards so each has 13 cards. harry and ron has total 26 cards out of which 5 are spades. now harry sould have 3 spades from 5 so 5C3 and remaining 10 cards from rest. so (26-5 = 21) i.e 21C10 divided by total combinations of cards harry gets from 26 choosing 13 at a time i.e 26C13

5. Prabha is working in a software company. Her manager is running a dinner for those employees having atleast one son. If Prabha is invited to the dinner and everyone knows she has two children. What is the probability that they are both boys?

This answer should be 1/2.

Reason :
Prabha can have 2 girls, one girl + one boy, 2 boys. (in the solution, {b,g} and {g,b} are considered as 2 different case, but we should consider it as combination, not a permutation as no order given between boy and girl). so 3 different combinations are possible.

As she invited in Party then one among the 2 cases among 3 might be true. and Probability of it is 2/3.

Now Probability of Prabha has 2 boys are 1/3.

Then answer would be ((1/3)/(2/3)) = ½

Q-7) Suppose that a bag contains 8 blue cubes and 4 green cubes. We draw 2 cubes from the bag without replacement. It is given that blue balls are of weight 1Kg and green balls are of weight 0.5 Kg. Suppose that the probability that a given cube in the bag is the next one selected is its weight divided by the sum of the weights of all cubes currently in the bag. What is the probability that both cubes are blue?
ANS
total weight = 10kg(8(1kg) + 4(0.5kg)),P(blue)=1/10 and P(green)=0.5/10(i.e 1/20),
P(blue/cube) = 8/12 * 1/10, P(green/cube) = 4/12 * 1/20
P(b1b2/cube) = 8/12 * 1/10 * A (here blue is picked without replacement so again calculate total weights)
total weight = 9kg(7(1kg) + 4(0.5kg)), P(blue) = 1/9 and P(green) = 0.5/9(i.e 1/18),
P(blue/cube) = 7/11 * 1/9, P(green/cube) = 4/11 * 1/18
A = P(b2/cube) = 7/11 * 1/9. so
P(b1b2/cube) = (8/12 * 1/10)* (7/11 * 1/9) = (2/30 * 7/99)

==========

Laplace smoothing

In general if alpha gets larger, the model move towards underfitting . And laplace smoothing will be used for all datapoints.

Hyperparameter tuning is basically used to adjust the model hyperparameters so as to maintain bias-variance tradeoff i.e., underfitting or overfitting that is done using a cross validation dataset, that result in the most skillful predictions. For example, value of k in K-nearest neighbor classification and alpha in naive bayes.

Why does the naive Bayes classifier perform well even when the features are correlated?

https://www.quora.com/Why-does-the-naive-Bayes-classifier-perform-well-even-when-the-features-are-correlated

After certain loss decrement, the loss does not decrease much, that means the loss seems constant after that point. or the accuracy seems constant. That means there wont be much improvement in accuracy further. That is what is known as asymptotic accuracy. And Naive bayes achieve that accuracy faster than other models as it is seen experimentally that it is around logn. But this doesnt make it better model to use. Although it performs decently given some data when compared t logistic regression, but if the data is very large then naive bayes will not perform better than LR.

==========
In Naive bayes, you can try both count based and binary BOW/TFIDF.
But in case of Count based vectorized data, you have to use MultinomialNB and in case of binary data, you have to BernoulliNB.

1. I didnt understood the the parameter "binarize". what is threshhold for binarizing?

The input given to BernoulliNB has to be in binary form. In case if the given data is not in binary form, then if we initialize the binarize value as 7.5(say for example) then all the values in the data matrix with >=7.5 will be changed as 1 and all other values <7.5 are changed as 0. Here 7.5 is the threshold.

2. how to we map class priors in array passed to "class_prior" parameter.
If we are already aware of the class probabilities(ie., P(C1),P(C2),P(C3)...P(Cn) where C1,C2,C3,....Cn are the classes in the dataset) then we have to mention tese values in an array and have to initialize 'class_prior' with this array. Otherwise, it wil automatically compute these class probabilities according to the dataset given.

https://stats.stackexchange.com/questions/33185/difference-between-naive-bayes-multinomial-naive-bayes

Multinomial Naive Bayes is used when the input data features consists of numerical discrete values (like counts of words, TF-IDF scores, etc)
Bernoulli Naive Bayes is used when the input data features consist of only boolean/binary values.
In case, if we have the data matrix containing only binary values, then we go for BernoulliNB. Otherwise we go for MultinomialNB.
In the assignment, if you want to build a Naive bayes model using Count based BOW or TF-IDF, then you have to go with MultinomialNB. Whereas if you are using binary BOW, then you have to go with BernoulliNB.
Let us assume our dataset consists of continuous, binary and disrete numerical data. Then
a) For continuous numerical features, we have to compute the likelihoods using the PDF formula

b) For discrete and binary data, we can compute the likelihoods by taking number of occurrences into consideration.

When it comes to class label prediction, you have to manually compute the values of P(yq=1|xq) and P(yq=0|xq)

https://stackoverflow.com/questions/61586946/how-to-calculate-feature-log-prob-in-the-naive-bayes-multinomialnb

Gaussian Naive Bayes implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian. Generally used if used if the given data is present in continuous format.

Multinomial Naive Bayes: multinomially distributed data used in text classification where the data are typically represented as word vector counts, {TF-IDF,BoW,W2V}

Bernoulli Naive Bayes: There may be multiple features but each one is assumed to be a binary-valued (boolean) variable this class requires samples to be represented as binary-valued feature vectors .{Binary Bow}

if the data is not gaussian distributed ? We can transform it to gaussian by using techniques like box cox transform.

If the given data is in numerical continuous format, then we have to go for GaussianNB.
If the given data is present in binary format, then we have to go for BernoulliNB.
If the given data is present in discrete numerical format, then we have to go for MultinomialNB.

The above mentioned variations of Naive bayes, give the best results on the type of data mentioned above. Our main intention is also to make the model give the better results in prediction.

https://scikit-learn.org/stable/modules/naive_bayes.html

Feature selection for categorical data
https://machinelearningmastery.com/feature-selection-with-categorical-data/
https://www.youtube.com/watch?v=fMIwIKLGke0 ⇒ Chi-Squared For Feature Selection using SelectKBest

1) For algorithm like KNN, in order to compute the feature importance, we have to use algorithm independent techniques like Forward Feature Selection/Backward Elimination Technique.
2) In Naïve bayes, if we want to find out the important features for both class 0 and class 1, and if we had 'd' features (say x_1,x_2,x_3,....x_d) Then

a) those features with higher values of P(y=1|x_i) are considered as important features associated with class 1
b) those features with higher values of P(y=0|x_i) are considered as important features associated with class 0

feature importance in Naive Bayes

feature importance is straight forward in NB. more likelihood probability more importance is that word. suppose for +ve class P(w1 | +ve) = 0.89 , P(w2 | +ve) = 0.08, P(w3 | +ve) = 0.23 then w1 is important feature in +ve class followed by w3 and lastly w2

```
S = ["abc def abc def","abc pqr pqr pqr cdf", "abc def pqr abc"]
y = [0, 1, 0]
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(S)
print(vectorizer.get_feature_names())
print(X.toarray())
clf = MultinomialNB()
clf.fit(X, y)
print(clf.feature_count_)
# we add alpha values = 1
smoothed_fc = np.log(clf.feature_count_+1)
# log(a/b) = log(a)-log(b)
#  P(Xi/y=0) = #number of times word appeared in class 0/ total number of words in class 0
# log(P(Xi/y=0))  = log(#number of times word appeared in class 0) - log(total number of words in class 0)
smoothed_cc = (clf.feature_count_+1).sum(axis=1)
smoothed_cc = np.log(smoothed_cc.reshape(-1, 1))
print(np.exp(smoothed_fc - smoothed_cc))
# Empirical log probability of features given a class, P(x_i|y)
# for each class you will find the probabilities which ever is more we can that feature is more importent
# here for the 0th class we have "abc", "def" and for the 1st class "pqr"
# the same works for tfidf also
print(np.all(np.exp(clf.feature_log_prob_)==np.exp(smoothed_fc - smoothed_cc)))
```

```
['abc', 'cdf', 'def', 'pqr']
[[2 0 2 0]
 [1 1 0 3]
 [2 0 1 1]]
[[4. 0. 3. 1.]
 [1. 1. 0. 3.]]
[[0.41666667 0.08333333 0.33333333 0.16666667]
 [0.22222222 0.22222222 0.11111111 0.44444444]]
True
```

the probability is calculated as P(Xi/y=0) = #number of times word appeared in class 0/ total number of words in class 0, the more the probability the more the importance. similarly for tfidf also

```
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(S)
print(vectorizer.get_feature_names())
print(X.toarray())
print(X.toarray()/X.toarray().sum(axis=1)[:,None])
clf = MultinomialNB()
clf.fit(X, y)
# Empirical log probability of features given a class, P(x_i|y)
np.exp(clf.feature_log_prob_)
# for each class you will find the probabilities which ever is more we can that feature is more importent
# here for the 0th class we have "abc", "def" and for the 1st class "pqr"
```

```
['abc', 'cdf', 'def', 'pqr']
[[0.61335554 0.         0.78980693 0.        ]
 [0.23069494 0.39060049 0.         0.89118522]
 [0.73941068 0.         0.47606294 0.47606294]]
[[0.43712368 0.         0.56287632 0.        ]
 [0.15252753 0.25825156 0.         0.58922091]
 [0.43712368 0.         0.28143816 0.28143816]]

array([[0.33162312, 0.14095031, 0.31937505, 0.20805152],
       [0.2232561 , 0.25226401, 0.18140653, 0.34307335]])
```

here we simply add +1 for smoothing and calculate probabilities as #number of times word appeared in class 0/ total number of words in class 0

Three friends in Seattle told you its rainy. Each of them has a probability of 1/3 of lying. What is the probability of Seattle being rainy? (Asked in Microsoft interview - https://www.glassdoor.com/Interview/Three-friends-in-Seattle-told-you-it-s-rainy-Each-has-a-probability-of-1-3-of-lying-What-s-the-probability-of-Seattle-is-QTN_2336604.htm)
Is the answer 8/54 correct?
https://mindyourdecisions.com/blog/2015/06/14/can-you-solve-this-facebook-interview-question-facebook-raining-in-seattle-probability
https://math.stackexchange.com/questions/1335235/facebook-question-data-science

How will you regularise your naive bayes model
1. Using smoothing techniques like Laplace smoothing
2. While creating the features, we can remove some features which rarely occur using some threshold so we may reduce some variance in the model..

Revision Questions Naive Bayes:

1. Bayes Theorem problem: https://youtu.be/LadMzl8MaXM
2. More Bayes Theorem problems: https://www.math.upenn.edu/~mmerling/math107%20docs/practice%20on%20Bayes%20solutions.pdf http://gtribello.github.io/mathNET/bayes-theorem-problems.html
   http://wwwf.imperial.ac.uk/~ayoung/m2s1/WorkedExamples1.pdf
3. What is Conditional probability? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2989/conditional-probability/3/module-3-foundations-of-natural-language-processing-and-machine-learnin
4. Define Independent vs Mutually exclusive events?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2990/independent-vs-mutually-exclusive-events/3/module-3-foundations-of-natural-language-processing-and-machine-learning
5. Explain Bayes Theorem with example? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2991/bayes-theorem-with-examples/3/module-3-foundations-of-natural-language-processing-and-machine-learning
6. How to apply Naive Bayes on Text data? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2995/naive-bayes-on-text-data/3/module-3-foundations-of-natural-language-processing-and-machine-learning
7. What is Laplace/Additive Smoothing? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2996/laplaceadditive-smoothing/3/module-3-foundations-of-natural-language-processing-and-machine-learning

8. Explain Log-probabilities for numerical stability? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2997/log-probabilities-for-numerical-stability/3/module-3-foundations-of-natural-language-processing-and-machine-learning
9. In Naive bayes how to handle Bias and Variance tradeoff? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/2998/bias-and-variance-tradeoff/3/module-3-foundations-of-natural-language-processing-and-machine-learning
10. What Imbalanced data? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3000/imbalanced-data/3/module-3-foundations-of-natural-language-processing-and-machine-learning
11. What is Outliers and how to handle outliers? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3001/outliers/3/module-3-foundations-of-natural-language-processing-and-machine-learning
12. How to handle Missing values? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3002/missing-values/3/module-3-foundations-of-natural-language-processing-and-machine-learning
13. How to Handling Numerical features (Gaussian NB) https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3003/handling-numerical-features-gaussian-nb/3/module-3-foundations-of-natural-language-processing-and-machine-learning
14. Define Multiclass classification.? https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3004/multiclass-classification/3/module-3-foundations-of-natural-language-processing-and-machine-learning

LOSS function

Loss function is basically the function which is minimized and whose optimum is tried to be achieved. Every algorithm has a loss function and it is different for different algorithms. You can think of it as a metric against which the model is evaluated. For eg, in logistic regression we are finding a plane which best separates our dataset. Hence here we have to find the equation of that plane. And after soing some modifications which are mathematically correct we arrived at the logloss function which we had to minimize.

---

1. I have personally seen elastic net being used in production as it tends to find a balance between L1 and L2. But, note that the elastic net is not always guaranteed to work better than L1 or L2 in terms of model performance. I personally like it as we are letting the dataset decide which regulation is better by just tuning the hyper-param, alpha, that balances L1 and L2.

2. Yes, they are used extensively to hyper-param tune in practice. While there are more mathematically powerful tools like Bayesian optimization for hyper-param tuning, they tend to be more cumbersome than these simpler alternatives. https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f

3. GridSearch and RandomSearch are techniques to pick a set of values for the hyper-parameters while CV (k-fold or simple-CV) performance/error is the key metric using which we decide which of these hyper-param values are optimal. So, GrdiSearch/RandomSearch are used in conjunction with Simple-CV or K-fold CV.

we can use Log Loss in Classification problems. Log Loss is the most important classification metric based on probabilities. It's hard to interpret raw log-loss values, but log-loss is still a good metric for comparing models. For any given problem, a lower log-loss value means better predictions.

What evaluation measure used when?

- Are you predicting probabilities?
    - Do you need class labels?
        - Is the positive class more important?
            - Use Precision-Recall AUC
        - Are both classes important?
            - Use ROC AUC
    - Do you need probabilities?
        - Use Brier Score and Brier Skill Score
- Are you predicting class labels?
    - Is the positive class more important?
        - Are False Negatives and False Positives Equally Important?
            - Use F1-Measure
        - Are False Negatives More Important?
            - Use F2-Measure
        - Are False Positives More Important?
            - Use F0.5-Measure
    - Are both classes important?
        - Do you have < 80%-90% Examples for the Majority Class?
            - Use Accuracy

Precisio Recall Area Under Curve
https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/#:~:text=The%20Precision%2DRecall%20AUC%20is,a%20model%20with%20perfect%20skill.
==========

Logistic regression is called "regression" and not classification because generally it produces real values of output based on the input features(just like linear regression works) but since these real values are applied to a sigmoid or a logistic function, so any value greater than 0.5 gets labelled as positive or 1 and values less than 0.5 gets labelled as 0 or negative.
https://www.quora.com/Why-is-logistic-regression-called-regression-if-it-doesnt-model-continuous-outcomes
https://stats.stackexchange.com/questions/127042/why-isnt-logistic-regression-called-logistic-classification

Why use Regularization
https://www.youtube.com/watch?v=iUm2Z1SKuGk

Why does PCA choose covariance matrix to get the principal components of features X?

https://www.quora.com/Why-does-PCA-choose-covariance-matrix-to-get-the-principal-components-of-features-X

https://www.youtube.com/watch?v=IdMZWFkVhGA

Revison Questions

1. **Explain about Logistic regression?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3011/geometric-intuition-of-logistic-regression/3/module-3-foundations-of-natural-language-processing-and-machine-learning**
2. **What is Sigmoid function & Squashing ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3012/sigmoid-function-squashing/3/module-3-foundations-of-natural-language-processing-and-machine-learning**
3. **Explain about Optimization problem in logistic regression. https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3013/mathematical-formulation-of-objective-function/3/module-3-foundations-of-natural-language-processing-and-machine-learning**
4. **Expalain Importance of Weight vector in logistic regression.https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3014/weight-vector/3/module-3-foundations-of-natural-language-processing-and-machine-learning**
5. **L2 Regularization: Overfitting and Underfittinghttps://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3015/l2-regularization-overfitting-and-**

underfitting/3/module-3-foundations-of-natural-language-processing-and-machine-learning

6. L1 regularization and sparsity. https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3016/l1-regularization-and-sparsity/3/module-3-foundations-of-natural-language-processing-and-machine-learning
7. What is Probabilistic Interpretation: Gaussian Naive Bayes ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3017/probabilistic-interpretation-gaussian-naive-bayes/3/module-3-foundations-of-natural-language-processing-and-machine-learning
8. Explain about Hyperparameter search: Grid Search and Random Search ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3019/hyperparameters-and-random-search/3/module-3-foundations-of-natural-language-processing-and-machine-learning
9. What is Column Standardization.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3020/column-standardization/3/module-3-foundations-of-natural-language-processing-and-machine-learning
10. Explain about Collinearity of features?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3022/collinearity-of-features/3/module-3-foundations-of-natural-language-processing-and-machine-learning
11. Find Train & Run time space and time complexity of Logistic regression?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3023/testrun-time-space-and-time-complexity/3/module-3-foundations-of-natural-language-processing-and-machine-learning

Interview Questions on Logistic Regression and Linear Regression

1. Outliers and Loss Functions: https://youtu.be/jiOBCCZCtug
2. After analyzing the model, your manager has informed us 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?(https://google-interview-hacks.blogspot.in/2017/04/after-analyzing-model-your-manager-has.html)
3. What are the basic assumptions to be made for linear regression?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intuition-1-2-copy-8/)
4. What is the difference between stochastic gradient descent (SGD) and gradient descent (GD)?(https://stats.stackexchange.com/questions/317675/gradient-descent-gd-vs-stochastic-gradient-descent-sgd)

5. When would you use GD over SDG, and vice-versa?(https://elitedatascience.com/machine-learning-interview-questions-answers)
6. How do you decide whether your linear regression model fits the data?(https://www.researchgate.net/post/What_statistical_test_is_required_to_assess_goodness_of_fit_of_a_linear_or_nonlinear_regression_equation)
7. Is it possible to perform logistic regression with Microsoft Excel?(https://www.youtube.com/watch?v=EKRjDurXau0)
8. When will you use classification over regression?(https://www.quora.com/When-will-you-use-classification-over-regression)
9. Why isn't Logistic Regression called Logistic Classification?(Refer :https://stats.stackexchange.com/questions/127042/why-isnt-logistic-regression-called-logistic-classification/127044)
More External Resources:
1.https://www.analyticsvidhya.com/blog/2017/08/skilltest-logistic-regression/
2.https://www.listendata.com/2017/03/predictive-modeling-interview-questions.html
3.https://www.analyticsvidhya.com/blog/2017/07/30-questions-to-test-a-data-scientist-on-linear-regression/ 4.https://www.analyticsvidhya.com/blog/2016/12/45-questions-to-test-a-data-scientist-on-regression-skill-test-regression-solution/
5. https://www.listendata.com/2018/03/regression-analysis.html

))

Differences between LinearSVC and SVC:

1) By default the loss minimized in SVC is the regular Hinge loss which has been discussed in our course. Whereas in LinearSVC, by default the squared hinge loss(square of this regular hinge loss) is minimized. If you want the regular hinge loss to be minimized in LineaSVC, then you have to specify the value manually.

2) While solving a multi-class classification problem, LinearSVC uses One-vs-Rest approach, whereas SVC uses One-vs-One approach.

For example, if we have total 5 classes in our dataset. They are 'A', 'B', 'C', 'D', 'E'. Then,
The models built by LinearSVC are (One-vs-rest approach)

A (vs) others (Predicts whether the point belongs to class 'A' or not)
B (vs) others (Predicts whether the point belongs to class 'B' or not)
C (vs) others (Predicts whether the point belongs to class 'C' or not)
D (vs) others (Predicts whether the point belongs to class 'D' or not)
E (vs) others (Predicts whether the point belongs to class 'E' or not)

The models built by SVC are (One-vs-One approach)

A (vs) B (Takes only the points of classes 'A' and 'B'. Predicts whether the point belongs to class 'A' or 'B')
A (vs) C (Takes only the points of classes 'A' and 'C'. Predicts whether the point belongs to class 'A' or 'C')
A (vs) D (Takes only the points of classes 'A' and 'D'. Predicts whether the point belongs to class 'A' or 'D')
A (vs) E (Takes only the points of classes 'A' and 'E'. Predicts whether the point belongs to class 'A' or 'E')
B (vs) C (Takes only the points of classes 'B' and 'C'. Predicts whether the point belongs to class 'B' or 'C')
B (vs) D (Takes only the points of classes 'B' and 'D'. Predicts whether the point belongs to class 'B' or 'D')
B (vs) E (Takes only the points of classes 'B' and 'E'. Predicts whether the point belongs to class 'B' or 'E')
C (vs) D (Takes only the points of classes 'C' and 'D'. Predicts whether the point belongs to class 'C' or 'D')
C (vs) E (Takes only the points of classes 'C' and 'E'. Predicts whether the point belongs to class 'C' or 'E')
D (vs) E (Takes only the points of classes 'D' and 'E'. Predicts whether the point belongs to class 'D' or 'E')

3) LinearSVC can be used only to solve only the problems associated with linearly separable data whereas SVC can be used for both linear as well as non-linear data.

4) The underlying estimators in LinearSVC are 'liblinear' which penalize the intercept. Whereas in SVC, the underlying estimators are 'libsvm'. 'liblinear' estimators are optimized for linear separable case and converge faster on large amounts of data when compared to 'libsvm'. That's the reason we see LinearSVC always to be faster than 'libsvm'.

Oneclass SVM is a technique that is used for novelty detection. Consider a data set of n observations from the same distribution described by p features. Consider now that we add one more observation to that data set. Is the new observation so different from the others that we can doubt it is regular? (i.e. does it come from the same distribution?) Or on the contrary, is it so similar to the other that we cannot distinguish it from the original observations? This is the question addressed by the novelty detection tools like one class SVM

SVM Revision Questions:

Explain About SVM?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3047/geometric-intution/4/module-4-machine-learning-ii-supervised-learning-models

What is Hinge Loss?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3050/loss-function-hinge-loss-based-interpretation/4/module-4-machine-learning-ii-supervised-learning-models

Dual form of SVM formulation.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3051/dual-form-of-svm-formulation/4/module-4-machine-learning-ii-supervised-learning-models

What is Kernel trick.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3052/kernel-trick/4/module-4-machine-learning-ii-supervised-learning-models

What is Polynomial kernel.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3053/polynomial-kernel/4/module-4-machine-learning-ii-supervised-learning-models
What is RBF-Kernel.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3054/rbf-kernel/4/module-4-machine-learning-ii-supervised-learning-models

Explain about Domain specific Kernels. ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3055/domain-specific-kernels/4/module-4-machine-learning-ii-supervised-learning-models

Find Train and run time complexities for SVM?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3056/train-and-run-time-complexities/4/module-4-machine-learning-ii-supervised-learning-models

Explain about SVM Regression. ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3058/svm-regression/4/module-4-machine-learning-ii-supervised-learning-models

1. **Give some situations where you will use an SVM over a RandomForest Machine Learning algorithm and vice-versa.(https://datascience.stackexchange.com/questions/6838/when-to-use-random-forest-over-svm-and-vice-versa)**
2. **What is convex hull ?(https://en.wikipedia.org/wiki/Convex_hull)**
3. **What is a large margin classifier?**

4. **Why SVM is an example of a large margin classifier?**
5. **SVM being a large margin classifier, is it influenced by outliers? (Yes, if C is large, otherwise not)**
6. **What is the role of C in SVM?**
7. **In SVM, what is the angle between the decision boundary and theta?**
8. **What is the mathematical intuition of a large margin classifier?**
9. **What is a kernel in SVM? Why do we use kernels in SVM?**
10. **What is a similarity function in SVM? Why it is named so?**
11. **How are the landmarks initially chosen in an SVM? How many and where?**
12. **Can we apply the kernel trick to logistic regression? Why is it not used in practice then?**
13. **What is the difference between logistic regression and SVM without a kernel? (Only in implementation – one is much more efficient and has good optimization packages)**
14. **How does the SVM parameter C affect the bias/variance trade off? (Remember C = 1/lambda; lambda increases means variance decreases)**
15. **How does the SVM kernel parameter sigma^2 affect the bias/variance trade off?**
16. **Can any similarity function be used for SVM? (No, have to satisfy Mercer's theorem)**
17. **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)**
18. **What is the difference between supervised and unsupervised machine learning?**

https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/

SVM interview ques session slides

https://docs.google.com/presentation/d/1pnYB7oZks2Lz5sHKINUnLJ_yZLOFoIYXnNF_kzDfrAU/edit#slide=id.g608a935b5f_0_114

Entropy

Entropy measures uncertainty or we can say randomness.In other words, its a measure of unpredictability.
"Does it say that the feature with higher entropy has more random values than the one with lower entropy?" -yes. Let's take an example of a coin toss.

Suppose we tossed a coin 4 times, and the output of the events came as {Head, Tail, Tail, Head}. Based solely on this observation, if you have to guess what will be the output of the coin toss, what would be your guess?

- two heads and two tails. Fifty percent probability of having head and fifty percent probability of having a tail. You can not be sure. The output is a random event between head and tail.

But what if we have a biased coin, which when tossed four times, gives following output: {Tail, Tail, Tail, Head}. Here, if you have to guess the output of the coin toss, what would be your guess? Chances are you will go with Tail, and why? Because seventy-five percent chance is the output is tail based on the sample set that we have. In other words, the result is less random in case of the biased coin than what it was in case of the perfect coin.

hence, We would say that an unbiased coin has a high entropy, because the uncertainty of the value of X is the highest possible.

The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain.
We use entropy of y_i's which we will in turn use to compute information-gain which in turn helps us decide which feature to split a decision tree at a level.

Information gain is KL divergence.
https://en.wikipedia.org/wiki/Information_gain_in_decision_trees

1. We calculate information gain for all types of features.
2. We find Infomation gain of all the features, we use the feature which maximizes the infomation gain.

If you look at binary cross entropy which is kind of KL divergence formula only. That is widely used in neural netwoks and for that the function must be differentiable.

Constructing Decision Tree

First we calculate the entropy of the complete data . Now we check for the information gain on each feature we do so by splitting at a particular feature and check how many positive and negative points are there .we calculate the weighted entropy i.e if we have 5 points in the sunny then 5/14 multiplied by the entropy at sunny now we add all the weighted entropies and subtract it from the entropy of the whole data. This is our information gain .Now the feature with max IG is chosen as the feature to split first .Now for each split we check the IG using the splitting at other features excluding the previously split feature . Now our data will be reduced and the number of points in the denominator will reduce as well , we ll get IG for the remaining features and choose the feature with the best IG. In this way we ll select the features and ignore the previously split features . We ll split the data until we reach our

explicitly set threshold of points or we reach the leaf nodes only . This way we ll be better classifying the query points.
We generally stop growing a tree according to max_depth which is a hyper-parameter. We tune this parameter to get an optimal value. The tree generally stops growing at that particular value of depth. Here, we choose majority voting as the nodes may not be pure. The label of majority points will be assigned as the class label.

Decision Tree Revision Questions:

1. How to Building a decision Tree?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3064/geometric-intuition-of-decision-tree-axis-parallel-hyperplanes/4/module-4-machine-learning-ii-supervised-learning-models

2. What is Entropy?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3066/building-a-decision-treeentropy/4/module-4-machine-learning-ii-supervised-learning-models

3. What is information Gain ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3067/building-a-decision-treeinformation-gain/4/module-4-machine-learning-ii-supervised-learning-models

4. What is Gini Impurity?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3068/building-a-decision-tree-gini-impurity/4/module-4-machine-learning-ii-supervised-learning-models

5. How to Constructing a DT. ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3069/building-a-decision-tree-constructing-a-dt/4/module-4-machine-learning-ii-supervised-learning-models

6. Importance of Splitting numerical features.?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3070/building-a-decision-tree-splitting-numerical-features/4/module-4-machine-learning-ii-supervised-learning-models

7. How to handle Overfitting and Underfitting in DT?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3073/overfitting-and-underfitting/4/module-4-machine-learning-ii-supervised-learning-models

8. What are Train and Run time complexity for DT?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3074/train-and-run-time-complexity/4/module-4-machine-learning-ii-supervised-learning-models

9. How to implement Regression using Decision Trees?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3075/regression-using-decision-trees/4/module-4-machine-learning-ii-supervised-learning-models

Video:
1. Maximum entropy distirbution: https://youtu.be/f6SpsPn6cvs

Self Learning:
1. You are working on a time series data set. You 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 decision tree model. Can this happen? Why?(Refer :https://www.analyticsvidhya.com/blog/2016/09/40-interview-questions-asked-at-startups-in-machine-learning-data-science/)
2. Running a binary classification tree algorithm is the easy part. Do you know how does a tree splitting takes place i.e. how does the tree decide which variable to split at the root node and succeeding nodes?(Refer:https://www.analyticsvidhya.com/blog/2016/09/40-interview-questions-asked-at-startups-in-machine-learning-data-science/)

External Resources: 1.https://vitalflux.com/decision-tree-algorithm-concepts-interview-question

Random Forest

1. Column sampling has been shown to avoid extreme overfitting by the base models across all features as each model only gets to see a subset of features. The base models also have high variance amongst themselves when we sample columns as each model gets to see different subsets of features and hence is more different from others.

2. In a nutshell, CSR and RSR control the amount of overfitting in each base model and the variance across each of the base models. AS a RandomForest simply takes

a majority vote of base models thereby reducing variance while keeping the bias the same, the base model variance and bias impacts the final model.

CSR and RSR are hyper-params as their choice impacts the model performance through bias-variance tradeoff. Actually, the most important hyper-params in RF to tune are the number of base learners, CSR and RSR.

Gradient boosting
https://explained.ai/gradient-boosting/index.html
https://en.wikipedia.org/wiki/Gradient_boosting#Algorithm

Pseudo-residuals for log-loss and classification

https://www.youtube.com/watch?v=qEZvOS2caCg

https://stats.stackexchange.com/questions/219241/gradient-for-logistic-loss-function

Revision ques : ENSEMBLE MODELS

Revision Questions:

1. What are ensembles?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3081/what-are-ensembles/4/module-4-machine-learning-ii-supervised-learning-models
2. What is Bootstrapped Aggregation (Bagging) ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3082/bootstrapped-aggregation-bagging-intuition/4/module-4-machine-learning-ii-supervised-learning-models
3. Explain about Random Forest and their construction?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3083/random-forest-and-their-construction/4/module-4-machine-learning-ii-supervised-learning-models
4. Explain about Boosting?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3089/boosting-intuition/4/module-4-machine-learning-ii-supervised-learning-models
5. What are Residuals, Loss functions and gradients ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3090/residuals-loss-functions-and-gradients/4/module-4-machine-learning-ii-supervised-learning-models
6. Explain about Gradient Boosting?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3091/gradient-boosting/4/module-4-machine-learning-ii-supervised-learning-models
7. What is Regularization by Shrinkage?https://www.appliedaicourse.com/lecture/11/applied-machine-

learning-online-course/3092/regularization-by-shrinkage/4/module-4-machine-learning-ii-supervised-learning-models

8. Explain about XGBoost?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3094/xgboost-boosting-randomization/4/module-4-machine-learning-ii-supervised-learning-models
9. Explain about AdaBoost?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3095/adaboost-geometric-intuition/4/module-4-machine-learning-ii-supervised-learning-models
10. How do you implement Stacking models?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3096/stacking-models/4/module-4-machine-learning-ii-supervised-learning-models
11. Explain about cascading classifiers. ?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3097/cascading-classifiers/4/module-4-machine-learning-ii-supervised-learning-models

==========

Time series data

Storing the time stamp at which a particular review was written in case of Amazon fine food reviews would make it a time series data. To build time series data, we need to have a timestamp at which that particular observation was taken as a feature in dataset.

What is the difference between sequence data and time series data?
Sequential Data is any kind of data where the order matters as you said. So we can assume that time series is a kind of sequential data, because the order matters. A time series is a sequence taken at successive equally spaced points in time and it is not the only case of sequential data. In the latter the order is defined by the dimension of time. There are other cases of sequential data as data from text documents, where you can take into account the order of the terms or biological data (DNA sequence etc.). The fact that you have sequential data is important for two reasons. First, you can into account for the representation of the data and also you can take it into account for the data modeling (e.g. Conditional Random Fields, Hidden Markov Models for text or genes and ARIMA Models for time series problems).

Fourier Transform

General intuition of Fourier transforms is
1.Representing/converting more complex waveform into simple individual waves so that we can extract most information out of it.Let me explain by taking a simple example ,assume you have a INDIAN CURRY of specific type .
2. In order to analyze(amount of ingredients) the prepared curry is so difficult i.e ingredients like amount of salt ,Turmeric powder, chilli powder and type of masala

etc .So we pass the prepared curry through a complex method called Fourier transforms so that we get most individual ingredients out of prepared curry.
3.Note that ingredients obtained is sometimes in its initial/pure stage or sometimes impure.But we can analyze individual ingredients better than from the processed curry.
4. After passing the curry through F.T we get ingredients individually and to re-obtain the curry back we pass it through Inverse Fourier transforms.
5. In mathematical point of view curry is any complex signal like audio /Video etc and ingredients refers to frequency ,amplitude and phase.

Fourier Transform:
https://www.youtube.com/watch?v=spUNpyF58BY

SIFT - Scale Invariant Feature Transform

a.
1. In general, SIFT aims to find highly-distinctive interest points(or keypoints) in an image. The locations are not just restricted to 2D coordinates in an image, but locations in the image's scale space, meaning they have three coordinates: x, y, and scale. The process for finding SIFT key points would involve:
1. First you'd need blur and resample the image with different blur widths and sampling rates to create a scale space
2. Use the difference of gaussians method to detect blobs at different scales; the blob centers become our keypoints at a given x, y, and scale.(That is why in general the key points are the corners of an image where the gradient w.r.t to x and y is high)
3. Assign every key point a 128-dimensional feature vector based on the gradient orientations of pixels in 16 local neighborhoods.

These key point descriptors have to be robust against image transformations and scale independent and it may seem like the edges of the door do not conform to these parameters.

b. Most image processing applications tend to use grayscale as it tends to have much less information to process with than a full color image. In the original paper it has been quoted as:
"The features described in this paper use only a monochrome intensity image, so further distinctiveness could be derived from including illumination-invariant color descriptors (Funt and Finlayson, 1995; Brown and Lowe, 2002).".

https://aishack.in/tutorials/sift-scale-invariant-feature-transform-scale-space/

==========
We can either one-hot encoding or use mean-replacement (or response-coding) where we replace each categorical value with P(y_i|categorical value).

https://developers.google.com/machine-learning/crash-course/cancer-prediction

If you want to feature-bin zipcode, you can do it by using geographic location. All zip codes in a city/state would lie in a range and hence can be thought of as one category in binning.

BINNING

1. https://stackoverflow.com/questions/14298433/convert-data-to-the-quantile-bin
2. quantile based binning: https://towardsdatascience.com/understanding-feature-engineering-part-1-continuous-numeric-data-da4e47099a7b

> - Bins should be all the same size. For example, groups of ten or a hundred.
> - Bins should include *all* of the data, even outliers. If your outliers fall way outside of your other data, consider lumping them in with your first or last bin.
> - The larger the data set, the more likely you'll want a large number of bins. For example, a set of 12 data pieces might warrant 5 bins but a set of 1000 numbers will probably be more useful with 20 bins. The exact number of bins is usually a judgment call.
> - If at all possible, try to make your data set evenly divisible by the number of bins. For example, if you have 10 pieces of data, work with 5 bins instead of 6 or 7.

https://towardsdatascience.com/understanding-feature-engineering-part-1-continuous-numeric-data-da4e47099a7b

https://stats.stackexchange.com/questions/68834/what-is-the-benefit-of-breaking-up-a-continuous-predictor-variable

Information Gain is used to decide, node split.
Steps it takes to split:
Consider a classification dataset with 2 features hair length, height
Class Label is male or female

Each decision to split the input dataset into 2 datasets is decided by a threshold value of a feature which can achieve Maximum Information Gain, or minimum Entropy(measure of impurity).
Here our interest is to maximize the information gain and not to get the equal bin sizes.
You can see that it never tries to keep equal bin ranges... It Focuses more on Producing more almost pure nodes wrt class labels.

So DT never guarantees you of equal bin sizes.

Equivalence of Gaussian Naive Bayes and Logistic Regression: An Explanation

https://appliedmachinelearning.blog/2019/09/30/equivalence-of-gaussian-naive-bayes-and-logistic-regression-an-explanation/

collinear and multicollinear

https://towardsdatascience.com/multicollinearity-in-data-science-c5f6c0fe6edf

How can we test if features are independent to each other?
You can check the linear dependencies between independent variable using pertubation test or vif.

1. **Linear regression** tries to find the best fit line that minimizes the mean square error(or any error function you choose). It tries to find a hyperplane that minimizes this error.
2. **Multicollinearity** mean that a feature is a linear function of one or more features. Mathematically, a set of variables is perfectly multicollinear if there exist one or more exact linear relationships among some of the variables. Fi = function(f1,f2...fn).
a. Multicollinearity makes it hard to interpret the relative importance of independent variables in explaining the variation(weight values become interpretable).
b. It can increase the variance of the coefficient estimates and make the estimates very sensitive to minor changes in the model.

How to design new features which are orthogonal - not close to existing info?

This can be done using additional data.
whether customers will purchase the product in the coming days?
say, you did not use customer search logs for prediction when creating the initial model. Additional extremely rich data required - multiple time searched in the past then prob of purchasing it is higher.

DL can handle Feature Engineering. But we can use DL algorithms only if we have a huge dataset. Which may not be possible in many situations. In such cases, we use Conventional ML algorithms with manual Feature Engineering. There are some more use cases like Audio processing. Google developed algorithms like TACOTRON, WAVENET uses Feature Engineering to process the raw input sound signal to Spectrogram before feeding it to models.
For more info please read https://medium.com/inside-machine-learning/feature-engineering-for-deep-learning-2b1fc7605ace

Calibration - Key idea

1. Instead of predicting class values directly for a classification problem, it can be convenient to predict the probability of an observation belonging to each possible class. Predicting probabilities allows some flexibility including deciding how to interpret the probabilities, presenting predictions with uncertainty, and providing more nuanced ways to evaluate the skill of the model. Predicted probabilities that match the expected distribution of probabilities for each class are referred to as calibrated. Calibration is a must if we want a probabilistic class-label as output i.e.,

P(Y_i|X_i), as calibration corrects these probabilities. If you are using a log-loss as a performance metric which needs the P(Y_i|X_i) values, calibration is a must. 2. The probabilities output by the models such as LR, naive bayes are often NOT well calibrated which can be observed by plotting the calibration plot as we discuss in the next video. Hence, we use calibration as a post-processing step to ensure that the final class-probabilities are well calibrated. If the calibration plot shows no need for calibration, we can skip it
The main idea is that, as the prediction probability is near to the actual probability value then we will say that it is well callibrated model. The probability that is given by any model is called predicted probability value.
Well calibrated classifiers are probabilistic classifiers for which the output of the predict_proba method can be directly interpreted as a confidence level. For instance, a well calibrated (binary) classifier should classify the samples such that among the samples to which it gave a predict_proba value close to 0.8, approximately 80% actually belong to the positive class. So if the probabilty is high then confidence is high, so the model with the higher probability values is better.
LogisticRegression returns well calibrated predictions by default as it directly optimizes log-loss. In contrast, the other methods return biased probabilities; with different biases per method: So all the models except Logistic Regression need to be calibrated including Deep learning models also.

1**. Calibration plots** are used in binary-classification cases. For multi-class settings, we can convert them into one-vs-rest binary classification tasks and apply calibration.
2. Calibration is a must if we want a probabilistic class-label as output i.e., P(Y_i|X_i), as calibration corrects these probabilities. If you are using a log-loss as a performance metric which needs the P(Y_i|X_i) values, calibration is a must.
3. Calibration can be applied on top of any binary classification algorithm.

According to the official sklearn's documentation:
We make use of CalibratedClassifier Mainly due to 2 reasons
1. To increase confidence in the predictions
2. Some models will give poor estimates of the class probabilities and some even do not support probability prediction. The calibration module allows us to better calibrate the probabilities of a given model or to add support for probability prediction.

CalibratedClassifier

When the data is passed and if , the model is not trained(fit), then you need to pass the model object(like LinearSVC(), SVC()) and should give an integer value for 'cv' argument so that it splits the train data into folds according to the value given for 'cv' and train the model.

If the model is already trained and if you just need the probabilistic estimates, then you need to use the value of cv='prefit' which means the model has already been fit and we need only the predictions.

==========

==========

Kaggle winner interviews --very good place to look at
https://medium.com/kaggle-blog
Sklearn suitable for big data?
https://stackoverflow.com/questions/17017878/is-scikit-learn-suitable-for-big-data-tasks

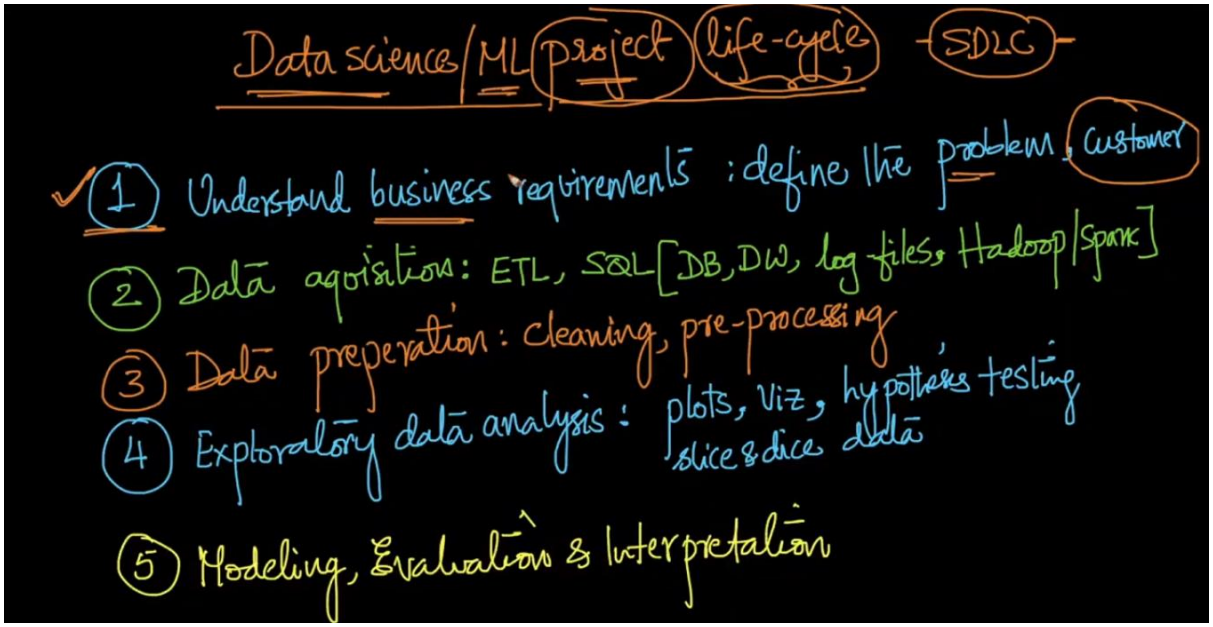
==========
Generally models like Gaussian NB, KNN, Linear Regression, Logistic Regression, SVM work better if the continuous numerical features in the given dataset are normally distributed. Algorithms like Decision Trees, RF, etc do not expect the features to follow Gaussian distribution

- Learn the topics under miscellaneous chapter except the ones related to deployment.
- Complete case studies.
- Come back and learn from the live sessions related to deployment.

If you don't have much time and bandwidth, first complete the essential part of course and assigments and then come back to live sessions.

==========

Data science / ML project Life Cycle



Case studies - ML
https://soundcloud.com/applied-ai-course/quora-question-pair

https://drive.google.com/drive/folders/1OWZoiQDvAvgOa-IUnEQ-6QKSEp_pw1XO

Performance Metric

1. If data is well balanced, both scores may give better interpretations. If your data is Heavily imbalanced, AUC interpretation may lead to wrong models.
2. You should use AUC when you ultimately care about ranking predictions and not necessarily about outputting well-calibrated probabilities.
3. You should not use it when your data is heavily imbalanced. It was discussed extensively in this article by Takaya Saito and Marc Rehmsmeier. The intuition is the following: the false-positive rate for highly imbalanced datasets is pulled down due to a large number of true negatives.
4. You should use it when you care equally about positive and negative classes. It naturally extends the imbalanced data discussion from the last section. If we care about true negatives as much as we care about true positives then it totally makes sense to use ROC AUC.

The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4349800/pdf/pone.0118432.pdf

FPR= FP / FP+TN
The lower FPR, the more negative points will be classified correctly. We are trying to reduce the FPR value in the AUC.

ROC Curves and Precision-Recall Curves for Imbalanced Classification

https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/

Temporal data : split

When we split the data by timestamps, we are simulating what our model will face in the production environment. For example, if we currently have a model that is trained a few days back before Trump was elected when there are very few or no questions about Donald Trump. After his election, there may be a surge of questions related to him the likes of which our model trained on older data has not encountered in its train data. This implies that our train and test data have a different distribution of words and entities and hence the model in production may not work well.we have to retrain our model again with the latest data available.
Cost of retraining is very high. We can go for online learning instead of retraining the entire data. Online learning is a method to train the model incrementally.

Quora ques pair similarity:
Checking the question semantically if they are same we try to find the text summarisation which is the famous NLP problem,next after text summarisation for each answer if they are semantically same we pick one and if not then merging is done based on grammar rules.

==========

word2vec - get nearest words

https://stackoverflow.com/questions/40074412/word2vec-get-nearest-words

we should consider the typos while processing text

For Example: "Capital" word has typo in Q2
Q1 - What is the capital of India?
Q2 - Cpaital of India?

https://pypi.org/project/pyspellchecker/

```
from spellchecker import SpellChecker

spell = SpellChecker()
```

```
Q1 = "What is the capital of India"
Q2 = "Cpaitl of India"

# find those words that may be misspelled
misspelled = spell.unknown(Q2.lower().split(" "))


for word in misspelled:
# Get the one `most likely` answer
print(spell.correction(word))

# Get a list of `likely` options
print(spell.candidates(word))
```

==========

good article --- Which model to use for your problem

https://blog.statsbot.co/machine-learning-algorithms-183cc73197c

https://towardsdatascience.com/ask-me-anything-session-with-a-kaggle-grandmaster-vladimir-i-iglovikov-942ad6a06acd

==========

PDF

https://www.khanacademy.org/math/statistics-probability/random-variables-stats-library/random-variables-continuous/v/probability-density-functions

==========

Building web apps for ML/AI using StreamLit

References: 1.https://www.streamlit.io/

2.https://docs.streamlit.io/en/latest/main_concepts.html

3. Code is taken from the official documentation and samples which can be found here:

https://docs.streamlit.io/en/stable/tutorial/create_a_data_explorer_app.html

https://github.com/streamlit/demo-self-driving

AI INDEX 2022 report

https://aiindex.stanford.edu/wp-content/uploads/2022/03/2022-AI-Index-Report_Master.pdf

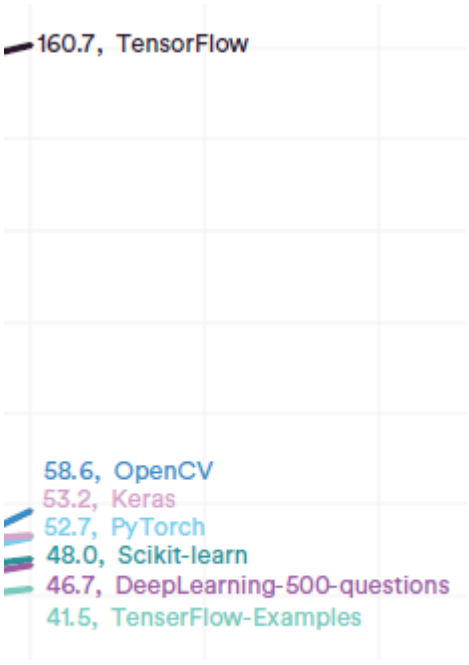
A 280 billion parameter model developed in 2021 shows a 29% increase in elicited toxicity over a 117 million parameter model considered the state of the art as of 2018.

Research on fairness and transparency in AI has exploded since 2014, with a fivefold increase in related publications at ethics-related conferences.Researchers with industry affiliations contributed 71% more publications year over year at AI ethics-focused conferences in recent years.

Since 2018, the cost to train an image classification system has decreased by 63.6%, while training times have improved by 94.4%. The trend of lower training cost but faster training time appears across other MLPerf task categories such as recommendation, object detection and language processing,

AI has not mastered complex language tasks, yet: AI already exceeds human performance levels on basic reading comprehension benchmarks like SuperGLUE and SQuAD by 1%–5%. Although AI systems are still unable to achieve human performance on more complex linguistic tasks such as abductive natural language inference (aNLI), the difference is narrowing.

TensorFlow remained by far the most popular in 2021, with around 161,000 cumulative GitHub stars—a slight increase over 2020. TensorFlow was about three times as popular in 2021 as the next-most-starred GitHub open-source AI software library, OpenCV, which was followed by Keras, PyTorch, and Scikit-learn.



technical progress in various subfields of artificial intelligence, including trends in computer vision, language, speech, recommendation, reinforcement learning, hardware, and robotics.

Computer Vision

image classification, object recognition, semantic segmentation, and face detection.

important real-world applications, such as autonomous driving, crowd surveillance, sports analytics, and video-game creation.

Image Classification - Image classification refers to the ability of machines to categorize what they see in images.

ImageNet is a database that includes over 14 million images across 20,000 categories publicly available to researchers working on image classification problems.

Top pretrained system was CoAtNets, produced by researchers on the Google Brain Team.

Image generation is the task of generating images that are indistinguishable from real ones. widely useful in generative domains where visual content has to be created, for example entertainment (companies like NVIDIA have already used image generators to create

virtual worlds for gaming), fashion (designers can let AI systems generate different design patterns), and healthcare (image generators can synthetically create novel drug compounds).

DEEPFAKE DETECTION

Many AI systems can now generate fake images that are indistinguishable from real ones.A related technology involves superimposing one person's face onto another, creating a so-called "deepfake."

HUMAN POSE ESTIMATION

Human pose estimation is the task of estimating different positions of human body joints (arms, head, torso, etc.) from a single image.

SEMANTIC SEGMENTATION

Semantic segmentation is the task of assigning individual image pixels a category (such as person, bicycle, or background)

autonomous driving (identifying which parts of the image a car sees are pedestrians and which parts are roads), image analysis (distinguishing

the foreground and background in photos.

FACE DETECTION AND RECOGNITION

In facial detection, AI systems are tasked with identifying individuals in images or videos. Although facial recognition technology has existed for several decades, the technical progress in the last few years has been significant. Some of today's top-performing facial recognition algorithms have a near 100% success rate on challenging datasets.

Facial recognition can be used in transportation to facilitate cross-border travel, in fraud prevention to protect sensitive documents, and in online proctoring to identify illicit examination behavior. The greatest practical promise of facial recognition, however, is in its potential to aid security.

OBJECT DETECTION

Object detection is the task of identifying objects within an image.

NLP

(1) English language understanding; (2) text

summarization; (3) natural language inference; (4) sentiment analysis; and (5) machine translation.

In the last decade, technical progress in NLP has been significant: The adoption of deep neural network–style machine learning methods has meant that many AI systems can now execute complex language tasks better than many human baselines.

SuperGLUE is a single-number metric that tracks technical progress on a diverse set of linguistic tasks.

The Stanford Question Answering Dataset (SQuAD) benchmarks performance on reading comprehension. F1 score

Text Summarization - Progress in text summarization is often scored on

ROUGE (Recall-Oriented Understudy for Gisting Evaluation).

Sentiment analysis - identify the sentiment (very negative, negative,

neutral, positive, very positive) of a given text

Machine Translation

A machine's translation capabilities are measured by the Bilingual Evaluation Understudy, or BLEU, score, which compares the extent to which a machine-translated text matches a reference human-generated translation. The higher the score, the better the translation.

2021 also saw the introduction of three open-source machine translation services (M2M-100, mBART, and OPUS).

Speech

Another important domain of AI research is the analysis, recognition, and synthesis of human speech. In this AI subfield, AI systems are typically rated on their ability to recognize speech and identify words and convert them into text; and also to recognize speakers and identify the individuals speaking.

Recommendation is the task of suggesting items that might be of interest to a user, such as movies to watch, articles to read, or products to purchase. Recommendation systems are crucial to businesses, such as Amazon, Netflix, Spotify, and YouTube.

In **reinforcement learning**, AI systems are trained to maximize performance on a given task by interactively learning from their prior

actions. Researchers train systems to optimize by rewarding them if they achieve a desired goal and then punishing them if they fail.

Systems experiment with different strategy sequences to solve their stated problem (e.g., playing chess or navigating through a maze) and select the strategies which maximize their rewards.Reinforcement learning can help autonomous vehicles change lanes, robots optimize manufacturing tasks, or time-series models predict future events.

Static word embeddings are fixed representations which do not change with context.Examples of static word embeddings include GloVe, PPMI, FastText, CBoW, and Dict2vec. In contrast, contextualized word

embeddings are dynamic representations of words that change based on the word's accompanying context. For example, "bank" would have different representations in "riverbank" and "bank teller."

Multilingual Word Embeddings

Large language models are often monolingual since they require a significant amount of text data to train. While English text can be easily sourced by scraping the internet, the challenge is greater with low-resource languages

Conferences –

Empirical Methods in Natural Language Processing(EMNLP)

NeurIPS, one of the largest AI conferences,

ICML

CVPR

ICCV

ICLR

AAAI

IROS

Deep Learning

Activation functions
Sigmoid : http://ronny.rest/blog/post_2017_08_10_sigmoid/
tanh :http://ronny.rest/blog/post_2017_08_16_tanh/

CNN - SOTA Arch
https://machinelearningmastery.com/review-of-architectural-innovations-for-convolutional-neural-networks-for-image-classification/

Interview question - Depp learning
https://www.youtube.com/watch?v=vwYmrKDvnlg

ReLU
Refer: http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf
https://en.wikipedia.org/wiki/Rectifier_(neural_networks)

Note that the input (x_i's) is standardized before being passed into a NN, which implies that some of the input-vectors values would -ve. If we now initialize the weights with only +ve weights, the chances of a dead ReLU would be more as we

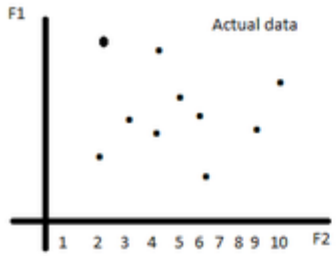
have some -ve negative values with all +ve weights. Hence, we initialize the weights to have both +ve and -ve weights randomly.
https://www.reddit.com/r/MachineLearning/comments/47fewl/initializing_weights_using_relu/

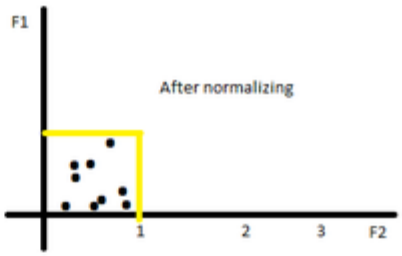
Most of the ML models work better if the numerical features follow gaussian distribution. Applying Column standardization makes the numerical features follow same scale with same mean and same standard deviation. Hence it is better to apply standardization than normalization as the models work better when the features have same mean and same standard deviation in most of the cases.
Column standardization makes the numerical features follow a distribution with mean=0 and standard deviation=1 (not gaussian all the time)
Every standardized distribution is not a gaussian distribution.

Naive Bayes when dealing with numerical features assumes all the features are gaussian. (Ref Gaussian Naive Bayes). And Logistic regression can also be proved mathematically that it is equivalent to Gaussian Naive Bayes.

1) What is standardization and normalization?
https://stats.stackexchange.com/questions/10289/whats-the-difference-between-normalization-and-standardization
2) the models work better when the features have same mean and same standard deviation in most of the cases"? what does this mean?

like in normalization we have a bounded range like all the values must be in range 0 to 1 while in standardization we donot have such boundary and even outliers doesnot effect us in standardization like if you look this all the weights in normalization are in a small box while in standardization they are spread across





Normalization rescales the values into a range of [0,1]. This might be useful in some cases where all parameters need to have the same positive scale. However, the outliers from the data set are lost.

$$X_{changed} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Standardization rescales data to have a mean (μ) of 0 and standard deviation (σ) of 1 (unit variance).

$$X_{changed} = \frac{X - \mu}{\sigma}$$

For most applications standardization is recommended.

When it comes to test data, how do we perform **batch normalization**?
For the training phase we calculate the batch mean and batch variance for every batch while on the test data are calculated using the population, rather than mini-batch, statistics(i.e. mean and the variance). Have a look at this:
https://imgur.com/Po2YMqC

You are asked why to use the squared loss and power 4 or modulus.
One of the major reason is squared loss upon derivative produces a single value of parameter set and hence gives one unique solution while others don't. However it is not true for one value of the parameter, the loss would be minimal. Loss can be minimal for multiple values also. Hence you can't tell that squared loss would only give you the least solution. Just because you have multiple minima doesn't mean that they need to be different in values. Honestly in Square and power 4 both generates minimal solution but the squared loss gives one solution while power 4 gives 3 solutions.
Imagine that you have a function y = x^2 + x + 3 and you find dy/dx, so it shall be 2x + 1. Now to calculate minima, you do 2x + 1 = 0 and hence x = -1/2. This means that for x = -1/2, this can be either minimum or maximum but if you would have had y = X^4+x^2+1 then dy/dx would have been 3x^3 + 2x = 0 or x(3x^2+2) = 0 => x = 0, or x = +sqrt(2/3) or x = -sqrt(2/3). Not mathematically we can see that 3 values are being minimum. Hence even functions without square too, there can be optimal solutions.

Training itself means trying to find out global minima.

Computationally complex problems are generally reffered as NP hard problems. Unfortunately, there is no way to know thr global minima without exhaustively looking for it. If there was a way, most of the world's hardest computational problems would be solved tomorrow.
https://en.wikipedia.org/wiki/NP-hardness

Stochastic Gradient Descent with Momentum

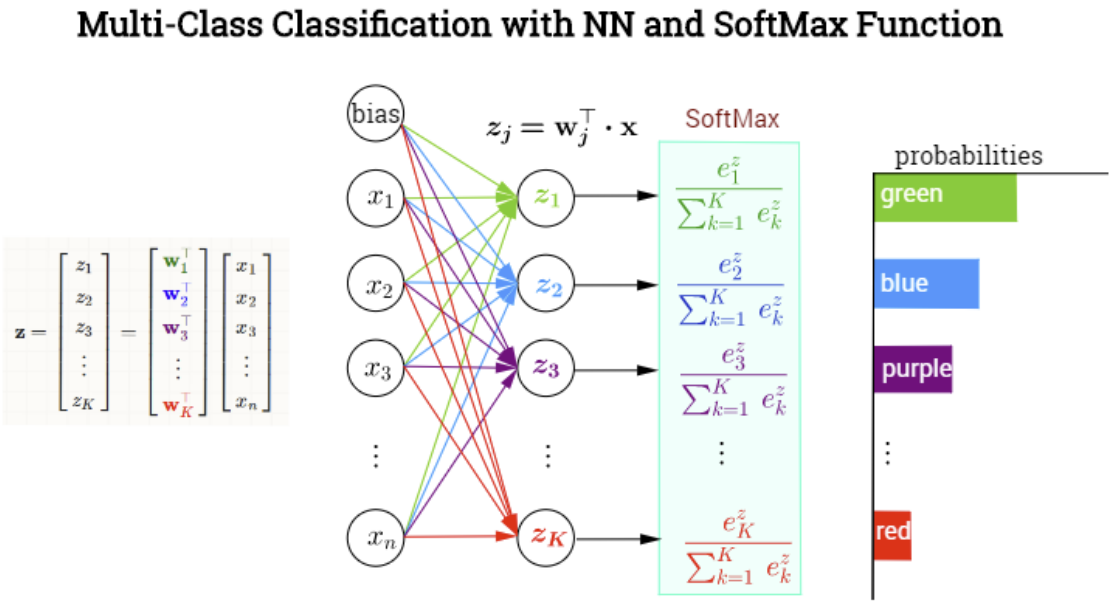
https://towardsdatascience.com/stochastic-gradient-descent-with-momentum-a84097641a5d

https://ruder.io/optimizing-gradient-descent/

ADAM: A METHOD FORSTOCHASTICOPTIMIZATION

https://arxiv.org/pdf/1412.6980.pdf

Softmax Classifier



proportional probability means assigning probability based on the values. Higher the value, higher will be the probability.

==========

AUTO ENCODER

==========

Nice tutorial

http://ufldl.stanford.edu/tutorial/unsupervised/Autoencoders/

1. In de-noising AE, we already have a robust-noise free data, then why we are adding noise to it from gaussian distribution. First, we have added noise to it then by AE we are again removing it. Then why to add noise when it ultimately has to be removed?

Ans : Denoising autoencoders can be trained on both noise-free or partially-noisy training-data. In the case of noise-free data, we are adding noise and building a representation robust to noise which could be observed in the future. This will help us make our AE work well on future test-points which may be partially-noisy. In practice, obtaining truly noise-free data is almost impossible.

If we train the data on partially-noisy data, we still add some small noise to all the points as we do not know which data points are actually corrupted and by what extent. Adding some noise to all training points is a good hack in this context.

Refer:https://en.wikipedia.org/wiki/Autoencoder#Denoising_autoencoder

2. L1 reg creates sparsity in weight vector that we got after solving logistic regression optimization problem, but here you are saying that L1 reg will create sparsity on the data

that we got after applying AE to reduce dimension. Is it correct?
Ans: I think we could improve our explanation of Sparse autoencoders. We will redo the sparse-autoencoders section of this video better. There are multiple ways of introducing sparsity to the x_i's (hidden unit activations). Following are some of them:
a. manually zero out the smallest (in absolute value) k-values of x_i'. This is this simplest and is called as k-sparse encoder.
Refer:https://en.wikipedia.org/wiki/Autoencoder#Sparse_autoencoder
b. Adding a constraint to the objective function based on probability distribution of the x_i's
Refer: Section 3 of https://web.stanford.edu/class/cs294a/sparseAutoencoder.pdf
c. Adding L1 reg on x_i's. Note that in the video we did NOT use L1 reg on weights but used L1-reg on x_i's where x_i'=W * x_i

Amazing power of Word vectors
https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/
Negative sampling is an algorithmic optimization to speed up Word2Vec computation. They are NOT different techniques. Target-word in this video is same as the context-word in Skip-gram based Word2Vec.

Reference material:
http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/

TensorFlow and Keras

https://medium.com/implodinggradients/tensorflow-or-keras-which-one-should-i-learn-5dd7fa3f9ca0

https://developers.google.com/machine-learning/crash-course/first-steps-with-tensorflow/toolkit

https://codelabs.developers.google.com/codelabs/cloud-tensorflow-mnist#0

https://cloud.google.com/blog/products/ai-machine-learning/learn-tensorflow-and-deep-learning-without-a-phd

DL models are also impacted by class imbalance. And yes, for imbalance dataset accuracy is not a right metric. In real world cases , we mostly use other metrics like auc score or f1 score in addition to accuracy.
it is recommended to add Dropout after batch normalization which works better.

Deep Learning is a powerful technique because the weights learned are more trustworthy,the backpropagation algorithm is the key to weight updation and due to different optimization techniques it's so successful

Library for Hyper-parameter tuning in Keras. It is very easy to learn. It's called Talos. Here's the link. - https://github.com/autonomio/talos
can we use this tool for hyperparameter tuning in the DL assignments?? Yes

CNN

Refer:https://www.youtube.com/watch?v=v20-E_2bT2c Refer: https://en.wikipedia.org/wiki/David_H._Hubel#Research Refer: http://www.cns.nyu.edu/~david/courses/perception/lecturenotes/what-where/what-where.html Refer: https://www.frontiersin.org/files/Articles/34845/fpsyg-04-00124-HTML/image_m/fpsyg-04-00124-g001.jpg

Refer: http://cs231n.github.io/convolutional-networks/ Refer: http://www.iro.umontreal.ca/~bengioy/talks/DL-Tutorial-NIPS2015.pdf

1. In the convolution operation, we multiply cell-wise a part of the input image and the kernel/filter to generate one-pixel/cell value in the output image. We apply ReLu to the output of the convolution operator before we place it in a cell/pixel of the output image. So, we perform: ReLU(Convolution(input-image, kernel))
2. It is not mandatory to use ReLU activation post convolution. It is not typically shown explicitly in most of the research papers as it creates more confusion as we perform element-wise ReLU.
3. The only difference between MLP and Convolution in terms of the optimization and weights part is that in MLP, we have vector multiplication as W.X after which we apply ReLu as ReLu(W.X). In convolution, we perform matrix-matrix convolution followed by ReLU as ReLU(Conv(Kernel, X)) and all the weights in the kernel are parameters which will be learned through optimization just like W is learned in MLP through optimization. Except that all of the math of back-prop is exactly the same.
4. We do not use ReLu's in max-pool-layer. ReLU is an optional part of the convolution layer.

Let say we have an input image of size 16*16. Then we applied convolution over it with padding = same and Strides = 1 with kernel size 4*4. Now this convolution operation will generate an output matrix of size 16*16 where each cell in a matrix is a pixel which is nothing but a real number. Let's call this output matrix as X. Now 16*16=256, so it means there are 256 numbers in matrix X. Now as per your reply what I have understood is that each of these 256 numbers of matrix X will pass through a single ReLU unit one by

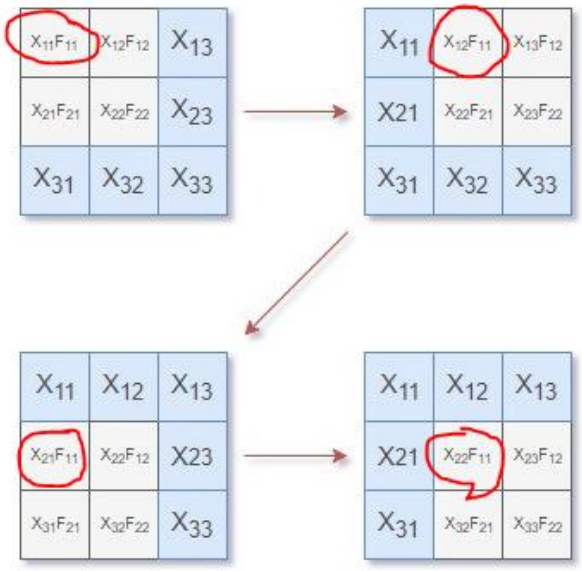
one and finally generate an output image of same size which is 16*16, right?
In a nutshell, after every convolution operation a matrix is generated and each cell value of that matrix pass through a single ReLU unit one-by-one and finally generate an output image of same size, here it is 16*16, right?

2. As per your reply what I have understood is that each cell value of a kernel/filter matrix is initialized in the same way as weights are initialized in MLP from he-normal or Xavier/Glorot initialization. In CNN also we use he-normal or Xavier/Glorot initializations for initializing cell values of kernel/filter matrix. And we keep on updating these kernel cell values using back-propagation. In short we learn kernel cell values similarly like weights in MLP??

3. So, after flatten operation does our CNN network works EXACTLY like MLP network with all the weights comes in place and ReLU is applied as ReLU(W*X)?? So, here also weights are initialized from initializers like he_normal, Xavier etc.

Forward and backpropagation in CNN:

https://medium.com/@2017csm1006/forward-and-backpropagation-in-convolutional-neural-network-4dfa96d7b37e



Now, to calculate the gradients of filter 'F' with respect to the error 'E', following equations needs to solved.

$$\frac{\partial E}{\partial F_{11}} = \frac{\partial E}{\partial O_{11}}\frac{\partial O_{11}}{\partial F_{11}} + \frac{\partial E}{\partial O_{12}}\frac{\partial O_{12}}{\partial F_{11}} + \frac{\partial E}{\partial O_{21}}\frac{\partial O_{21}}{\partial F_{11}} + \frac{\partial E}{\partial O_{22}}\frac{\partial O_{22}}{\partial F_{11}}$$

Backpropagation in CNN.
Refer: https://www.quora.com/How-are-the-parameters-of-max-pooling-represented-in-the-weights-nodes-of-a-neural-network

https://medium.com/@2017csm1006/forward-and-backpropagation-in-convolutional-neural-network-4dfa96d7b37e

https://becominghuman.ai/back-propagation-in-convolutional-neural-networks-intuition-and-code-714ef1c38199

LeNet
https://engmrk.com/lenet-5-a-classic-cnn-architecture/

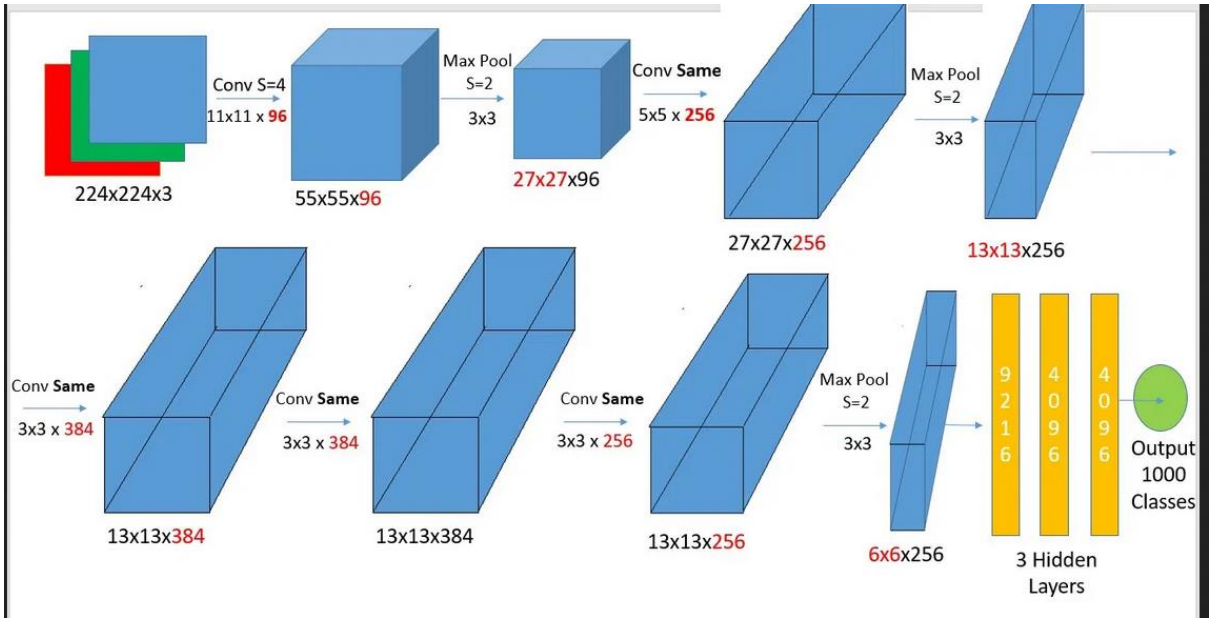
Refer slide 19
http://www.cs.cmu.edu/~10701/slides/10_Deep_Learning.pdf
#parameters = (5x5x3+1)x6 + (5x5x4 + 1) x 3 + (5x5x4 +1)x6 + (5x5x6+1)x1 = 1516

a)First 6 feature maps are connected to 3 contiguous input maps each (overlapping 2 maps) b) Second 6 feature maps are connected to 4 contiguous input maps (overlapping 3 maps) c) Next 3 feature maps are connected to 4 discontinuous input maps (overlapping 1 map) d) Last 1 feature map are connected to all 6 input maps.

5*5*3 represents each kernel of 5*5*3 dimensions, +1 represents addition of bias to each kernel/filter.
(5*5*3+1)*6--> here 6 represents number of kernels

KERAS CNN
Refer:https://keras.io/layers/convolutional/ Refer: https://keras.io/layers/pooling/ Refer: https://keras.io/layers/core/#flatten Refer: https://github.com/f00-/mnist-lenet-keras/blob/master/lenet.py

ALEXNET
Refer: https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf Refer: https://i0.wp.com/ramok.tech/wp-content/uploads/2017/12/2017-12-31_01h31_40.jpg Refer: http://euler.stat.yale.edu/~tba3/stat665/lectures/lec18/notebook18.html



Input size = 224
Kernel size = 11
strides = 4
padding = 0
Result = ((floor(224 - 11 + 2*0)/4) + 1) = 54.
In picture it is 55. But calculation is showing 54.

Actually, the original paper used 224X224 but did not mention anything about padding. Here is a very neat clarification of this from Stanford lecture notes: "If you read the actual paper it claims that the input images were 224x224, which is surely incorrect because (224 - 11)/4 + 1 is quite clearly not an integer. This has confused many people in the history of ConvNets and little is known about what happened. My own best guess is that Alex used zero-padding of 3 extra pixels that he does not mention in the paper." So, actually, the input size is 227x227 (with padding) and NOT 224x224. Unfortunately, many blogs and research papers do not mention this clearly and even we missed it until you asked about it. Now,(227 - 11)/4 + 1 = 55 and everything falls in place. Very importantly, it is always a good idea to ensure that we get an integer when we perform the division with stride-length (4 in the above example) so that we do not need to use floor at all. That way, the math is simpler to follow and debug when things go wrong.

Selecting Optimizer for training NN

1. AdaGrad penalizes the learning rate too harshly for parameters which are frequently updated and gives more learning rate to sparse parameters, parameters that are not updated as frequently. In several problems often the most critical information is present in the data that is not as frequent but sparse. So if the problem you are working on deals with sparse data such as tf-idf, etc. Adagrad can be useful.

2. AdaDelta, RMSProp almost works on similar lines with the only difference in Adadelta you don't require an initial learning rate constant to start with.

3. Adam combines the good properties of Adadelta and RMSprop and hence tend to do better for most of the problems.

4. Stochastic gradient descent is very basic and is seldom used now. One problem is with the global learning rate associated with the same. Hence it doesn't work well when the parameters are in different scales since a low learning rate will make the learning slow while a large learning rate might lead to oscillations. Also Stochastic gradient descent generally has a hard time escaping the saddle points. Adagrad, Adadelta, RMSprop, and ADAM generally handle saddle points better. SGD with momentum renders some speed to the optimization and also helps escape local minima better.

VGG

Refer: https://www.quora.com/What-is-the-VGG-neural-network
Refer: https://arxiv.org/pdf/1409.1556.pdf Refer: https://github.com/fchollet/deep-learning-models/blob/master/vgg16.py

Regularisation in VGG: The training is regularised by weight decay (the L2 penalty multiplier set to 5*10^?4) and dropout regularisation for the first two fully-connected layers (dropout ratio set to 0.5).

Residual Network / ResNet

Refer: https://arxiv.org/pdf/1512.03385.pdf

Refer: https://github.com/keras-team/keras/blob/master/keras/applications/resnet50.py

There are many interpretations for ResNet. Don't get confused with that. The ResNet is defined as follows xi+1 = xi + F(xi) , where xi is the input at layer i and xi+1 is the output we get after transforming the xi . Now, if we differentiate xi+1 with respect to xi we get I + d(F(xi)) where I is the identity matrix. Now, due to Identity matrix we have a stable gradient. Even though the gradient vanishes if we apply F, then the Identity matrix will take care of it.

irrespective of which activation function we have, using skip connections improves the gradient flow and therefore skip connections are preferred in case of deep neural network architectures where we might face the issue of low gradient flow.

Inception N/W

Refer: http://www.ashukumar27.io/CNN-Inception-Network/
Refer: https://github.com/keras-team/keras/blob/master/keras/applications/inception_v3.py
Refer: https://arxiv.org/pdf/1512.00567.pdf

Transfer Learning

someone with acces to large clusters of machine will train the model and save the weights. You can simply download the weights and initialize the model with those downloaded weights.
we do flattening to convert a matrix into a single 1D array
if you are planning to solve the segmentation type problem you should use the matrix as output, else if the problem is like classification type then go with the flattened layer output. Another way to flatten is to use Conv1D layers to flatten without using the flattening layer, this way we can preserve the spatial information it has learnt in the last layers.
1. In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. We can broadly divide image segmentation techniques into two types.
i. semantic segmentation

ii. instance segmentation
For more info, please read this article.
2. Spatial refers to space. So, what is space in images? Space represents the 2D plane(x-y) in images. Coming back to the question, 'What is spatial information in CNN?', for example in the first conv layer, it extracts spatial information like edges, corners, etc. and in another Conv layer it extracts spatial information like eyes, nose, etc. This is spatial information in images.

Convolutional neural networks, have internal structures that are designed to operate upon two-dimensional image data, and as such preserve the spatial relationships for what was learned by the model. Specifically, the two-dimensional filters learned by the model can be inspected and visualized to discover the types of features that the model will detect, and the activation maps output by convolutional layers can be inspected to understand exactly what features were detected for a given input image.

Generally, a large learning rate allows the model to learn faster, at the cost of arriving on a sub-optimal final set of weights. A smaller learning rate may allow the model to learn a more optimal or even globally optimal set of weights but may take significantly longer to train.
The learning rate is a hyper-parameter that controls how much you adjust the weights of your network. When you're using a pre-trained model based on CNN, it's smart to use a small learning rate because high learning rates increase the risk of losing previous knowledge.

Fine tuning is a process to take a network model that has already been trained for a given task, and make it perform a second similar task while **retraining** simply refers to re-running the process that generated the previously selected **model** on a new training set of data.
For 7 classes classifier problem but the images are GrayScale.Can I apply Transfer Learning of VGG16 Bottleneck Features? YES, Since VGG16 is trained on Imagenet with channel 3. You can stack your

image thrice to get channel 3 image and then can feed to pretrained model.
https://github.com/keras-team/keras/issues/12520
https://stackoverflow.com/questions/51995977/how-can-i-use-a-pre-trained-neural-network-with-grayscale-images

1. The size of kernel to be used is totally based on what work you wanted to achieve. Basically for most of the task 3*3 size is used generally. The number of kernels to be used again is the experimental thing. There is no science or concept or formula exist in world which can tell me what no of filters can we used for a layer in conv block. This is more type of experimental based approach.

2 We use these architectures to solve complex deep learning i.e recognition problems. We make their use and transfer learning is performed by using their weights. They are very important to most of the task we do.
3. Deep learning is not ML. It's a black box no-one knows why it even works ,interpretability is a long thought. We cannot extract the feature importance info as we do not know the feature it has learned.

Code Example : Cats vs Dogs using small data - Keras
Refer: https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html

Many DeepLearning researchers like Geoff Hinton do not like Maxpooling for invariances. Here is what he said: *"The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster."* While Capsule nets have been shown to work to some extent, they are not yet widely used as of 2018. Things might change in the near future as they always do in DL.
Capsule nets do achieve invariances as stated in the paper: *"When the capsule is working properly, the probability of the visual entity being present is locally invariant – it does not change as the entity moves over the manifold of possible appearances within the limited*

domain covered by the capsule" Refer: https://ai.stackexchange.com/a/4495

MNIST data
Refer:
https://github.com/keras-team/keras/blob/master/examples/mnist_cnn.py

https://drive.google.com/file/d/1I5kcAaQKEx0IwUNQvZdYkctwCcWFf81N open in Colab

LSTM

http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Within each LSTM cell there are 4 weights - W_f, W_i, W_c, W_o. By looking at the picture shown from colah's website, it looks like all 4 are concatenations of h_{t-1} and x_t. But the 4 weight vectors are different in the sense that they learn different weights(the weights are not shared).
Have a look at this: https://imgur.com/a/Ho2EkBm. The internal structure of LSTM can be broken down into 3 different gates, primarily the input, output and the forget gate.
1. Forget Gate(weights corresponding to W_f):
First, we have the forget gate(W_f). This gate decides what information should be thrown away or kept. Information from the previous hidden state(h(t-1)) and information from the current input(x(t)) is passed through the sigmoid function. Values come out between 0 and 1. The closer to 0 means to forget, and the closer to 1 means to keep.
2. Input Gate(weights corresponding to W_i):
To update the cell state, we have the input gate. First, we pass the previous hidden state(h(t-1)) and current input(x(t)) into a sigmoid function. That decides which values will be updated by transforming the values to be between 0 and 1. 0 means not important, and 1 means important. You also pass the hidden state(h(t-1)) and current input(x(t)) into the tanh function to squish values between -1 and 1 to help regulate the network. Then you multiply the tanh output with the sigmoid output. The sigmoid output will decide which information is important to keep from the tanh output.
3. Cell State
Now we should have enough information to calculate the cell state. First, the cell state gets pointwise multiplied by the forget vector. This has a possibility of dropping values in the cell state if it gets multiplied by values near 0. Then we take the output from the input gate and do a pointwise addition which updates

the cell state to new values that the neural network finds relevant. That gives us our new cell state(c(t)).
4. Output Gate
Last we have the output gate. The output gate decides what the next hidden state should be. Remember that the hidden state contains information on previous inputs. The hidden state is also used for predictions. First, we pass the previous hidden state and the current input into a sigmoid function. Then we pass the newly modified cell state to the tanh function. We multiply the tanh output with the sigmoid output to decide what information the hidden state should carry. The output is the hidden state. The new cell state and the new hidden is then carried over to the next time step.

To review, the Forget gate decides what is relevant to keep from prior steps. The input gate decides what information is relevant to add from the current step. The output gate determines what the next hidden state should be.

This is a good video explanation of internals of LSTM: https://www.youtube.com/watch?v=8HyCNIVRbSU.

GRU

Refer: https://www.slideshare.net/hytae/recent-progress-in-rnn-and-nlp-63762080
Refer: https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be

https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21
https://youtu.be/8HyCNIVRbSU

some common real world applications why we actually need to address long term dependencies.
These include a wide range of problems; from predicting sales to finding patterns in stock markets' data, from understanding movie plots to recognizing your way of speech, from language translations to predicting your next word on your iPhone's keyboard. Also music, just like text, is a sequence of notes (instead of characters), it can be generated as well by LSTM by taking into account the previously played notes (or combinations of notes). LSTMs can replicate a musical style and do some cool mash-ups of songs and genres. Check out the link to see more applications where LSTM is needed: https://en.wikipedia.org/wiki/Long_short-term_memory#Applications

The number of units in each layer of RNN is a hyperparameter that we need to tune. They're not dependent on the number of words in a sentence.

Number of words in sentence == Number of time a cell will be unfolded along time axis == Maximum length of sentence among all sentences **!=** number of units

Refer this for more.

IMDB Sentiment classification

https://drive.google.com/file/d/1iWpQBiZO95pfOWLdaG6qKwA27d_Q-EDg/

Derivation for the number of parameters in an LSTM layer: https://youtu.be/l5TAISVIYM0

GAN

Applications of GAN

https://jonathan-hui.medium.com/gan-some-cool-applications-of-gans-4c9ecca35900
GAN tricks and hacks
https://github.com/soumith/ganhacks
MNIST GAN code example
https://medium.datadriveninvestor.com/generative-adversarial-network-gan-using-keras-ce1c05cfdfd3
for more research/theory focussed
https://deepgenerativemodels.github.io/
Various GANs in Keras:
https://github.com/eriklindernoren/Keras-GAN

The objective of descriminator is to classify whether a given image is real or not. For that it needs to have fake images as well for training the model right?

2. The goal of the discriminator is to detect generated data, so the discriminative model is trained to minimise the final log loss whereas generative model is trained to maximize the log loss. This way we are improving both models one at a time

Note that the loss function for both generator and discriminator is same (log loss). Now look at this from generator perspective. If generator is powerful, then discriminator should not be able to distingush between the fake and real images. If it isn't able to distinguish, then the log loss will be more. So to make generator more powerful, we need to maximize the log loss. To make the discriminator more powerful we need to minimize the log loss. First we fix the discriminator and train the generator to maximize log loss. Now we fix the generator and train the discriminator to minimize log loss.
https://medium.com/@sanjay035/sketch-to-color-anime-translation-using-generative-adversarial-networks-gans-8f4f69594aeb

Encoder-Decoder Models

refer this , this , this and this and for code example this , this and this
The only difference (In a broad sense) between Ilya Sutskever and Cho's paper is that in the former model, the context vector is fed to the first lstm cell of decoder and in the later one, it is fed to each lstm cell of the decoder part.

Attention Based

1. We mentioned that Tx is a hyperparam, which is actually explained in a 2015 paper [https://arxiv.org/pdf/1508.04025.pdf] and not in the original 2014 Attention Models paper [https://arxiv.org/pdf/1409.0473.pdf] that we referred to in the above session. In the 2014 paper, Tx is the length of the whole input sentence.

2. We missed out one more minor equation which explains how S0 is computed in the decoder. S0 is the input left most input in the Decoder and its value is tanh(Ws * h1_backward_arrow). where Ws is a weight matrix and h1_backward arrow is the value of the output of the left most LSTM in the bi-directional LSTM in the encoder. The tanh() & Ws are simply equivalent to having a tanh activation fully-connected layer.

==========

1. I didn't understand the sum_(alpha_1i) = 1 and alpha_1i >=0 condition. Can you please throw some light on that?
2. For c2 , the connections h1,h2 and h3 are taken while for c3 , connections h2,h3 and h4 are taken. How do we shift the attention?

3. LSTM unit has three inputs, right? the word x_i_t , previous cell state c_t-1 and previous output h_t-1 , why are there only 2 inputs in encoder-decoder models we have seen so far?

1. we apply a softmax on the alpha values to produce the attention weights alpha1, alpha2, alpha3 ...alphan. The advantage of applying softmax is as below:
a) All the weights lie between 0 and 1, i.e., 0 < = alpha1, alpha2, alpha3, alpha4.. alphan < = 1
b) All the weights sum to 1, i.e., alpha1+alpha2+alpha3+ealpha4+.....alphan = 1
Thus we get a nice probabilistic interpretation of the attention weights. This means that while predicting the word at particular timestep, the decoder needs to put more attention on the states h1 and h2 (if
values of alpha1 and alpha2 are high) while ignoring the states h3, h4 and h5 (if the values of alpha3, alpha4 and alpha5 are very small).

2. the model first predicts a single aligned position pt for the current target word. A fixed window centered around the source position pt is then used to compute a context vector ct , a weighted average of the source hidden states in the window P are used to find attention weights. Please go through the paper for detail for local attention : https://arxiv.org/pdf/1508.04025.pdf

3. The word x_i_t , previous cell state c_t-1 and previous output h_t-1 - yes, correct. Here also we are providing the same by storing previous cell state and hidden state for every timestep that can be passed into next timestep.

YOLO v3 nObject Detection
==========
for a real time demo, have a look at this video by Joseph Redmon himself....
https://www.ted.com/talks/joseph_redmon_how_computers_learn_to_recognize_objects_instantly?language=en

==========

Conv2D Conv3D

Conv2D is generally used on Image data. It is called 2 dimensional CNN because the **kernel** slides along 2 dimensions on the data.

e.g Conv2D(1, kernel_size=(3,3), input_shape = (128, 128, 3)). Here Argument **input_shape** (128, 128, 3) represents (height, width, depth) of the image. Argument **kernel_size** (3, 3) represents (height, width) of the kernel, and kernel depth will be the same as the depth of the image.

In **Conv3D**, the **kernel** slides in 3 dimensions. Conv3D is mostly used with 3D image data. Such as **Magnetic Resonance Imaging** (MRI) data.

e.g Conv3D(1, kernel_size=(3,3,3), input_shape = (128, 128, 128, 3)) . Here argument **Input_shape** (128, 128, 128, 3) has 4 dimensions. A 3D image is also a 4-dimensional data where the fourth dimension represents the number of **colour channels**. Just like a flat 2D image has 3 dimensions, where the 3rd dimension represents colour channels. Argument **kernel_size** (3,3,3) represents (height, width, depth) of the kernel, and 4th dimension of the kernel will be the same as the colour channel.

So basically

- In **2D CNN**, kernel moves in **2** directions. Input and output data of 2D CNN is **3** dimensional. Mostly used on **Image** data.
- In **3D CNN**, kernel moves in **3** directions. Input and output data of 3D CNN is **4** dimensional. Mostly used on **3D Image** data (MRI, CT Scans).

Pretraining vs Taining

1 Pretraining is something a model is trained on for a general task, training is something more of fine tuning of model for specific task.
For better understanding follow this.
2.Layer normalization (Ba et al., 2016) was moved to the input of each sub-block, similar to a pre-activation residual network (He et al., 2016) and an additional layer normalization was added after the final self-attention block

Basics of Natural Language Processing(NLP):

1.Explain about Bag of Words?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bag-of-words-bow/)
Explain about Text Preprocessing: Stemming, Stop-word removal, Tokenization, Lemmatization.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/text-preprocessing-stemming-stop-word-removal-tokenization-lemmatization/)
Explain about uni-gram, bi-gram, n-grams.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/uni-gram-bi-gram-n-grams/)
What is tf-idf (term frequency- inverse document frequency)(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/tf-idf-term-frequency-inverse-document-frequency/)
Why use log in IDF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/why-use-log-in-idf/)

Explain about Word2Vec.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/word2vec/)
Explain about Avg-Word2Vec, tf-idf weighted Word2Vec?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/avg-word2vec-tf-idf-weighted-word2vec/)
Explain about Multi-Layered Perceptron (MLP)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/multi-layered-perceptron-mlp/)
How to train a single-neuron model?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/training-a-single-neuron-model/)
How to Train an MLP using Chain rule ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/training-an-mlp-2/)
How to Train an MLP using Memoization?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/training-an-mlp/)

Explain about Backpropagation algorithm?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/backpropagation/)
Describe about Vanishing and Exploding Gradient problem?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/vanishing-gradient-problem-2/)
Explain about Bias-Variance tradeoff in neural Networks?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bias-variance-tradeoff-23/)

Deep Learning:

1. What is sampled softmax?
2. Why is it difficult to train a RNN with SGD?
3. How do you tackle the problem of exploding gradients? (By gradient clipping)
4. What is the problem of vanishing gradients? (RNN doesn't tend to remember much things from the past)
5. How do you tackle the problem of vanishing gradients? (By using LSTM)
6. Explain the memory cell of a LSTM. (LSTM allows forgetting of data and using long memory when appropriate.)
7. What type of regularization do one use in LSTM?
8. What is the problem with sigmoid during backpropagation? (Very small, between 0.25 and zero.)
9. What is transfer learning?
10. What is backpropagation through time? (BPTT)
11. What is the difference between LSTM and GRU?
12. Explain Gradient Clipping.

13. Adam and RMSProp adjust the size of gradients based on previously seen gradients. Do they inherently perform gradient clipping? If no, why?

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

Reinforcement Learning (recorded)

https://web.stanford.edu/class/psych209/Readings/MnihEtAlHassibis15NatureControlDeepRL.pdf
https://ai.googleblog.com/2015/02/from-pixels-to-actions-human-level.html

DL case study -Self Driving car

Refer: https://github.com/commaai/research Refer: https://github.com/udacity/self-driving-car/tree/master/datasets Refer: http://data.apollo.auto/?locale=en-us&lang=en Refer: https://github.com/SullyChen/Autopilot-TensorFlow

NVIDIA's end to end CNN model.

Refer: model.py in zipped folder. Refer: https://arxiv.org/pdf/1604.07316.pdf Refer: https://devblogs.nvidia.com/deep-learning-self-driving-cars/

Refer: https://github.com/commaai/research

Music Generation using Deep Learning

This is artistic kind of case study that we covered to show/tell the capabilites of LSTMs even in creative industry like music.
However, the core lesson to be learned from this case study is: LSTM can be used to generate sequences.

You can create sequences of text(to create poems, write articles), or anything else too. You're just limited by dataset and your creativity.
Refer: https://folkrnn.org/ Refer: https://soundcloud.com/trivedigaurav/char-rnn-composes-long-composition Refer: https://soundcloud.com/sigur-ur-sk-li/neuralnet-music-1

https://medium.com/cindicator/music-generation-with-neural-networks-gan-of-the-week-b66d01e28200#:~:text=C%2DRNN%2DGAN%20is%20a,%2C%20for%20example%2C%20music%20files!

Music Representation
Refer: abc-notation: https://en.wikipedia.org/wiki/ABC_notation https://abcjs.net/abcjs-editor.html Refer: https://www.trivedigaurav.com/blog/machines-learn-to-play-tabla/ MIDI: https://towardsdatascience.com/how-to-generate-music-using-a-lstm-neural-network-in-keras-68786834d4c5

Generate Tabla Music

Refer: https://www.trivedigaurav.com/blog/machines-learn-to-play-tabla/ Refer: https://www.trivedigaurav.com/blog/machines-learn-to-play-tabla-part-2/

MIDI music generation
Refer: https://towardsdatascience.com/how-to-generate-music-using-a-lstm-neural-network-in-keras-68786834d4c5 Refer: https://github.com/Skuldur/Classical-Piano-Composer

Refer : https://medium.com/artists-and-machine-intelligence/neural-nets-for-generating-music-f46dffac21c0 Refer: https://magenta.tensorflow.org/

Linear Algebra

1. Define Point/Vector (2-D, 3-D, n-D)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/introduction-to-vectors2-d-3-d-n-d-copy-8/)
2. How to calculate Dot product and angle between 2 vectors?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/dot-product-and-angle-between-2-vectors-1/)
3. Define Projection, unit vector?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/projection-and-unit-vector-1/)
4. Equation of a line (2-D), plane(3-D) and hyperplane (n-D)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/equation-of-a-line-2-d-plane3-d-and-hyperplane-n-d-1/)
5. Distance of a point from a plane/hyperplane, half-spaces?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/distance-of-a-point-from-a-planehyperplane-half-spaces-1/)
6. Equation of a circle (2-D), sphere (3-D) and hypersphere (n-D)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/equation-of-a-circle-2-d-sphere-3-d-and-hypersphere-n-d-1/)
7. Equation of an ellipse (2-D), ellipsoid (3-D) and hyperellipsoid (n-D)?(https://www.appliedaicourse.com/course/applied-ai-course-

online/lessons/equation-of-an-ellipse-2-d-ellipsoid-3-d-and-hyperellipsoid-n-d-1/)

8. Square, Rectangle, Hyper-cube and Hyper-cuboid?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/square-rectangle/)

Probability And Statistics

1. What is Random variables: discrete and continuous?
2. Define Outliers (or) extreme points?.
3. What is PDF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/gaussiannormal-distribution-1/)
4. What is CDF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/introduction-to-correlation-and-co-variance-1/)
5. explain about 1-std-dev, 2-std-dev, 3-std-dev range?
6. What is Symmetric distribution, Skewness and Kurtosis?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/symmetric-distribution-skewness-and-kurtosis/)
7. How to do Standard normal variate (z) and standardization?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/standard-normal-variate-z-and-standardization/)
8. What is Kernel density estimation?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/kernel-density-estimation/)
9. Importance of Sampling distribution & Central Limit theorem.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/sampling-distribution-central-limit-theorem/)
10. Importance of Q-Q Plot: Is a given random variable Gaussian distributed?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/q-q-plothow-to-test-if-a-random-variable-is-normally-distributed-or-not/)
11. What is Uniform Distribution and random number generators(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/uniform-distribution-random-number-generators/)
12. What Discrete and Continuous Uniform distributions?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/uniform-distribution-and-its-parameters-pdf-and-cdf/)
13. How to randomly sample data points?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/uniform-distribution-random-number-generators/)
14. Explain about Bernoulli and Binomial distribution?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bernoulli-and-binomial-distribution/)
15. What is Log-normal and power law distribution?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/log-normal-distribution/)

16. What is Power-law & Pareto distributions: PDF, examples(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/power-law-distribution/)
17. Explain about Box-Cox/Power transform?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/box-cox-transform/)
18. What is Co-variance?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/co-variance/)
19. Importance of Pearson Correlation Coefficient?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/pearson-correlation-coefficient-3/)
20. Importance Spearman Rank Correlation Coefficient?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/spearman-rank-correlation-coefficient-3/)
21. Correlation vs Causation?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/correlation-vs-causation-3/)
22. What is Confidence Intervals?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/confidence-interval-c-i-introduction/)
23. Confidence Interval vs Point estimate?
24. Explain about Hypothesis testing?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hypothesis-testing-testing-methodology-null-hypothesis-p-value/)
25. Define Hypothesis Testing methodology, Null-hypothesis, test-statistic, p-value?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hypothesis-testing-testing-methodology-null-hypothesis-p-value/)
26. How to do K-S Test for similarity of two distributions?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-s-test-for-similarity-of-two-distributions-3/)

Dimensionality Reduction

1. What is dimensionality reduction? (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-is-dimensionality-reduction-1/)
2. Explain Principal Component Analysis?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intuition-of-pca/)
3. Importance of PCA?.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/why-learn-pca/)
4. Limitations of PCA?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/limitations-of-pca/)

5. What is t-SNE?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/t-distributed-stochastic-neighbourhood-embeddingt-sne-part-1/)
6. What is Crowding problem?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/crowding-problem-t-sne/)
7. How to apply t-SNE and interpret its output ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/how-to-use-t-sne-effectively/)

Performance Measurement Models:

1. What is Accuracy ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/accuracy-1/)
2. Explain about Confusion matrix, TPR, FPR, FNR, TNR?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/confusion-matrix-tpr-fpr-fnr-tnr-1/)
3. What do you understand about Precision & recall, F1-score? How would you use it?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/precision-and-recall-1/)
4. What is the ROC Curve and what is AUC (a.k.a. AUROC)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/receiver-operating-characteristic-curve-roc-curve-and-auc-1/)
5. What is Log-loss and how it helps to improve performance?.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/log-loss-1/)
6. Explain about R-Squared/ Coefficient of determination.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/r-squared-1/)
7. Explain about Median absolute deviation (MAD) ?Importance of MAD?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/median-absolute-deviation-mad-1/)
8. Define Distribution of errors?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/distribution-of-errors/)

Classification algorithms in various situations:

1. What is Imbalanced and balanced dataset.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/imbalanced-vs-balanced-dataset/)
2. Define Multi-class classification?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/multi-class-classification/)
3. Explain Impact of Outliers?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/impact-of-outliers/)

4. What is Local Outlier Factor?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/local-outlier-factor-simple-solution-mean-distance-to-knn/)
5. What is k-distance (A), N(A)(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-distanceana/)
6. Define reachability-distance(A, B)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/reachability-distanceab/)
7. What is Local-reachability-density(A)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/local-reachability-densitya/)
8. Define LOF(A)?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/local-outlier-factora/)
9. Impact of Scale & Column standardization?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/impact-of-scale-column-standardization/)
10. What is Interpretability?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/interpretability/)
11. Handling categorical and numerical features?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/handling-categorical-and-numerical-features/)
12. Handling missing values by imputation?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/handling-missing-values-by-imputation/)
13. Bias-Variance tradeoff?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bias-variance-tradeoff-3/)

K-NN(K Nearest Neighbour)

1. Explain about K-Nearest Neighbors?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-nearest-neighbors-geometric-intuition-with-a-toy-example-1/)
2. Failure cases of KNN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/failure-cases-of-knn/)
3. Define Distance measures: Euclidean(L2) , Manhattan(L1), Minkowski, Hamming?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/distance-measures-euclideanl2-manhattanl1-minkowski-hamming/)
4. What is Cosine Distance & Cosine Similarity?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/cosine-distance-cosine-similarity/)
5. How to measure the effectiveness of k-NN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/how-to-measure-the-effectiveness-of-k-nn/)
6. Limitations of KNN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/knn-limitations-1/)

7. How to handle Overfitting and Underfitting in KNN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/overfitting-and-underfitting/)
8. Need for Cross validation?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/need-for-cross-validation/)
9. What is K-fold cross validation?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-fold-cross-validation/)
10. What is Time based splitting?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/time-based-splitting/)
11. Explain k-NN for regression?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-nn-for-regression/)
12. Weighted k-NN ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/weighted-k-nn/)
13. How to build a kd-tree.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/how-to-build-a-kd-tree/)
14. Find nearest neighbors using kd-tree?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/find-nearest-neighbours-using-kd-tree/)
15. What is Locality sensitive Hashing (LSH)?(
16. Hashing vs LSH?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hashing-vs-lsh/)
17. LSH for cosine similarity?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/lsh-for-cosine-similarity/)
18. LSH for euclidean distance?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/lsh-for-euclidean-distance/)

Naive Bayes

1. What is Conditional probability?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/conditional-probability-1/)
2. Define Independent vs Mutually exclusive events?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/independent-vs-mutually-exclusive-events-3/)
3. Explain Bayes Theorem with example?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bayes-theorem-with-examples/)
4. How to apply Naive Bayes on Text data?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/naive-bayes-on-text-data/)
5. What is Laplace/Additive Smoothing?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/laplace-additive-smoothing/)

6. Explain Log-probabilities for numerical stability?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/log-probabilities-for-numerical-stability/)
7. In Naive bayes how to handle Bias and Variance tradeoff?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bias-and-variance-tradeoff/)
8. What Imbalanced data?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/imbalanced-data/)
9. What is Outliers and how to handle outliers?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/outliers/)
10. How to handle Missing values?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/missing-values/)
11. How to Handling Numerical features (Gaussian NB) (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/handling-numerical-features-gaussian-nb/)
12. Define Multiclass classification.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/multiclass-classification/)

Logistic Regression and Linear Regression

1. Explain about Logistic regression?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intuition-1/)
2. What is Sigmoid function & Squashing ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/sigmoid-function-squashing-1/)
3. Explain about Optimization problem in logistic regression. (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/mathematical-formulation-of-objective-function-1/)
4. Importance of Weight vector in logistic regression.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/weight-vector-1/)
5. L2 Regularization: Overfitting and Underfitting.(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/l2-regularization-overfitting-and-underfitting/)
6. L1 regularization and sparsity. (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/l1-regularization-and-sparsity/)
7. What is Probabilistic Interpretation: Gaussian Naive Bayes ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/probabilistic-interpretation-gaussian-naive-bayes-1/)

8. Explain about Hyperparameter search: Grid Search and Random Search ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hyperparameter-search-grid-search-and-random-search/)
9. What is Column Standardization.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/column-standardization/)
10. Explain about Collinearity of features?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/collinearity-of-features-1/)
11. Find Train & Run time space and time complexity of Logistic regression?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/testrun-time-space-and-time-complexity-1/)

Support Vector Machine

1. Explain About SVM? (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intution-1/)
2. What is Hinge Loss?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/loss-function-hinge-loss-based-interpretation-copy-8/)
3. Dual form of SVM formulation.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/dual-form-of-svm-formulation/)
4. What is Kernel trick.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/kernel-trick/)
5. What is Polynomial kernel.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/polynomial-kernel-copy-8/)
6. What is RBF-Kernel.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/rbf-kernel-copy-8/)
7. Explain about Domain specific Kernels. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/domain-specific-kernels-copy-8/)
8. Find Train and run time complexities for SVM?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/train-and-run-time-complexities-copy-8/)

Explain about SVM Regression. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/svm-regression-copy-8/)

Decision Trees

1. How to Building a decision Tree?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/geometric-intuition-axis-parallel-hyperplanes-1/)
2. What is Entropy? (https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/building-a-decision-treeentropy/)

3. What is information Gain ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/building-a-decision-treeinformation-gain/)
4. What is Gini Impurity?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/building-a-decision-tree-gini-impurity/)
5. How to Constructing a DT. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/building-a-decision-tree-constructing-a-dt/)
6. Importance of Splitting numerical features.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/building-a-decision-tree-splitting-numerical-features/)
7. How to handle Overfitting and Underfitting in DT?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/overfitting-and-underfitting-4/)
8. What are Train and Run time complexity for DT?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/train-and-run-time-complexity/)
9. How to implement Regression using Decision Trees?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/regression-using-decision-trees-2/)

Ensemble Models:

1. What are ensembles?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/what-are-ensembles/)
2. What is Bootstrapped Aggregation (Bagging) ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/bootstrapped-aggregation-bagging-intuition/)
3. Explain about Random Forest and their construction?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/random-forest-and-their-construction-2/)
4. Explain about Boosting?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/boosting-intuition/)
5. What are Residuals, Loss functions and gradients ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/residuals-loss-functions-and-gradients/)
6. Explain about Gradient Boosting?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/gradient-boosting/)
7. What is Regularization by Shrinkage?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/regularization-by-shrinkage/)
8. Explain about XGBoost?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/xgboost-boosting-randomization/)
9. Explain about AdaBoost?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/adaboost-geometric-intuition-2/)

10. How do you implement Stacking models?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/stacking-models/)
11. Explain about cascading classifiers. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/cascading-classifiers/)

Clustering:

1. What is K-means? How can you select K for K-means?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-means-algorithm/)
2. How is KNN different from k-means clustering?
3. Explain about Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/agglomerative-divisive-dendrograms/)
4. Limitations of Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/limitations-of-hierarchical-clustering/)
5. Time complexity of Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/time-and-space-complexity-3/)
6. Explain about DBSCAN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/dbscan-algorithm-2/)
7. Advantages and Limitations of DBSCAN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/advantages-and-limitations-of-dbscan/)

Recommender Systems and Matrix Factorisation.

1. Explain about Content based and Collaborative Filtering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/content-based-vs-collaborative-filtering-copy-5/)
2. What is PCA, SVD?(hWhat is K-means? How can you select K for K-means?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/k-means-algorithm/)
3. How is KNN different from k-means clustering?
4. Explain about Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/agglomerative-divisive-dendrograms/)
5. Limitations of Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/limitations-of-hierarchical-clustering/)
6. Time complexity of Hierarchical clustering?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/time-and-space-complexity-3/)

7. Explain about DBSCAN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/dbscan-algorithm-2/)
8. Advantages and Limitations of DBSCAN?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/advantages-and-limitations-of-dbscan/)ttps://www.appliedaicourse.com/course/applied-ai-course-online/lessons/matrix-factorization-pca-svd/)
9. What is NMF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/matrix-factorization-nmf/)
10. How to do MF for Collaborative filtering ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/matrix-factorization-for-collaborative-filtering/)
11. How to do MF for feature engineering.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/matrix-factorization-for-feature-engineering/)
12. Explain relation between Clustering And MF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/clustering-as-mf/)
13. What is Hyperparameter tuning. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/hyperparameter-tuning/)
14. Explain about Cold Start problem.?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/cold-start-problem/)
15. How to solve Word Vectors using MF?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/word-vectors-as-mf/)
16. Explain about Eigenfaces. ?(https://www.appliedaicourse.com/course/applied-ai-course-online/lessons/eigen-faces/)

External resources for interview

References: https://medium.com/acing-ai/salesforce-ai-interview-questions-acing-the-ai-interview-75e177c4734 https://medium.com/acing-ai/microsoft-ai-interview-questions-acing-the-ai-interview-be6972f790ea https://medium.com/acing-ai/apple-ai-interview-questions-acing-the-ai-interview-803a65b0e795 https://medium.com/acing-ai/amazon-ai-interview-questions-acing-the-ai-interview-3ed4e671920f https://medium.com/acing-ai/uber-ai-interview-questions-acing-the-ai-interview-9532794bc057 https://medium.com/acing-ai/steps-to-ace-the-ai-interview-part-1-298249080e59 https://medium.com/acing-ai/steps-to-ace-the-ai-interview-part-2-b25f91582f5f https://medium.com/acing-ai/google-ai-interview-questions-acing-the-ai-interview-1791ad7dc3ae https://medium.com/acing-ai/facebook-ai-interview-questions-acing-the-ai-interview-5982add0af55 https://www.analyticsvidhya.com/blog/2016/09/40-interview-questions-asked-at-startups-in-machine-learning-data-science/ https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/ 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://medium.com/acing-ai/adobe-ai-interview-questions-acing-the-ai-interview-ef7a8099110b
https://www.listendata.com/2018/03/regression-analysis.html
https://www.analyticsvidhya.com/blog/2017/10/svm-skilltest/
https://vitalflux.com/decision-tree-algorithm-concepts-interview-question
https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-tree-based-models/
https://www.analyticsvidhya.com/blog/2017/01/must-know-questions-deep-learning/
https://www.kdnuggets.com/tag/interview-questions

1-How we will decide no. of epochs
2-How we will decide which activation and optimizer need to use.
3-How we will know need to use regularization. or etc....

1) Determining the no of epoch is a hard task. Because it all depends on your validation loss. If the model overfits then we have to stop there.

So we can take any large epoch and can use:

Early stopping and **Model Checkpoint**.

Please refer to this answer.

Please go through this blog(under hyperparameter tuning section)

3) You have to see whether your model overfits or not.

Complex models such as deep neural networks are prone to overfitting because of their flexibility in memorizing the idiosyncratic patterns in the training set, instead of generalizing to unseen data.

MODULE 7 - UNSUPERVISED LEARNING/ CLUSTERING

DBSCAN

Refer: http://scikit-learn.org/stable/auto_examples/cluster/plot_dbscan.html#sphx-glr-auto-examples-cluster-plot-dbscan-py Refer: http://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

Revision Questions:

1. What is K-means? How can you select K for K-means?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3263/k-means-algorithm/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies
2.
3. How is KNN different from k-means clustering?
4. Explain about Hierarchical clustering?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3271/agglomerative-divisive-dendrograms/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies
5.
6. Limitations of Hierarchical clustering?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3275/limitations-of-hierarchical-clustering/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies
7.
8. Time complexity of Hierarchical clustering?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3274/time-and-space-complexity/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies
9.
10. Explain about DBSCAN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3282/dbscan-algorithm/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies
11.
12. Advantages and Limitations of DBSCAN?https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3284/advantages-and-limitations-of-dbscan/6/module-7-data-miningunsupervised-learning-and-recommender-systems-real-world-case-studies

Interview questions on Recommender Systems

Self Learning:

1. How would you implement a recommendation system for our company's users?(https://www.infoworld.com/article/3241852/machine-learning/how-to-implement-a-recommender-system.html)

2. How would you approach the “Netflix Prize” competition?(Refer http://blog.echen.me/2011/10/24/winning-the-netflix-prize-a-summary/)
3. ‘People who bought this, also bought…’ recommendations seen on amazon is a result of which algorithm?(Please refer Apparel recommendation system case study,Refer:https://measuringu.com/affinity-analysis/)

How do you overcome the Cold Start Problem, Explain end to end?

https://kojinoshiba.com/recsys-cold-start/

In K-means if certain points are having null value then how to do imputation for such values?

Same as all other algorithms. you can impute using some imputation techniques like mean, median... Handling missing values by imputation - Check

what does seeding refer to clustering?

In k-means, initially, we take k-centroids randomly (As you know different initializers will result in different centroids). These centroids are picked randomly based on the random number generator. The seed() method is used to initialize this generator. This generator needs a number to start with (a seed value), to be able to generate a random number. (By default, the random number generator uses the current system time.)

The seed number is any random integer given to points as initial value, like in clustering the centroids are given initial value.

If we have more than 100 categorical classification, which algorithm is best to resolve? and Explain

Logistic regression will be a better option in this case because it's fairly simple and don't take much time to train, its good from both train and run time complexity. Other models like SVM takes so much time to build.

If you have 10mill records with 100dimension each for a clustering task. Which algorithm will you try first and why ?

First we try with a simpler algorithm like K-means as the time complexity for Agglomerative and DBScan are high.

When would you use k means cluster and when would you use hierarchical cluster?

When we have circular shapes of data, then kmeans is said to work well.When we have large datasets, it's better to use kmeans instead of hiearchial as it is computationally very expensive.But when we want the reproducable results, we use the hiearchial clustering.

Netflix used which solution to achieve the prize money.

No, it is a collabarative based filtering only.

pls refer https://analyticsindiamag.com/collaborative-filtering-vs-content-based-filtering-for-recommender-systems/

On what basis would you choose agglomerative clustering over k means clustering and vice versa ?

refer this

Assuming a clustering model's labels are known , how do you evaluate the performance of the model?

for clustering we have dunn index , Silhouette index etc.

Recommender System (Eigen Faces)

Refer:https://bugra.github.io/work/notes/2014-11-16/an-introduction-to-unsupervised-learning-scikit-learn/ (alternate link)

Refer: http://scikit-learn.org/stable/auto_examples/decomposition/plot_faces_decomposition.html#sphx-glr-auto-examples-decomposition-plot-faces-decomposition-py

Truncated SVD

Refer:http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html
Refer:http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html
Refer: http://scikit-learn.org/stable/auto_examples/decomposition/plot_faces_decomposition.html#sphx-glr-auto-examples-decomposition-plot-faces-decomposition-py

AMAZON FASHION APPAREL RECOMMENDATION

how did you find there are duplicates in images with same colour with different sizes or same title with different colours is there any code ?

This Github Python script finds duplicate images using a perspective hash (pHash) to compare images. pHash ignores the image size and file size and instead creates a hash based on the pixels of the image. This allows you to find duplicate pictures that have been rotated, have changed metadata, and slightly edited.

https://github.com/philipbl/duplicate-images

for the answer to the question "can I remove the same size, same image but different name?"

https://medium.com/@urvisoni/removing-duplicate-images-through-python-23c5fdc7479e

for more info:

https://stackoverflow.com/questions/3383892/is-it-possible-to-detect-duplicate-image-files

Removing duplicates from apparels - here the problem is to recommend similar products right. if we have duplicates we end up recommending those duplicates to user which is of no use because the user is already looking at that item

If we use total data points then we have many color value and price data is null. Can we Put "NA" as a color value where color data is missing and mean of price data in price columns.

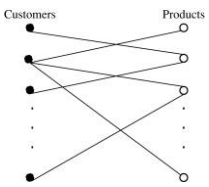
Imputation techniques work well when we have fewer missing values to impute. As most of the value in price column is empty, imputation won't be an effective strategy.

If we have too few (0.01%) or so of data points as missing values, then we can drop those rows . There is no thumb rule on percentage of error for which we can impute but typically if we have upto 30-40 % we can impute. If more than 30 -40% we can drop the feature.

https://towardsdatascience.com/6-different-ways-to-compensate-for-missing-values-data-imputation-with-examples-6022d9ca0779

Practical implementation of a bipartite graph is that they can be viewed as a relationship between customers and products. Could you please elucidate on this topic.

For example : The customer purchase behavior at *AllElectronics* can be represented in a *bipartite graph*. In a bipartite graph, vertices can be divided into two disjoint sets so that each edge connects a vertex in one set to a vertex in the other set. For the *AllElectronics* customer purchase data, one set of vertices represents customers, with one customer per vertex. The other set represents products, with one product per vertex. An edge connects a customer to a product, representing the purchase of the product by the customer.



May 18, 2021 23:48 PM

Stemming a lemmatized word is redundant if you get the same result than just stemming it, doing both stemming and lemmatization or only one will result in really SLIGHT differences. The choice between lemmatization and stemming highly depends on the task that you want to solve. Both approaches try to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. Before applying one of the approach know the principal difference between them: 1.Both stemmers and lemmatizers try to bring inflected words to the same form. 2.Stemmers use an algorithmic approach of removing prefixes and suffixes. The result might not be an actual dictionary word. Lemmatizers use a corpus. The result is always a dictionary word. 3.Lemmatizers need extra info about the part of speech they are processing. 4.Stemmers are faster than lemmatizers. So if speed is an issue use stemmer. Please, go through the link for more details.

Word2Vec

Every word can be represented as a vector so in pairwise distance it is finding the distance between these two vectors and then that is shown in the heatmap.

Metrics for Recommendation systems

https://towardsdatascience.com/evaluation-metrics-for-recommender-systems-df56c6611093#:~:text=Mean%20Average%20Precision%20at%20K,insights%20into%20a%20model's%20performance.

A/B Testing

Refer to these blogs:

https://www.invespcro.com/blog/ab-testing-statistics-made-simple/

https://neilpatel.com/blog/how-ab-testing-works/

16k_data_cnn_features.npy → contains already pretrained model vgg16, the file contains the weights of the model vgg16 and after training it on a huge number of images. using those weights we convert a given image into numerical vector. you will understand more about in the deep learning videos.

AB Testing In Real Life

https://towardsdatascience.com/ab-testing-in-real-life-9b490b3c50d1

How much time does a company takes to evaluate A/B results after the model goes in to production. A can go to different region and B can go to another

region. There can be a lot of possiblities since it is like a trial and error ? What would be the typical time frame to decide upon ?

It depends on the scale of the project you are working on. Here is a video (check from 13:30) from an Amazon engineer who says it will take usually a month to get all the feedback of the model performance.

Assume you have a model that gives 60% accuracy, Now you have redefined the architecture and tried a new model and say this is also giving accuracy near 60% or worse then we will not do A/B testing. We only want to deploy the new model when it outperforms the existing model. Ok, assume you get 75% accuracy on the new model, then you can try A/B testing to understand how the performance of the new model is on real time data. If the performance 75% is maintained then we slowly increase the number of users who are using the new model and again evaluate. So, if it is still working fine then you can deploy it for all the users else you can continue with your old model.

https://www.quora.com/Where-do-recommender-systems-fall-in-machine-learning-approaches

Exploratory Data Analysis:Sparse matrix representation

https://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.sparse.csr_matrix.html

I am working on private grocery store retails dataset, I have the bills in csv format of past three months of all customers, now I want to recommend each user their next month basket (basket containing items which are bought together at the start of the month by a specific user)

Or

I want find the common basket for most of user.

can you give me logical view how should I solve this problem?

For this problem,

1. Firstly, you can simply see repeating purchases for a customer by set intersection and surely suggest them.

2. Look at products that are bought together like bread and eggs by many customers using simple counts/frequency. If a customer only bought bread last month, you can suggest them eggs as your data suggest that bread and eggs exist in most purchases together. These simple approaches give you surprisingly good results.

3. If you have lots of data for many thousands of customers for many months, you can always build a model which has features for a customer and features for a product and y_i=1 if the customer buys this product and y=0 otherwise. You can now build a classification model here.

4. You can also compute customer-customer similarity using the customer-item matrix and find similar customers and group them into a set. Now, you can suggest most often purchase products in the set to all the customers in that set.

==========

About Application: It is an Enterprise Search Engine that contains only Legal documents format would be[PDF, Text, OCR]. Legal documents could be Database Search, Email Search, File Search these are indexed in ElasticSearch. For instance, if the public enter the query it will be feting the related documents via ElasticSearch.

Scope: I have to create an NLP Engine if any user enters the query NLP engine has to fetch the most related legal documents.

Doubt: I think if we do Normal Entity Extraction it won't work because data are indexed in ElasicSearch so we have to train entities based on ElasicSearch am I right? please correct me if I'm wrong.

Basically you are looking for information extraction.

check this

https://www.google.com/url?sa=t&source=web&rct=j&url=https://medium.com/%40venali/tutorial-series-on-nlp-information-extraction-tasks-99cd8309e2ef&ved=2ahUKEwj9ouC874joAhXZ7HMBHSQTDQkQFjABegQIAxAB&usg=AOvVaw0IfQMoUKFix13-pGOXyoLV

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.searchtechnologies.com/blog/natural-language-processing-techniques&ved=2ahUKEwj9ouC874joAhXZ7HMBHSQTDQkQFjAEegQIBhAB&usg=AOvVaw24pkCGpgv2OM0us2ELuqi6

Netflix Movie Recommendation system

Surprise library

Refer:http://surpriselib.com/

Refer: http://courses.ischool.berkeley.edu/i290-dm/s11/SECURE/a1-koren.pdf

lightfm is a very good library for recommendation systems as well. Have you guys had experience with that library? If so, how does it compare with surprise library?

https://lyst.github.io/lightfm/docs/home.html

Is it reliable to interpret this model's feature importance from the plot as shown in the video? The new feature 'bslpr' is equal to global avg + User Avg + Movie Avg that are already one among those 13 features. I thought with this kind of collinearity, interpreting feature importances (weights of features) would not be reliable and only 'forward feature selection' can be considered to interpret the model.

Could you kindly clarify

Ans

1. "Is it reliable to interpret this model's feature importance from the plot as shown in the video? " - yes

2. Using feature importance in random forest, when the dataset has two (or more) correlated features, then from the point of view of the model, any of these correlated features can be used as the predictor, with no concrete preference of one over the others. But once one of them is used, the importance of others is significantly reduced since effectively the impurity they can remove is already removed by the first feature. As a consequence, they will have a lower reported importance. This is not an issue when we want to use feature selection to reduce overfitting, since it makes sense to remove features that are mostly duplicated by other features.

Decision trees are by nature immune to multicollinearity. For example, if you have 2 features which are 99% correlated, when deciding upon a split the tree will choose only one of them. Other models such as Logistic regression would use both the features. so at each time we are calculating the Gini and based on these all Gini scores we are giving feature importance so you can use this feature importance scores.

Shrunk Correlation Coefficient is used to get smooth values. It is also mentioned that the Shrunk Correlation Coefficient is somewhat similar to Laplace smoothing. Laplace smoothing is used in Naive Bayes to make sure we handle the unknown categorical variables encountered during training correctly. It is also used to avoid overfitting when

only a few ratings are available. Here are more details: https://surprise.readthedocs.io/en/stable/similarities.html#surprise.similarities.pearson_baseline

Uij: the set of all users that have rated both items i and j

Iuv : the set of all items rated by both users u and v.

|Uij| refers to the number of elements in that set. Same is the case with |Iuv|

==========

Practice more on Python basics. Consistency is the key. You can solve these question for pandas and numpy.

==========

Case Study 14: Building a Smart Gym Assistant from scratch

Do Record this - **Module 6 – Machine Learning Real World case studies – TODO**
16 hrs 47 mins Done: 8.84%

Module 6 -

Case Study 1: Quora question Pair Similarity Problem

1. When to use PCA and When to use T-sne? Does it depend on the no of features?

2. If T-Sne performs average on a dataset,then does that mean PCA will also perform the same way?

Check these links :

https://stats.stackexchange.com/questions/238538/are-there-cases-where-pca-is-more-suitable-than-t-sne

When feature importance is important we do not use pca and if feature are large in no we usually use tsne

The interpretability of the features is lost by transforming the features from high to low dimensions by using PCA or t-SNE. For more info on this please read these blogs

https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c

https://medium.com/data-design/how-to-not-be-dumb-at-applying-principal-component-analysis-pca-6c14de5b3c9d

PCA is an unsupervised technique used to preprocess and reduce the dimensionality of high-dimensional datasets while preserving the original structure and relationships inherent to the original dataset. We transform each data point in the dataset hence we don't lose information regarding its class label.

Please refer this blog - https://medium.com/analytics-vidhya/pca-vs-t-sne-17bcd882bf3d#:~:text=t%2DSNE%20is%20also%20a,large%20pairwise%20distance%20maximize%20variance.&text=It%20takes%20a%20set%20of,it%20into%20low%20dimensional%20data.

CalibratedClassifierCV is used for computing probability estimates where some algorithms do not provide this feature.

Some algorithms also provide this feature of computing probability estimates, but couldn't give accurate figures. In order to get the accurate figures of the probability estimates, we use CalibratedClassifierCV.

https://machinelearningmastery.com/calibrated-classification-model-in-scikit-learn/

CalibratedCV is used to get probabilities of each output label. Using Logistic regression, we only get the output labels but not the probabilities. We are using log-loss as the performance metric, which requires probabilities.

you don't need to calibrate the Logistic regression model. But you should calibrate all other models including Deep Learning also.

As you can see in sklearn's documentation it has been clearly stated that: LogisticRegression returns well-calibrated predictions by default as it directly optimizes log-loss. In contrast, the other methods return biased probabilities; with different biases per method.

When performing classification you often want not only to predict the class label, but also obtain a probability of the respective label. This probability gives you some kind of confidence on the prediction. Some models can give you poor estimates of the class probabilities and some even do not support probability prediction. The calibration module allows you to better calibrate the probabilities of a given model, or to add support for probability prediction.

Well calibrated classifiers are probabilistic classifiers for which the output of the predict_proba method can be directly interpreted as a confidence level. For instance, a well calibrated (binary) classifier should classify the samples such that among the samples to which it gave a predict_proba value close to 0.8, approximately 80% actually belong to the positive class.

LogisticRegression returns well calibrated predictions by default as it directly optimizes log-loss. In contrast, the other methods return biased probabilities; with different biases per method:

Other models is Gaussian NB, Random Forest, Deep Learning technioques need to be calibrated.

In the sklearn implementation of logistic regression, it calculates the total average loss and then use gradient descent to minimize that loss. So, at each iteration we go over the entire dataset and calculate the total average loss and then apply gradient descent on it. So, if we have n iterations we would have made n updates to the parameters or weights (w = w-lr*dw)

Whereas in SGD, we don't go over the entire dataset in a single iteration and calculate the loss. We split our dataset into batches. Let's assume we have 100 points and splitted our dataset into 10 partitions where each partition contains 10 points. Let's assume the iteration number is 1. In this iteration, we calculate the loss in the first partition and then apply gradient descent on it. Again we calculate the loss for the remaining partitions individually and then apply gradient descent. So, for a single iteration we are making 10 updates. For n iterations , we would make 10*n updates. This is more than the updates made by the simple gradient descent. Please refer here https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3040/sgd-algorithm/3/module-3-foundations-of-natural-language-processing-and-machine-learning

1. In general, linear SVM and logistic regression perform very similarly. While logistic regression tries to penalize all the points, linear SVM only penalizes points which lie inside the margin(this way it penalizes the points which are harder to classify). Have a look at this thread where they compare the linear SVM with logistic regression: https://stats.stackexchange.com/questions/95340/comparing-svm-and-logistic-regression.

2. For time complexity take a look at this: https://www.researchgate.net/post/what_is_Time_complexity_for_SVM_and_logistic_regression.

refer examples here here we can build model in two ways. 1) fit the train data to base model than use this in calibrated CV and again fit calibrated CV with CV data. i.e example 2. 2) we define base model and use it in calibrated CV and fit data to calibrated CV only i.e example 1. point2) once refer 'Pintas' comment here

—-----

If error in both train, test is high so it is underfit .

reference for the same:

https://datascience.stackexchange.com/questions/361/when-is-a-model-underfitted/628#628.

Linear models are less powerfull hence less overfit they are hence high biased.

Suppose let's say AUC is our metric and we got a train auc of 0.58 and test auc of 0.55 , although the difference between them is small, their score is very low. Hence we can conclude that the model is underfitting. In simpler terms, low scores of train data will result in model underfitting.

Case study 4:Taxi demand prediction in New York City

https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

FFT Plot

points which are of highest displacement and their distance from the mean position is called amplitude refer to

https://www.youtube.com/watch?v=_R1OBhXROzQ&ab_channel=Clapp

Note that we want to predict the number of pickups for a clusterID at a given time, t. So, we can use the data till t-1 even in 2016 to compute the ratios and use them to predict the number of pickups at time t for a given clusterID. In this case, we are using all the data, including 2016, till (t-1) as our train data to predict the value at time, t.

please refer this:https://www.researchgate.net/post/Is_there_a_cut-off_point_for_the_mean_absolute_percentage_error_MAPE

Malware detection Case study

Please click here.

Machine Learning design: Search engine for Q&A

https://www.youtube.com/watch?v=J_0ms1ayqKE

References: 1. ElasticSearch+ BERT: https://www.elastic.co/blog/text-similarity-search-with-vectors-in-elasticsearch 2. FAISS: https://engineering.fb.com/data-infrastructure/faiss-a-library-for-efficient-similarity-search/ 3. DiskANN: https://www.microsoft.com/en-us/research/publication/diskann-fast-accurate-billion-point-nearest-neighbor-search-on-a-single-node/

ML System Design: Feature Store

https://www.youtube.com/watch?v=ZxHo9WGn6KQ

References: https://medium.com/data-for-ai/comprehensive-and-comparative-list-of-feature-store-architectures-for-data-scientists-and-big-data-86ea8c4d853b

There are no specific classification here it depends on problem at hand. We discussed some design approaches here and here and here refer company blogs like refer "ML systems at big companies" here

BIG DATA and ML

can you give me best resources(videos) to learn bigdata concepts(hadoop,hive,etc)..

First, cover the live sessions that we've conducted related to big data on topics like spark, mlLib, general big data.

And then you can go with this book and this playlist.

what was the fundamental difference that didn't allowed hadoop to store intermediate data into RAM but spark??

Hadoop was released in 2006, at that time size of RAM were low as compared to today. So, data was stored into disk instead of RAM.

what is difference between spark sql and pyspark.

https://towardsdatascience.com/pyspark-and-sparksql-basics-6cb4bf967e53

LIVE: An overview of AI Algorithms (a 10,000 feet view)

https://www.youtube.com/watch?v=ocSnzl7r_wM

LIVE: Big-Data & Cloud Storage for ML/AI Applications

https://www.youtube.com/watch?v=WUTEK2tqlO4

Introduction to Spark for Data Science and Machine Learning [Recorded Live Session]

https://www.youtube.com/watch?v=UD3zgYZ4gi4

==========

Motivation to complete course reply
https://soundcloud.com/applied-ai-course/utkarsh-comment/s-ayYy9
--- consistently **2 hrs in a day** consistently - avoid mobile/youtube

Please share you code related queries to team@appliedaicourse.com

==========

Bishop's "Pattern Recognition and Machine Learning" is a great resource for an ML and General DS foundation.

****

https://github.com/vertika6/Sparks-Foundation/blob/main/Task2TheSparksFoundation.ipynb

---

Python library to quickly build BI analytics dashboards on top of any data sources with a few lines of code directly from Jupyter notebook.

A Python notebook instead of Excel!

atoti is great for the last mile of analytics, you can easily slice and dice your data, create and filter interactive charts, pivot tables, and web applications.

Run analysis against millions of data points and share the results with peers.

---
pip install atoti[jupyterlab]

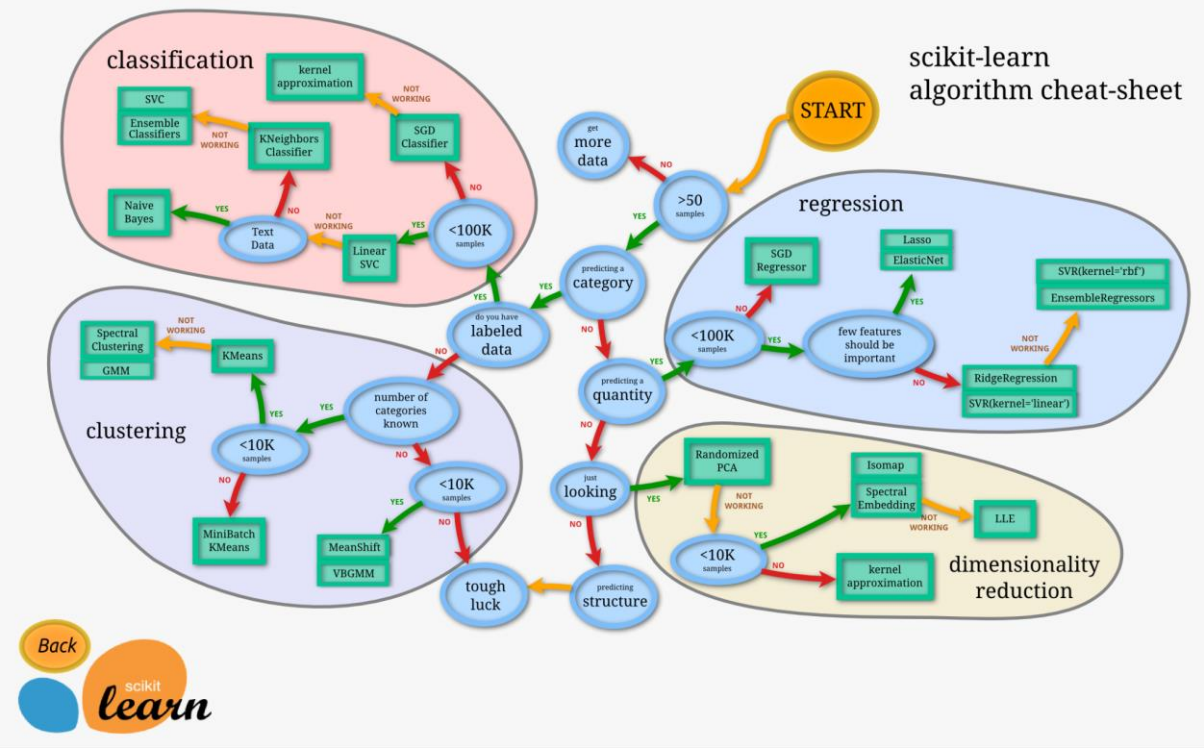
==========

Why not use Deep learning for every problem?

https://www.youtube.com/watch?v=AYzv3QmCE7s&list=PLupD_xFct8mHiNvSNO6AuWqCV9l_W2nB4&index=13

Choosing the right estimator

https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html



==========

```
# How to read others code :

# 1. High level Scan --- Understand logical modules. Donot panic!

# 2. Go line by line --- API Reference + Python basics are key

# 3. Modify the code and see what happens ---- Experiment + Patience


 https://ramsane.github.io/create-dataset/

# Create datasets in 2D of any shape to experiment.
```

scikit learn examples

https://scikit-learn.org/stable/auto_examples/index.html

Heuristic Algorithm

1. The term heuristic is used for algorithms that find solutions among all possible ones, but they do not guarantee that the best will be found, therefore they may be considered as approximately and not accurate algorithms.

2. Heuristic algorithms oftentimes used to solve NP-complete problems, a class of decision problems. In these problems, there is no known efficient way to find a solution quickly and

accurately although solutions can be verified when given. Heuristics can produce a solution individually or be used to provide a good baseline and are supplemented with optimization algorithms. Heuristic algorithms are most often employed when approximate solutions are sufficient and exact solutions are necessarily computationally expensive.

==========

1. Read Seminal Papers

2. Take Advantage of arXiv Sanity

3. Subscribe to Papers with Code

Refer this blog for more info.

BLOGS

https://machinelearningmastery.com/

https://www.kaggle.com/

https://ruder.io/

to check this --

The Algorithms - Python

https://github.com/TheAlgorithms/Python

1. All algorithms implemented in Python

By: The Algorithms

https://lnkd.in/guaDFtA

2. Playground and Cheatsheet for Learning Python

By: Oleksii Trekhleb

https://github.com/trekhleb/learn-python

3. Learn Python 3

By: Jerry Pussinen

https://lnkd.in/gRm8xYs

Also, check out OpenSpotlit, they share many interesting projects daily.

Day1:https://lnkd.in/grJyMGN

Day2:https://lnkd.in/ghGrHtd

Day3:https://lnkd.in/gfZeN6z

Day4:https://lnkd.in/g3KpvmN

Day5:https://lnkd.in/gfvsCbj

Day6: https://lnkd.in/gh47ysM

==========

Excited to be building up a data science library! Lots to learn from these great authors.

Data Points - Nathan Yau

Visualize This - Nathan Yau

Story Telling with Data - Cole Nussbaumer Knaflic

100 Page Machine Learning Book - Andriy Burkov

Machine Learning - Tom Mitchell

Hands-On ML with Scikit-Learn, Keras, and TensorFLow - Aurelien Geron

Fundamentals of Database Systems - Elmasri, Navathe

Designing Data Intensive Applications - Martin Kleppmann

Rise of the Data Cloud - Steve Hamm, Frank Slootman

Build a Career in Data Science - Emily Robinson, Jacqueline Nolis

Data Science for Business - Foster Provost

Machine Learning for Algorithmic Trading - Stefan Jansen

PMBOK - Because everyone needs to know the language of project managers

Hitchikers Guide to the Galaxy - Because everyone needs a break, and this my favorite book.

JD --

Create data flow automation using tools such as Azkaban and Kubeflow

Perform ETL with massive data from multiple applications and 300 M+ customers

---

I would also like to express my gratitude to all Youtube content creators - Nicholas White, Gaurav Sen, Raj Vikramaditya (Take u forward), Rachit Jain, Aditya Verma, Love Babbar, and YK Sugi for their valuable content.

Here's an ML strategy to try.

When you have a problem to solve, build two solutions. One is a deep Bayesian transformer running on multicloud Kubernetes, the other is a SQL query built on a stack of egregiously oversimplifying assumptions. Put one on your resume, and the other in production. Everyone goes home happy.

The first model I developed and deployed at AWS was a logistic regression that was directly implemented in SQL with hard-coded coefficients. It took about a week to develop/deploy and started immediately adding value far above what existed before, which were a couple outdated rules. That not only bought me time, but gave me opportunity to study where the simple model was failing and therefore where to focus my efforts. The model that replaced it was better, but it took months to develop and tons of infrastructure and engineering work to orchestrate everything. -- Senior ML Research scientist at AWS

==========

https://streamlit.io/

Kubernetes, Kuberflow, Docker

v. Imp. → How can ML be applied to current projects?

==========

https://opensearch.org/

JIRA

Cognitive computing

https://www.youtube.com/watch?v=NsID1iM8gRw

https://probmods.org/

https://arxiv.org/pdf/1604.00289.pdf Building Machines That Learn and Think Like People

https://www.thedrive.com/tech/43439/where-the-hell-are-the-robotaxis-we-were-promised

Liz Spelke

Dark, Beyond Deep: A Paradigm Shift to Cognitive AI with Humanlike Common Sense

https://reader.elsevier.com/reader/sd/pii/S2095809920300345?token=301151E5B6A867F61CF15DE204B4A8D0C7C31FBABDFBC015BF8BDC4336E184CC0667D5DA8669530F8ED8D693F21855C0&originRegion=eu-west-1&originCreation=20220411042637

functionality, physics, intent, causality, and utility (FPICU) as the

five core domains of cognitive AI with humanlike common sense. power of this perspective to develop cogni-tive AI systems with humanlike common sense by showing how to observe and apply FPICU with little training data to solve a wide range of challenging tasks, including tool use, planning, utility inference, and social learning.

—----------

The Easiest Way to Deploy Your ML/DL Models in 2022: Streamlit + BentoML + DagsHub

The future of search retrieval is hybrid. Combining AI-powered vector search with unsupervised sparse text search methods.

We know that semantic search, using AI-learned vector representations built on NLP Transformer models, can really shine in-domain, beating sparse methods like BM25 by a large margin. This is a fact.

But when applying semantic search using these learned vector representations outside of the domain they were trained on, they underperform compared to unsupervised simple methods like BM25. This is still the case in 2022.

For data-rich domains, with plenty of interaction data, it's possible to build and train representations that outcompete unsupervised sparse methods. Given that you have a data science team with competence in deep learning.

What if you don't have a data science team to build dense vector representation models? Or if you don't have interaction data as interactions with your search or recommendation system are noisy because the baseline model sucks. What to do then?

And what if you start out fresh, building a new search system with zero interactions to train a model on - where do you start then building a baseline search system? Your best shot might be sparse BM25.

You can also turn to a generic representation model, trained with open data from multiple domains. One example of such an embedding model is built by

@Nils_Reimers

Or you can combine the two approaches to get the best of both sparse and dense using a hybrid search: Dense offers relaxed semantic matching, avoid vocabulary mismatch and never return 0 results. Sparse offers exact matching, handling phrase search, and more.

Combining sparse and dense works much better than both sparse and dense alone when applied to a new domain. The two methods are complementary.

Our paper "Out-of-Domain Semantics to the Rescue! Zero-Shot Hybrid Retrieval Models" (by @tao_chen @_Mingyang_Zhang Jing Lu @bemikelive @marc_najork; To appear in ECIR, 2022) is now on arXiv: https://arxiv.org/abs/2201.10582

With hybrid retrieval, you can combine the best of these two techniques. And if you use http://Vespa.ai for search serving, you don't need to manage different technology stacks. Vespa supports both sparse and dense, or hybrid, the future of search

Hybrid is already standard practise for the simple reason that the AI-power (i.e., neural retrieval) models are so computationally expensive that you need to do stage 1 retrieval with a 'cheap' sparse model (BM25) followed by a smaller rerank with the expensive neural model.

phd application

http://www.ipu.ac.in/Pubinfo2022/adm22brphd030322.pdf

Arvix Sanity Preserver, Two Minute Papers, and discord channels can help! Also being focused on a specific area of ML (NLP for me) helps tremendously.

ML interview advice and resources

https://forums.fast.ai/t/wiki-ml-interviews-resources-advice-s/70528

`For Sigmoid activation fn - Glorot/Xavier weight initialization is used.`

For Relu - He Normal weight initialization is used

the long term goal is to build AGI that loves people the way parents love their children

Yann LeCun

@ylecun

· May 23, 2021

Replying to @iamtemibabs

It does not exist. Even human intelligence is highly specialized. We can talk about human-level AI. But AGI is nonsense.

Artificial Intelligence Meets Mental Health Therapy | Andy Blackwell | TEDxNatick

Depression and Anxiety are everywhere.

4 free sites for finding remote dev jobs: -

https://remoteok.com/

https://weworkremotely.com/

https://ai-jobs.net/

https://www.nocsdegree.com/jobs/

https://arc.dev/

-> remoteok .io -> freelancer .com -> remotive .io -> remoteglobal .com -> devsnap .io -> working nomads .co -> triplebyte .com -> nodes .co -> epic jobs .co -> remotehunt .com -> weworkremotely .com -> flexjobs .com

[1] http://remoteok.com [2] http://weworkremotely.com [3] http://ai-jobs.net (AI/ML and Big Data) [4] http://nocsdegree.com/jobs [5] http://talent.hubstaff.com [6] http://jobspresso.co/remote-software-jobs

[7] http://workingnomads.com/remote-development-jobs [8] http://remote.co/remote-jobs/developer [9] http://arc.dev (companies apply to you) [10] http://freelancer.com (for freelancing) [11] http://remotive.com [12] http://triplebyte.com/jobs/l/remote [13] http://epicjobs.co

[14] http://remotehunt.com [15] http://flexjobs.com [16] http://trueup.io [17] http://jobboardsearch.com [18] http://nowhiteboard.org (jobs without whiteboarding interviews) [19] http://remote3.co (web3 jobs) [20] http://angel.co (jobs at startups)

[21] http://remoteleads.io [22] http://justremote.co/remote-developer-jobs [23] http://jsremotely.com (JavaScript jobs) [24] http://otta.com [25] http://splash.ripplematch.com (US-based)

[26] http://interviewquery.com/jobs (data jobs) [27] Twitter [28] LinkedIn [29] http://jobboardsearch.com/remote-jobs (a list of remote job boards)

Here are 29 free sites for doing exactly that 👇

[1] remoteok.com

[2] weworkremotely.com

[3] ai-jobs.net (AI/ML and Big Data)

[4] nocsdegree.com/jobs

[5] talent.hubstaff.com

[6] jobspresso.co/remote-softwar…

[7] workingnomads.com/remote-develop…

[8] remote.co/remote-jobs/de…

[9] arc.dev (companies apply to you)

[10] freelancer.com (for freelancing)

[11] remotive.com

[12] triplebyte.com/jobs/l/remote

[13] epicjobs.co

[14] remotehunt.com

[15] flexjobs.com

[16] trueup.io

[17] jobboardsearch.com

[18] nowhiteboard.org (jobs without whiteboarding interviews)

[19] remote3.co (web3 jobs)

[20] angel.co (jobs at startups)

[21] remoteleads.io

[22] justremote.co/remote-develop…

[23] jsremotely.com (JavaScript jobs)

[24] otta.com

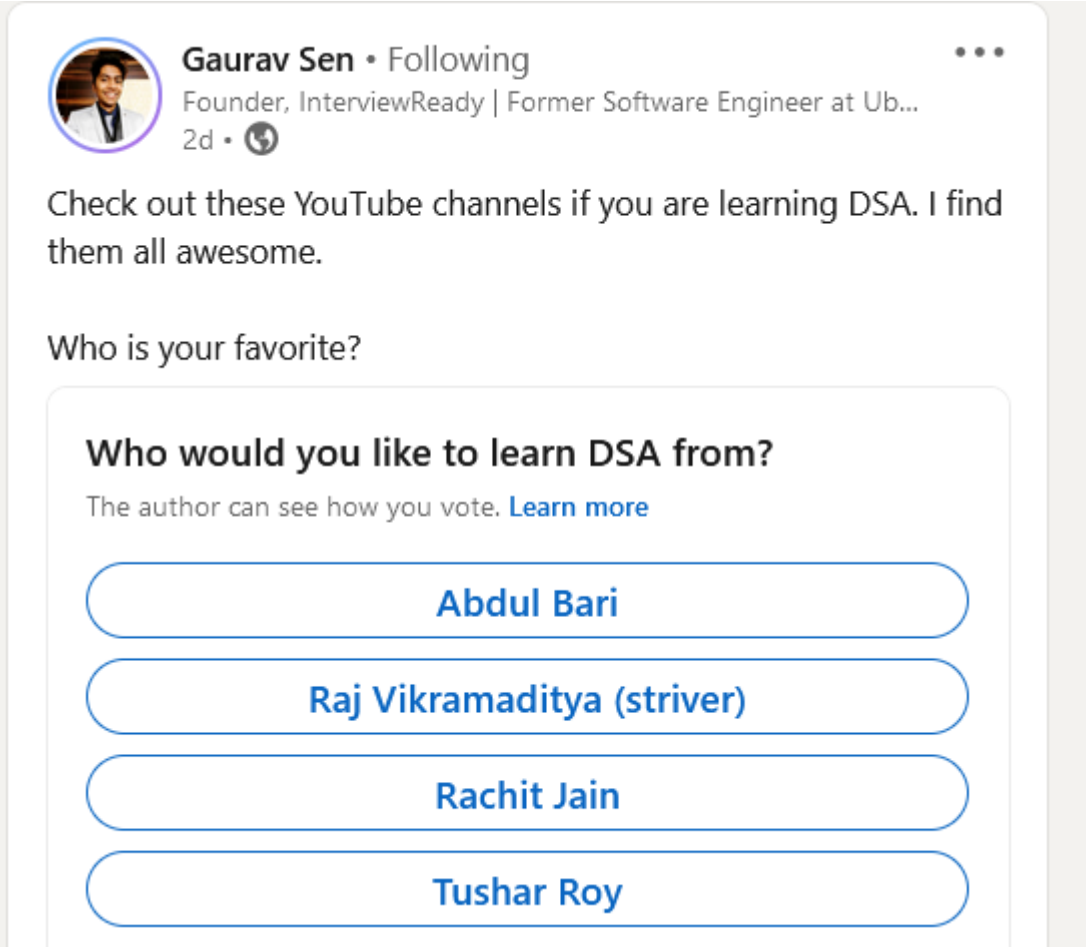
[25] splash.ripplematch.com (US-based)

[26] interviewquery.com/jobs (data jobs)

[27] Twitter

[28] LinkedIn

[29] jobboardsearch.com/remote-jobs (a list of remote job boards)



A helpful GitHub repo:

Best-of Machine Learning with Python - A ranked list of awesome machine learning Python libraries.

https://github.com/ml-tooling/best-of-ml-python

==========

reference books in information retrieval:

- Introduction to Information Retrieval

— by Chris Manning, Prabhakar Raghavan, and Hinrich Schütze

Modern Information Retrieval

— by Ricardo Baeza-Yates and Berthier Ribeiro-Neto

Search Engines: Information Retrieval in Practice

— by Bruce Croft, Don Metzler, and Trevor Strohman

==========

If you have a background in IR, then reading papers published at the top IR conferences (SIGIR, CIKM, WSDM, WWW) is about as good as you can get.

https://github.com/dabl/dabl

Data Analysis Baseline Library

==========

CVIT-PIB corpus that is the largest multilingual corpus available for Indian languages

http://preon.iiit.ac.in/~jerin/bhasha/

Nice articles with visualizations

https://amitness.com/

A Visual Guide to FastText Word Embeddings

https://amitness.com/2020/06/fasttext-embeddings/

ZERO SHOT LEARNING

https://amitness.com/2020/05/zero-shot-text-classification/

https://read.deeplearning.ai/the-batch/issue-146/

—----------

No Language Left Behind:

Scaling Human-Centered Machine Translation

https://scontent.fdel29-1.fna.fbcdn.net/v/t39.8562-6/292295068_402295381932691_8903854229220968087_n.pdf?_nc_cat=102&ccb=1-

7&_nc_sid=ad8a9d&_nc_ohc=KF3MAms3HygAX8P4R11&_nc_ht=scontent.fdel29-1.fna&oh=00_AT-lx2Ng1ZiBpOKThIbqOHirQaSL-UpJGOc9ZQQN6rp1IA&oe=62CC86D3

https://github.com/facebookresearch/fairseq/tree/nllb

==========

https://cds.nyu.edu/deep-learning/ – **Yann LeCun's Deep Learning Course at CDS**

The NLP Pandect, one of the most comprehensive resources for all things NLP, has reached over 1800 **stars on GitHub**!

https://github.com/ivan-bilan/The-NLP-Pandect

==========

Machine Learning University (MLU) is an education initiative from Amazon designed to teach machine learning theory and practical application.

Random Forest

https://mlu-explain.github.io/random-forest/?utm_campaign=Data_Elixir&utm_source=Data_Elixir_399

Logistic Regression article

https://mlu-explain.github.io/logistic-regression/

Train test Validation sets

https://mlu-explain.github.io/train-test-validation/

Precision Recall

https://mlu-explain.github.io/precision-recall/

Resource Bank of Computer Vision, Natural Langauge Processing and MLops

https://github.com/ashishpatel26/ResourceBank_CV_NLP_MLOPS_2022

==========

Flet is a framework that enables you to easily build realtime web, mobile and desktop apps in your favorite language and securely share them with your team. No frontend experience required.

https://github.com/flet-dev/flet

https://flet.dev/

DocQuery: Document Query Engine Powered by NLP

https://github.com/impira/docquery?utm_campaign=Data_Elixir&utm_source=Data_Elixir_403#readme

https://ibis-project.org/docs/3.1.0/#sql

==========

https://tech.instacart.com/how-instacart-uses-machine-learning-driven-autocomplete-to-help-people-fill-their-carts-9bc56d22bafb?utm_campaign=Data_Elixir&utm_source=Data_Elixir_403

—----------

Vakyansh

https://open-speech-ekstep.github.io/

Harveen Singh Chadha ⇒ Data & Applied Scientist II at Microsoft | Vakyansh

Following takeaways from the picture:

✓ Predict likely categories - Classification

✓ Discover groups - Clustering

✓ Predict numeric values - Regression

✓ Identify unusual class - Anomaly Detection

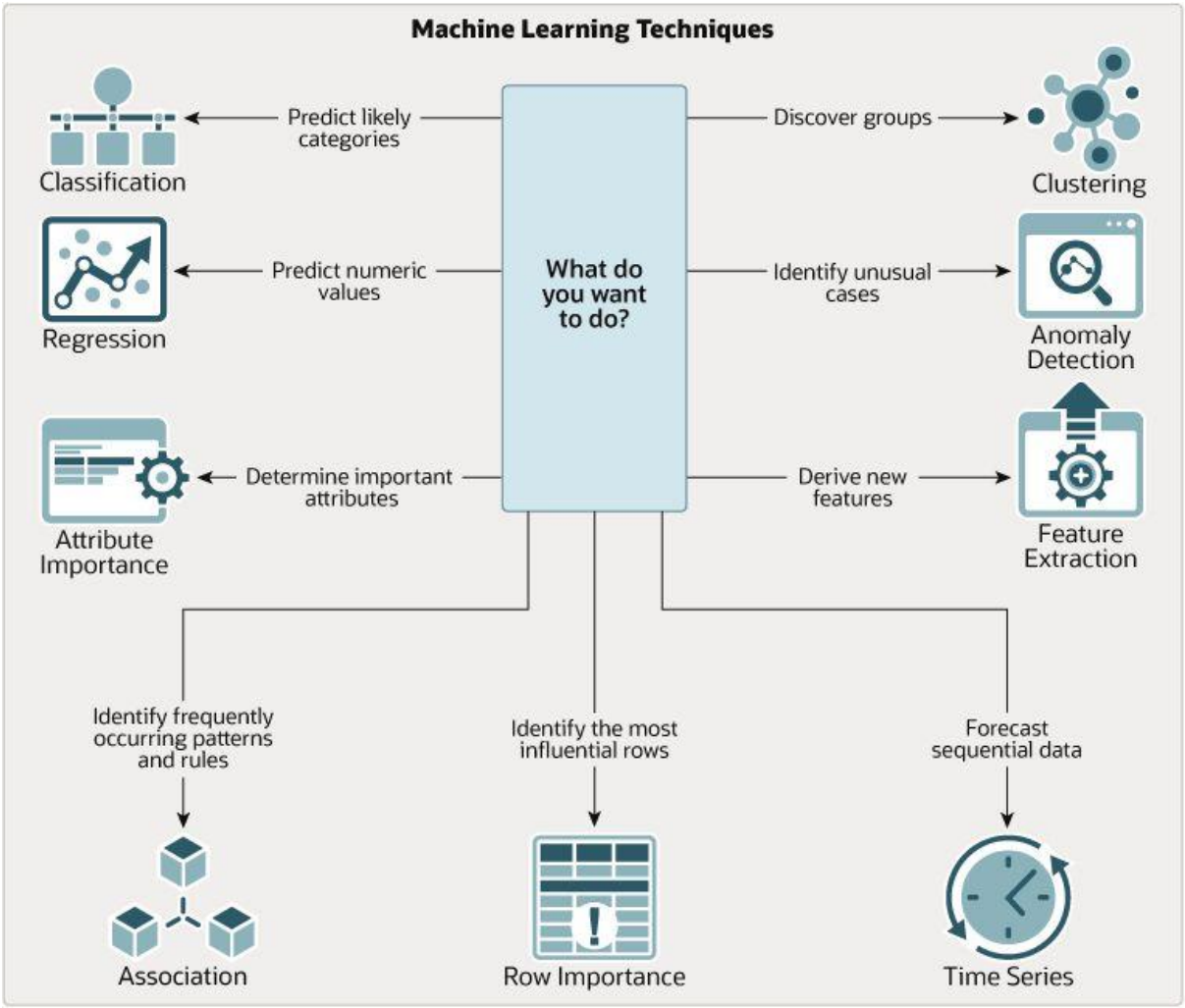
✓ Determine important attributes - Attribute Importance

✓ Derive new features - Feature Extraction

✓ Identify frequently occurring patterns and rules - Association

✓ Identify the most influential row - Row Importance

✓ Forecast sequential data - Time Series



Top 6 Github Repos to Learn Python and Data Science 🤖

1. Learn Python 3

By Jerry Pussinen

🌐 https://lnkd.in/eVbza36G

2. All algorithms implemented in Python

By The Algorithms

🌐 https://lnkd.in/eEU8uNkZ

3. Data Science Resources

By Jonathan Bower

🌐 https://lnkd.in/e8Qwnw_K

4. Awesome Data Science

By Fatih Aktürk, Hüseyin Mert & Osman Ungur, Recep Erol

🌐 https://lnkd.in/e-Nj8iRJ

5. Data Science Best Resources

By Tirthajyoti Sarkar

🌐 https://lnkd.in/eEgxBKqG

6. Data Scientist Roadmap

By MrMimic

🌐 https://lnkd.in/e7fJ8p3Y

Having Python Skills is greatly increases your value in Job Market...

Please don't pay for courses. You can learn for free here.....

1. Install Anaconda distribution from here..

https://lnkd.in/gxmiZdCP

1.1 Or practise from online IDE here.....

https://lnkd.in/gTNN_GjZ

2. start your python journey from here

https://lnkd.in/gBxgnbkY

https://lnkd.in/g5xaNsXP

https://lnkd.in/gfbMm7Xx

3. Practise Python challenges from here.....

https://lnkd.in/g8nAMKgp

4. onto your next step! start working on 190+ projects here....

https://lnkd.in/gDtM-ZfN

5. Finally list down your projects here.....

https://github.com/

follow this 5 step journey to put yourself on top.

Resources are short and crispy. definitely recommended.

I personally like #telsuko youtube channel python course.

💥 you can start off, with W3 schools here https://lnkd.in/gxuXenSi for basics understanding and problem solving.

💥 if you want to have detailed complete courses with video explanations do check out here.....https://lnkd.in/g4rfWdz5

🎯 follow #T3SOTutorials YouTube channel, it has more than 300+ problems to solve in easy mode. find the link here.....https://lnkd.in/gH52RJYX

🎯 follow Aman Kharwal, he has created 190+ python projects with source code and categorized to various experience levels. find the link here.....https://lnkd.in/geJ5fNcK

==========

https://github.com/guipsamora/pandas_exercises

HackerRank actually has a section for numpy:

https://www.hackerrank.com/domains/python/numpy/page/1

Start easy by doing regression or classification etc with pytorch/keras

==========

A common beginner's block - so I decided to document a 5-step roadmap for 100 days of ML learning! ⚡

1. Do some introductory coursework (5 weeks): Hands-on ML is easy but not unique unless you understand the working of these algorithms. A good theory course is helpful. I recommend one of:

a. Machine Learning Specialization by Andrew Ng (Coursera)

b. Stanford CS229 (YouTube)

c. Machine learning crash course (Google Developers)

Or in books: An Introduction to Statistical Learning (ISLR)

2. Learn Hands-on ML in Python (3 weeks): The best way is through practice, but boot camps can help cover the essentials. I recommend one of:

a. Python for Data science and ML bootcamp (Udemy)

b. Machine learning A-Z - Hands-on Python & R (Udemy)

Or in books: Hands-on ML with Sklearn, Keras, and Tensorflow (O'Reilly)

3. Review how to do EDA (1 week): Exploratory Data Analysis and Feature engineering are crucial blocks in the ML pipeline. Explore tutorials and existing work on Kaggle and GitHub. I am adding two examples in the comments.

4. Take part in your first Kaggle contests / Do your first two mini-projects (4 weeks): Start implementing your skills through some competitions on Kaggle: Titanic, House prices, and Digit recognizer are good starting sets.

Alternatively, how about doing two mini-projects?: Build your models using some datasets from the UCI Machine Learning Repository.

5. Read and plan for the future (1 week):

By now you have basic proficiency in working with ML algorithms. ML is a broad field, and you should now start thinking about sub-domains to explore (CV, NLP, RL, Optimization, Security, Fairness, etc.). Give time to read articles, work, and experiences of people in these fields. Plan your next steps.

This 100 days of preparation should give you 2 mini-projects, 1 rigorous theoretical coursework, and at least 6 weeks of hands-on experience. A strong foundation in ML. Pass it on!

==========

Resources:

Machine Learning Specialization: https://www.coursera.org/specializations/machine-learning-introduction

Stanford CS229: https://www.youtube.com/playlist?list=PLoROMvodv4rMiGQp3WXShtMGgzqpfVfbU

Machine learning crash course: https://developers.google.com/machine-learning/crash-course/

ISLR: https://www.statlearning.com/

Python for Data science and ML Bootcamp: https://www.udemy.com/course/python-for-data-science-and-machine-learning-bootcamp/

Machine learning A-Z - Hands-on Python & R: https://www.udemy.com/course/machinelearning/

Hands-on ML by O'Reilly: https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/

EDA on Titanic dataset: https://www.kaggle.com/code/ash316/eda-to-prediction-dietanic

EDA on text (Comment toxicity): https://www.kaggle.com/code/jagangupta/stop-the-s-toxic-comments-eda

==========

Awesome Book for Best practices of Data Cleaning and Exploration with Machine Learning

✍ I discovered several flows in the data science practice that data scientists engage in during my data science journey.

🔷 Rush to create the model, and as a result, they make serious mistakes.

🔷Data extraction, transformation, cleansing, and inquiry are still widely utilized in data science. But the work of a data scientist does not go in a linear fashion, from cleaning through exploration, preprocessing, modeling, and evaluation.

🤔 Data scientists must assess missing values, outliers, variable distributions, and correlations. 🤔 Every data scientist thinks, "Which algorithm is best?" These are common questions for aspiring data scientists.

Today I'm going to recommend a book to you that will help you find the solutions to all your problems.

Book: Data Cleaning and Exploration with Machine Learning by Mike Walker

Special Thanks to Shifa Ansari for providing the review copy of the book.

🚀 Key Ideas:

🏵 Data Cleaning

--

🔷Ch.1 - Examining the Distribution of features and target best practices - Code: https://bit.ly/3NreNB6

🔷Ch.2 - Examining Bivariate and Multivariate Relationships between Features and Targets(Special: KNN for outlier) - Code: https://bit.ly/3NoFV3O

🔷Ch.3 - Identifying and imputing the best practice of Missing values(KNN imputation is a special one) - Code: https://bit.ly/3FKgVC2

🔷Ch.4 - Encoding, transforming, and scaling features (Special: Data Leakage, K-means Binning) - Code: https://bit.ly/3frGPQy

❁ Feature Selection:

--

◈Ch.5 - Feature Selection (Special : classification - [Mutual Information, Anova F-value], Regression - [F-test, mutual information], Forward selection, backward selection, Exhausive feature selection, Eliminating features, Boruta for Feature Selection, Regularization, embedded methods) - Code:https://bit.ly/3DR5jMd

◈Ch.6 - Preparing for Model Evaluatation(Special : CAP, ROC and Precision-sensitivity curves for Binary classification, Multiclass Evaluation best practises) - Code:https://bit.ly/3frGVYq

❁ Regression:

--

◈Ch.7 - Linear Regression Models(Special : Lasso Regression, LR with Gradient Descent) - Code: https://bit.ly/3FGTABz

◈Ch.8 - Support Vector Regression(Special : SVM with Linear and non-Linear Kernels) - Code: https://bit.ly/3UhSrUV

◈Ch.9 - KNN, DecisionTree, Random Forest, and GBR- Code: https://bit.ly/3fn2DwL

❁ Classification:

--

◈Ch.10 - Logistic Regression (Special: Extension of LR) - Code: https://bit.ly/3FwZViN

◈Ch.11 - Decision Tree and Random Forest - Code: https://bit.ly/3zyMSJG

◈Ch.12 - KNN- Code: https://bit.ly/3T5am0g

◈Ch.13 - SVM - Code: https://bit.ly/3SSTrxD

◈Ch.14 - Naive Bayes- Code: https://bit.ly/3h28TdH

❁ Dimensionality Reduction:

--

◈Ch.15 - PCA - Code: https://bit.ly/3UjVQCP

🏵 Clustering:

--

◈Ch.16 - KMean and DBScan- Code: https://bit.ly/3UenxNh

—----------

ToDo - create an application that detect the

—----------

Generative AI

https://github.com/krishnaik06/Roadmap-To-Learn-Generative-AI-In-2024

NLP –

AI - do its tasks without the help of human intervention.

ML is the subset of AI. ML provides statistics tools to study/ analyze data. DL is subset of ML. DL is multi-layered, NN. In DL, we try to mimic human brain.

NLP deals with text,so we can convert text to vectors for machine to process. Chatbot, text summarization, lang translation, recommendation systems.

Machine is not able to capture sarcasm. NVIDIA open source model which can detect some sort of sarcasm.

Text Pre Processing - I (we are cleaning data so that it can be given to model) - tokenization,Lemmatization, Stop words,Stemming

Text Pre Processing-II : convert text to vectors→ Bag of words, TF-IDF,

Word2Vec, Unigrams, Bigrams

Text Pre Processing-III : word2vec, Avg Word2vec

ML variances : Spam classification, Chatbot, Text Summarization

DL : RNN, LSTM RNN, GRU RNN (Pre-requisite : Loss fns, Optimization)

Text Preprocessing Advanced : Word embeddings

Advanced DL : Bidirectional LSTm, Encoders, Decoders, Attention Models

Transformers

BERT

Libraries : NLTK, Spacy, TextBlob, Hugging face

TensorFlow/PyTorch

DALL-E : convert text to image

STEP1 : Text Pre-Processing

1. TOKENIZATION - Say you are building an email Spam classifier. Data set is email body (feature 1) and email subject (feature 2). Output - Spam/ham. ham means not a spam.

Independent features - email body,email subject.

Tokenization -> Stemming with stop words → Lemmatization

Tokenization is converting sentence into words.

2. Stop words like the,to etc not important. so can be removed. NLTK has 'not' in the stop words.
3. Stemming – process of reducing words to base/word stem. find base of a specific word.

eg historical history has base/stem same. (Root word)

history, historical gives histori

final,finally,finalized gives fina. <meaning lost>

Stemming is fast but root words can be meaningless.

use cases - spam classification, Review classification

4. Lemmatization => gives meaningful words.

final,finally,finalized gives final

history, historical gives history

Lemmatization is slow.

use cases - - Text summarization, Translation, Chatbot

Step 2 : Words to vectors

BoW -

TF-IDF

==========

If you're a Data Scientist, you should learn how to deploy models in production. Here are 3 books, 3 courses, 3 blogs and 3 podcasts if you want to level up your MLOps skills:

Books:

- Designing machine learning systems by Chip Huyen with actual case studies of ML systems in enterprise: https://lnkd.in/e2r7VAiK

- Practical MLOps, Operationalizing Machine Learning Models by Noah Gift & Alfreda Reza. https://lnkd.in/ek8MRKtz

- Machine learning engineering in action by Benjamin Wilson: https://lnkd.in/ejbJyCBj.

Courses:

- Deployment of machine learning model by Soledad Galli and Chris Samiullah. You get to learn how to deploy a model with 1) fast api, 2) containers 3) with IaaS on AWS: https://lnkd.in/efZUddpM

- Machine learning specialisation by Noah Gift and Alfredo Deza! If you want a full course on MLOPS, this is the one: https://lnkd.in/ea_7kbnF

- MLOps specialisation by the one and only Andrew Ng but also Laurence Moroney, Robert Crowe and Cristian Bartolome Aramburu: https://lnkd.in/eyEmyrPN

Blogs:

- Uber engineering blog: https://lnkd.in/egBQYmgP

- Airbnb Engineering and data science blog: https://airbnb.io/

- A hidden champion, Nubank: https://lnkd.in/ezKfZwcJ

Podcasts:

- MLOps Community Podcast, THE podcast if you want to learn more about MLOps: https://lnkd.in/eZhw57Z5

I recorded two great MLOps episodes on the AI Stories podcast:

- One with Demetrios Brinkmann, CEO of the MLOps community (and host of the podcast): https://lnkd.in/eSzGuJer

- Another one with Noah Gift, MLOps Leader and book author: https://lnkd.in/dBEb_Ns8

—----------