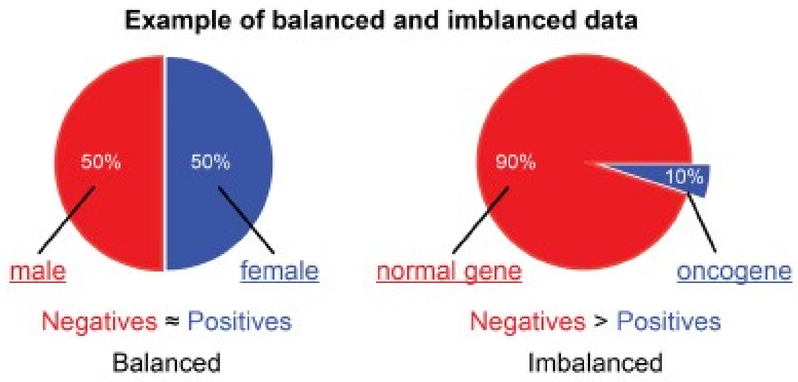
Question And Answer 2

1. What is Imbalanced and balanced dataset?

Ans:-



Balanced Dataset: — let's take a simple example if in our data set we have positive values which are approximately same as negative values. Then we can say our dataset in balance.

If my total data set D = 1000

If my +ve point = 600

And my –ve point = 400 it is balanced dataset example

Imbalanced Dataset: — If there is the very high different between the positive values and negative values. Then we can say our dataset in Imbalance Dataset.

If my total data set D = 1000

But if my +ve point = 900

And my –ve point = 100 it is imbalanced dataset example

It is the saviour issue in the model as our model give the accuracy 95% but not all the point so to resolve this we use two tech.

1. under sampling

2. Over sampling

1. under Sampling:

In under sampling we create a new data set and copy all the less point to this. Among rest are randomly sample.

Then it goes to balanced dataset so all model train using this new data set not old dataset.

Here we choose less amount of data among big data so it is under sampling.

In under-sampling, the simplest technique involves removing random records from the majority class, which can cause loss of information.

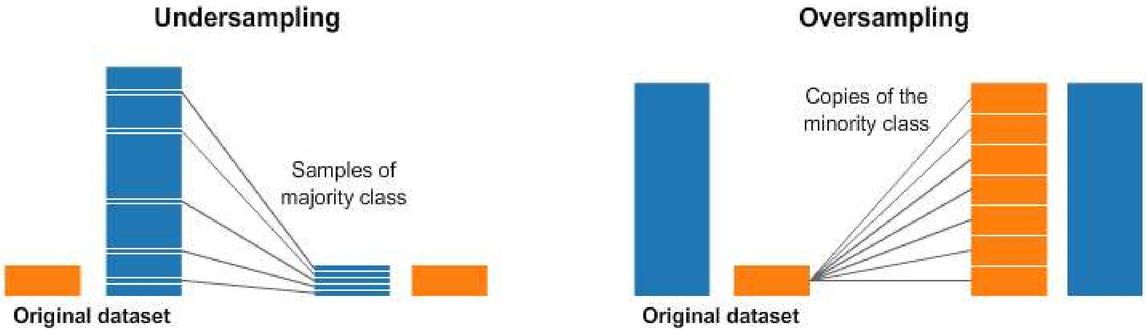
Limitation is here we not use large data as we use only small amount of data so it is not good.

2. Oversampling:

Here we repeat the data set to equal with other data set.

The simplest implementation of over-sampling is to duplicate random records from the minority class, which can cause overfishing.

As n1 = 900 and n2 = 100 here we repeat n2 9times to bring equal value compare with n1.



2. Define Multi-class classification?

In machine learning, multiclass or multinomial classification is the problem of classifying instances into one of three or more classes (classifying instances into one of two classes is called binary classification).

e.g., classify a set of images of fruits which may be oranges, apples, or pears. Multi-class classification makes the assumption that each sample is assigned to one and only one label: a fruit can be either an apple or a pear but not both at the same time.

The existing multi-class classification techniques can be categorized into

(i) Transformation to binary

(ii) Extension from binary

(iii) Hierarchical classification.

Transformation to binary

It Can reducing the problem of multiclass classification to multiple binary classification problems

It can be categorized into one vs rest and one vs one.

One-vs.-rest Or One vs All (oVr):

One-vs-rest (OvR for short, also referred to as One-vs-All or OvA) is a heuristic method for using binary classification algorithms for multi-class classification.

Strategy involves training a single classifier per class, with the samples of that class as positive samples and all other samples as negatives.

It involves splitting the multi-class dataset into multiple binary classification problems. A binary classifier is then trained on each binary classification problem and predictions are made using the model that is the most confident.

Ex: *For example, given a multi-class classification problem with examples for each class 'red,' 'blue,' and 'green'. This could be divided into three binary classification datasets as follows:*

Binary Classification Problem 1: red vs [blue, green]
Binary Classification Problem 2: blue vs [red, green]
Binary Classification Problem 3: green vs [red, blue]

One vs one (OvO):

One-vs-One (OvO for short) is another heuristic method for using binary classification algorithms for multi-class classification.

Like one-vs-rest it splits a multi-class classification dataset into binary classification problems. Unlike one-vs-rest that splits it into one binary dataset for each class, the one-vs-one approach splits the dataset into one dataset for each class versus every other class.

If we see OvO in the above example looks like:

Binary Classification Problem 1: red vs. blue
Binary Classification Problem 2: red vs. green
Binary Classification Problem 3: red vs. yellow
Binary Classification Problem 4: blue vs. green
Binary Classification Problem 5: blue vs. yellow
Binary Classification Problem 6: green vs. yellow

This is significantly more datasets, than one vs rest.

Given a multiclass classification (c) -> it require c binary classification problem

3. Explain Impact of Outliers?

Outliers often has a significant effect on your mean and standard deviation. Because of this, we must take steps to remove outliers from our data sets.

Outliers can have a disproportionate effect on statistical results, such as the mean, which can result in misleading interpretations.

It also affect skewness of the data. If there is outlier in data then it will be right or left skew not normal distribution.

4. What is Local Outlier Factor?

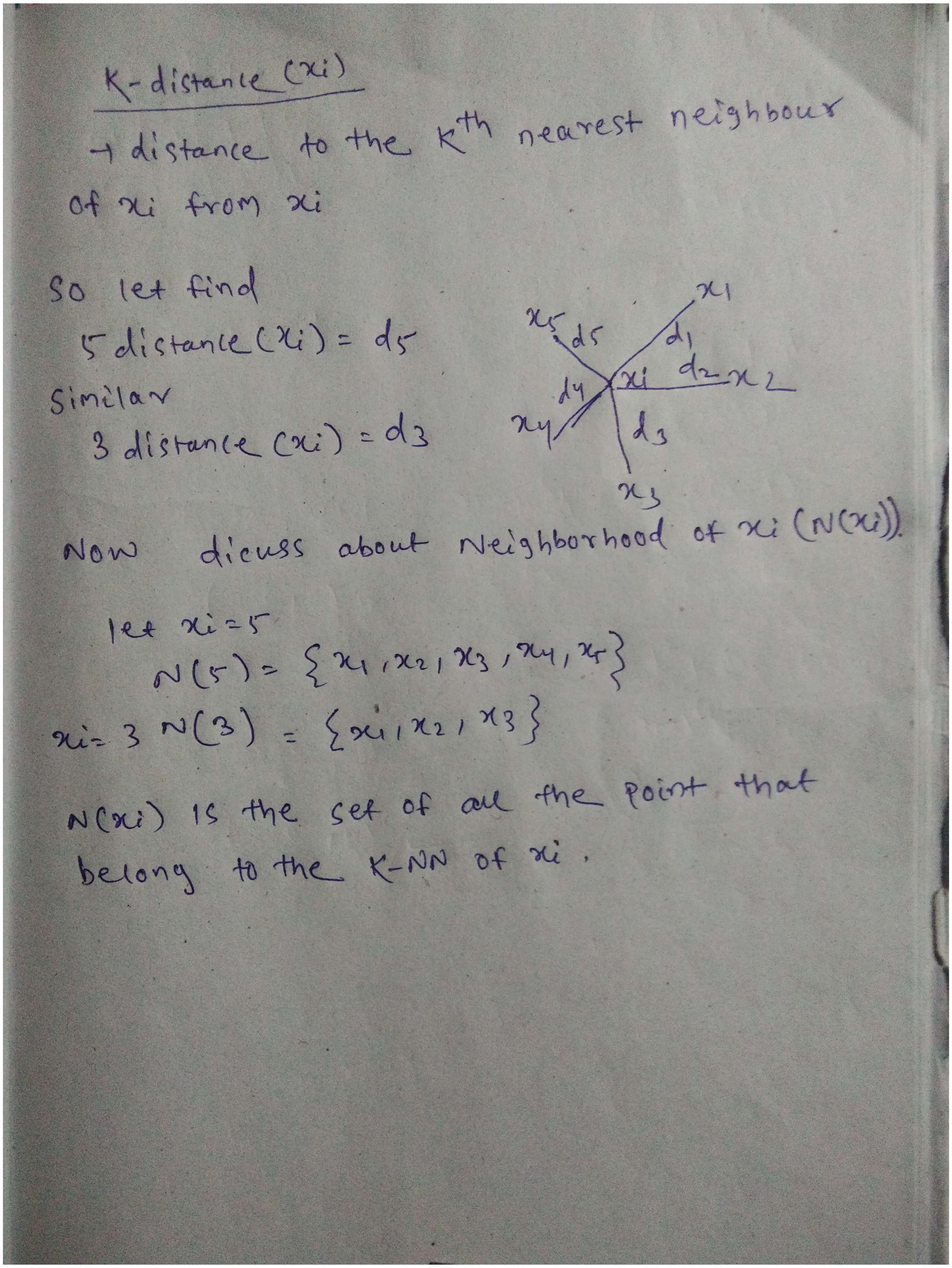
Ans – Local Outlier Factor (LOF) is a score that tells how likely a certain data point is an outlier/anomaly.

If LOF ≈1 -> No Outlier

If LOF >> 1 -> Outlier

The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbours. It considers as outliers the samples that have a substantially lower density than their neighbours.

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



K-distance (x_i)

→ distance to the K^{th} nearest neighbour of x_i from x_i

So let find

5 distance $(x_i) = d_5$

Similar

3 distance $(x_i) = d_3$

x_1
x_5
d_5
d_1
d_2
x_2
x_i
d_4
x_4
d_3
x_3

Now dicuss about Neighborhood of x_i $(N(x_i))$.

let $x_i = 5$

$N(5) = \{x_1, x_2, x_3, x_4, x_5\}$

$x_i = 3$ $N(3) = \{x_i, x_2, x_3\}$

$N(x_i)$ is the set of all the point that belong to the K-NN of x_i.

6. Define reachability-distance (a,b)?

Ans:- The k-distance is now used to calculate the reachability distance. This distance measure is simply the maximum of the distance of two points and the k-distance of the second point.

$$\text{reach-dist}(a,b) = \max\{\text{k-distance}(b), \text{dist}(a,b)\}$$

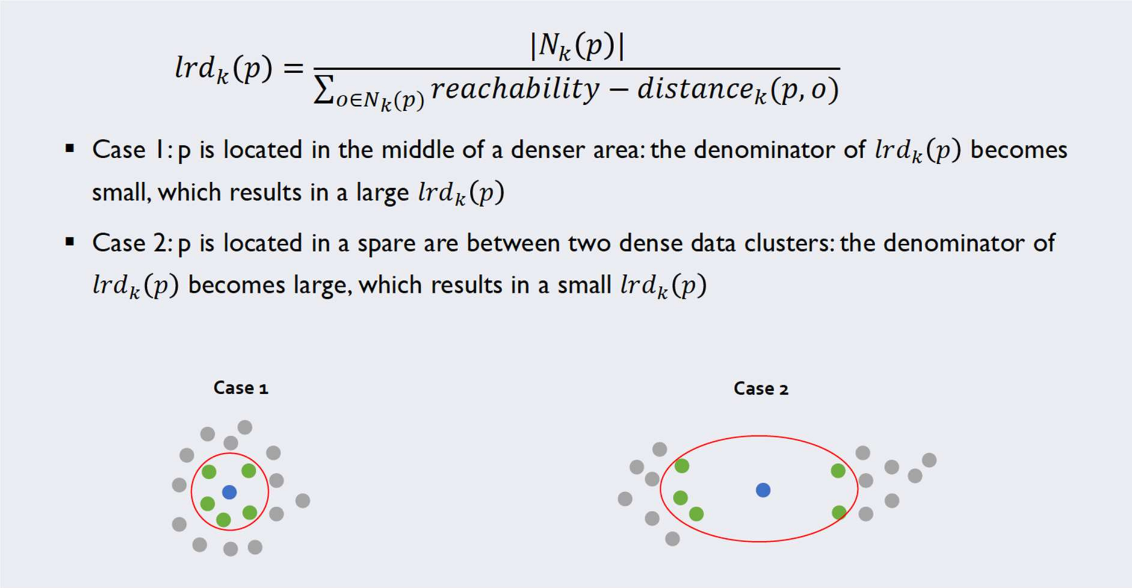
Where :

Dist (a,b) = actual distance

k-distance (b) = = distance to the kth nearest neighbour of a from a.

Basically, if point a is within the k neighbors of point b, the reach-dist(a,b) will be the k-distance of b. Otherwise, it will be the real distance of a and b. This is just a “smoothing factor”. For simplicity, consider this the usual distance between two points.

7. What is Local-reachability-density (lrd(p))?



$$lrd_k(p) = \frac{|N_k(p)|}{\sum_{o \in N_k(p)} reachability - distance_k(p,o)}$$

- Case 1: p is located in the middle of a denser area: the denominator of $lrd_k(p)$ becomes small, which results in a large $lrd_k(p)$
- Case 2: p is located in a spare are between two dense data clusters: the denominator of $lrd_k(p)$ becomes large, which results in a small $lrd_k(p)$

8. Define LOF?

The lrd of each point will then be compared to the lrd of their k neighbours. More specifically, k ratios of the lrd of each point to its neighbouring points will be calculated and averaged. The LOF is basically the average ratio of the lrds of the neighbours of a to the lrd of a. If the ratio is greater than 1, the density of point a is on average smaller than the density of its neighbours and, thus, from point a, we have to travel longer distances to get to the next point or cluster of points than from a's neighbours to their next neighbours. Keep in mind, the neighbours of a point a may don't consider a a neighbour as they have points in their reach which are way closer.

9. Impact of Scale & Column standardization?

Ans :- 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 attribute 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, such as linear regression, logistic regression, and linear discriminant analysis.

10.What is Interpretability Vs BlackBox?

Ans:- black box model – such model gives the output without any reason. It take output with itself.

But interpretable model gives output with proper reason .Means it says why this output/how this output came.

k-nn is interpretable model when dimension is small and k also small.

11.Handling categorical features?

Ans:- Here are a few examples Of Categorical Data:

The city where a person lives: Delhi, Mumbai, Ahmedabad, Bangalore, etc.
The department a person works in: Finance, Human resources, IT, Production.
The highest degree: High school, Diploma, Bachelors, Masters, PhD.
The grades of a student: A+, A, B+, B, B- etc.
Gender: Male, Female

In the above examples, the variables only have definite possible values. Further, we can see there are two kinds of categorical data-

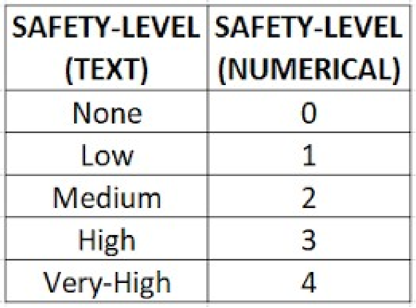
- **Ordinal Data:** The categories have an inherent order (Highest degree)
- **Nominal Data:** The categories do not have an inherent order (Gender)

All Type Of Method:

- Label Encoding or Ordinal Encoding
- One hot Encoding
- Dummy Encoding
- Effect Encoding
- Binary Encoding
- BaseN Encoding

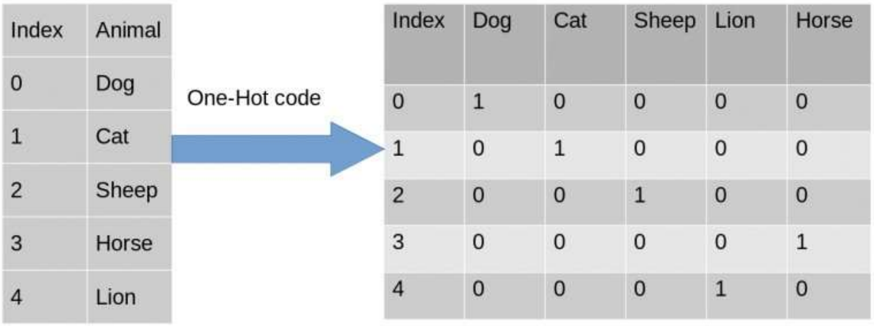
- Hash Encoding
- Target Encoding

a. Label Endcoding:



SAFETY-LEVEL (TEXT)	SAFETY-LEVEL (NUMERICAL)
None	0
Low	1
Medium	2
High	3
Very-High	4

b. One hot Encoding

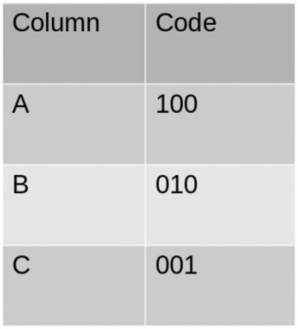


Index	Animal
0	Dog
1	Cat
2	Sheep
3	Horse
4	Lion

One-Hot code

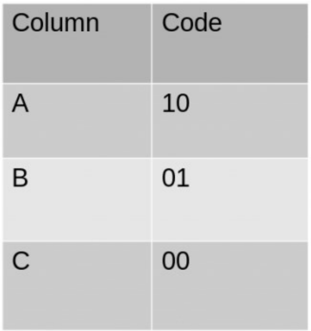
Index	Dog	Cat	Sheep	Lion	Horse
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	0	1
4	0	0	0	1	0

c. Dummy Encoding



Column	Code
A	100
B	010
C	001

One- Hot Coding



Column	Code
A	10
B	01
C	00

Dummy Code

Dummy coding scheme is similar to one-hot encoding. This categorical data encoding method transforms the categorical variable into a set of binary variables (also known as dummy variables).

In the case of one-hot encoding, for N categories in a variable, it uses N binary variables. The dummy encoding is a small improvement over one-hot-encoding. Dummy encoding uses N-1 features to represent N labels/categories.

12. Handling missing values?

Ans:- Why missing values treatment is required?

Missing data in the training data set can reduce the power / fit of a model or can lead to a biased model because we have not analysed the behaviour and relationship with other variables correctly. It can lead to wrong prediction or classification.

There are several method to fix this. i.e

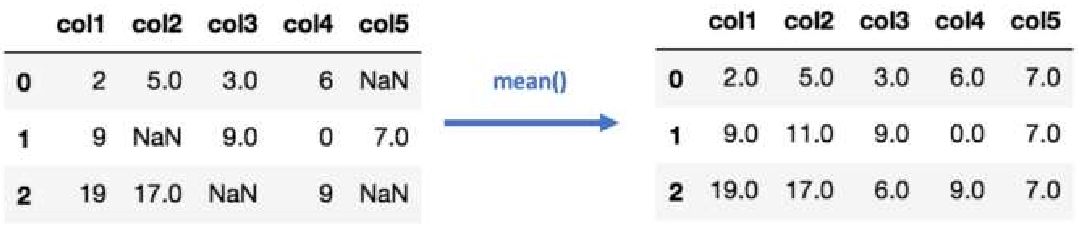
1- Do Nothing:

That’s an easy one. You just let the algorithm handle the missing data. Some algorithms can factor in the missing values and learn the best imputation values for the missing data based on the training loss reduction (ie. XGBoost).

However, other algorithms will panic and throw an error complaining about the missing values (ie. Scikit learn — Linear Regression). In that case, you will need to handle the missing data and clean it before feeding it to the algorithm.

2- Imputation Using Mean/Median Value :

This works by calculating the mean/median of the non-missing values in a column and then replacing the missing values within each column separately and independently from the others. It can only be used with numeric data.



	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

	col1	col2	col3	col4	col5
0	2.0	5.0	3.0	6.0	7.0
1	9.0	11.0	9.0	0.0	7.0
2	19.0	17.0	6.0	9.0	7.0

3- Imputing Using Mode or (Zero/constant) Value :

Most Frequent is another statistical strategy to impute missing values and YES!! It works with categorical features (strings or numerical representations) by replacing missing data with the most frequent values within each column.

Zero or Constant imputation - it replaces the missing values with either zero or any constant value you specify