TECHVIDYA

#1
LIMITED EDITION

VOL 107

QUESTIONS

SUPERVISED LEARNING USING PYTHON

"STAY UPDATED, STAY AHEAD"

TechVidya is ISO Certified 9001:2015 accredited EdTech Company registered with ROC under the companies act 1956, offers self-paced, online and offline programs. TechVidya pedagogy is rooted in the principle that every young mind should be equipped with exceptional knowledge and skills that benefit them in real life.

YEAR 2023

Company Info
TechVidya
EdTech Company

Company Website
techvidya.education
Official Website

Company Contact
+91 83759 66700
Official Number

1. What is Machine Learning ?

a. Have you ever shopped online? while checking for a product, did you notice it recommends product similar to what you are looking for? or did you see "the person bought this product also bought this" combination of products. How are they doing this recommendation? This is machine learning.

b. Did yo u ever get a call from any bank or finance company asking you to take loan or any insurance policy? Do they call everyone? No, they call only selected customers who they think will purchase their product. How do they select? This is target marketing and can be applied using Clustering. This is machine learning

c. Do you go to supermar ket for groceries or something? Ever noticed chips are placed near cold drinks? Why? Whoever buys cold drink will most likely buy chips. How did they get to know this relation? This is association rule mining (market basket analysis). This is machine learning

d. Do you know most of the time when you chat on a website, it’s not a human with whom you are talking to. It’s a bot. This is machine learning

e. Driverless car is a direct product of machine learning and artificial intelligence

2. What is Supervised Learning?

- We can go on and on with the bas ic definition of Supervised learning, but that would defeat the purpose of the book. Supervised means a thing which you can monitor. Supervised learning includes all the algorithms where you know the output of some data. You train your model on these data assuming the fact that these are correct data points. And then you build a model on top of it.

Example – We want to know the number of customers which will come to my restaurant in November. Now, I have the number of customers who have visited my restaurant in the last 3 years. So, we have some data points of the past, we can build a forecasting model using these data points and then we can predict the customers visiting in coming November.

Anything for which we know the output for few data points will fall under supervised learning

3. What are the applications of supervised learning?

- PC Games
- Chat Bots
- Forecastin g number of visitors on Amazon
- Classification of objects for Tesla

-

4. What is unsupervised learning?

- A supervised learning needs some o utput to build a model. An unsupervised learning algorithm needs nothing. It will build a model on your training dataset y finding connection between different values and it will keep iterating the process until all the data points are considered. An example will help you understand better:-

Example – You have things with different geometric shape, some are circular, some are oval, square, rectangular, etc. You need to make bucket these into 4 parts. Now the algorithm which you will use does not know anything about bucketing, it only knows that you need 4 buckets. It will most probably take the first 4 items and place them on a co-ordinate. Now each object coming in will be allocated near to one of the four buckets. The algorithm will keep iterating till you are done with all the items. By then end of the run, you will have 4 buckets. This is unsupervised learning

5. What is reinforcement learning?

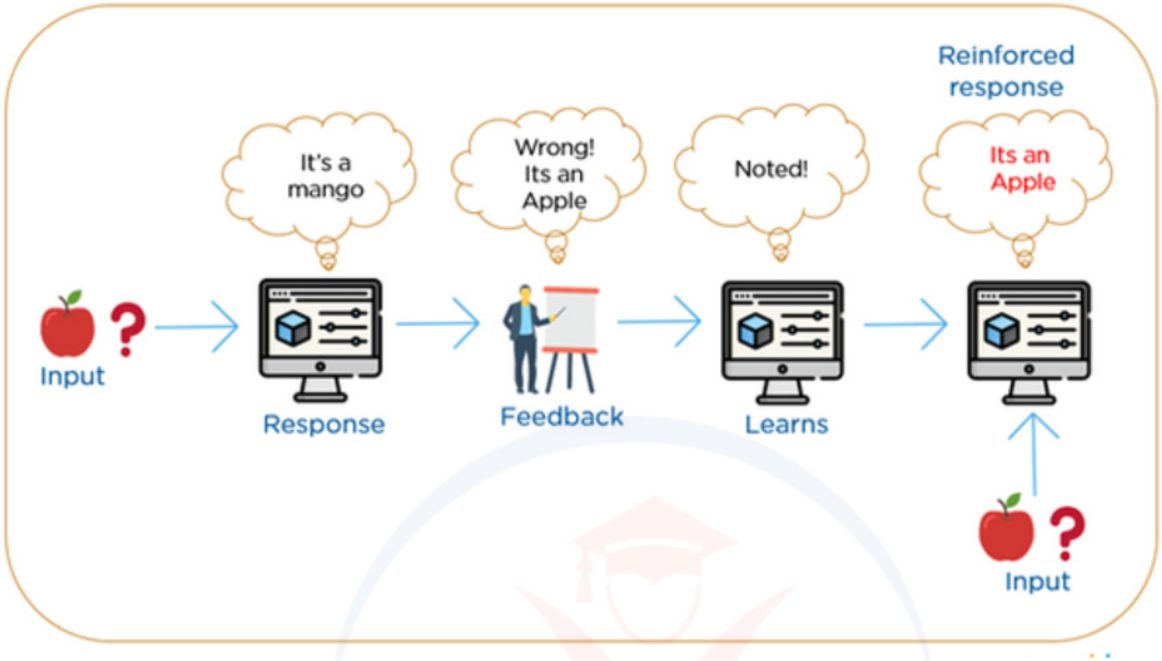
We talked about supervised and unsup ervised learning. But this is not all. There is this third type of machine learning, which is called reinforcement learning.

Example - Let’s imagine that a new born baby comes across a lit candle.

Now, the baby does not know what happens if it touches the flame. Eventually, out of curiosity, the baby tries to touch the flame and gets hurt. After this incident, the baby learns that repeating the same thing again might get him hurt. So, the next time it sees a burning candle, it will be more cautious

That is exactly how Reinforcement learning works. Reinforcement learning is a kind of Machine Learning where in the system that is to be trained to do a particular job, learns on it’s own based on its previous experiences and outcomes while doing a similar kind of a job.

6. Explain Reinforcement learning with an example(Sandeep Dayananda)



Look at the image here.
1. You provide the syste m with an image of an apple and ask it to identify it.
2. The computer comes up with an answer as you can see on the image...it says it's a 'mango'.
3. You tell the syste m that it's a wrong answer and the image is of an apple. That's the feedback.
4. The machine learn s from the feedback.
5. Finally, if it comes across another imag e of an apple, it will be able to identify it correctly.

That's reinforcement learning.

7. What are the applications of unsupervised learning?

- Image recognition or identification

8. What are the applications of Reinforcement learning?

- PC Games
- Robotics

9. Classification vs regression?

10. What are the predictor and target variable?

A target variable is the one which you want to pred ict or forecast and the predictor variables are those which actually affects the prediction. Example – Suppose I want to predict the number of customers com ing in my restaurant next Monday, then the number of customers will be target variable and predictor variable will be something like, is it a holiday (Binary variable), season of the year(may be more people visits a restaurant in winter), festival – If there is some festival then more people may come to the restaurant. These are your predictor variable.

11. Different names for predictor and target variable

The reason why this question is here is because you will be exploring a lot of thing if you are into supervised modeling and you will go through multiple blogs and study materials. Different m=naming conventions will definitely confuse you. So, remember what is written below

a. Features = Predictor Variable = Independent variable

b. Target variable = Dependent variable = Response Var iable

12. Uses of Supervised Learning

- Predicting the future
- Demand Supply pred iction
- Diagnosis

13. How to get labeled data?

To perform Supervised learnin g all you need is a labeled data i.e. a dataset which have the target values. There are three ways in which you can get the labeled data:-

a. Historic da ta – The restaurant example will again come to my rescue, suppose I have historic data of last 3 years, so I have the target variable which is monthly number of customers.

b. Experiments to label data

c. Crowd source labeled data

14. What are the packages in Python?

There are multiple packages which are v ery handy for supervised learning, but we will use the most common package i.e. scikit-learn or sklearn package

Other im portant libraries are Tensor flow and Keras.

15. What is scikit?

Scikit-learn provide s a range of supervised and unsupervised learning algorithms via a consistent interface in Python.

It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use.

The library is built upon the SciPy (Scientific Python) that must be installed before you can use scikit-learn. This stack that includes:

NumPy: Base n-dimensional array package

SciPy: Fundamental library for scientific co mputing

Matplotlib: Comprehensive 2D/3D plotting

IPython: Enhanced interactive console

Sympy: Symbolic mathematics

Pandas: Data structures and ana lysis

16. What is the main difference between classification problems and regression problems in machine learning?

Regression is used to predict continuous valu es. Classification is used to predict which class a data point is part of (discrete value).
Example: I have a house with W rooms, X bathrooms, Y s quare-footage and Z lot-size. Based on other houses in the area that have recently sold, how much (dollar amount) can I sell my house for? I would use regression for this kind of problem.
Example: I have an unkno wn fruit that is yellow in color, 5.5 inches long, diameter of an inch, and density of X. What fruit is this? I would use classification for this kind of problem to classify it as a banana (as opposed to an apple or orange).

17. Is logistic regression classification or regression?

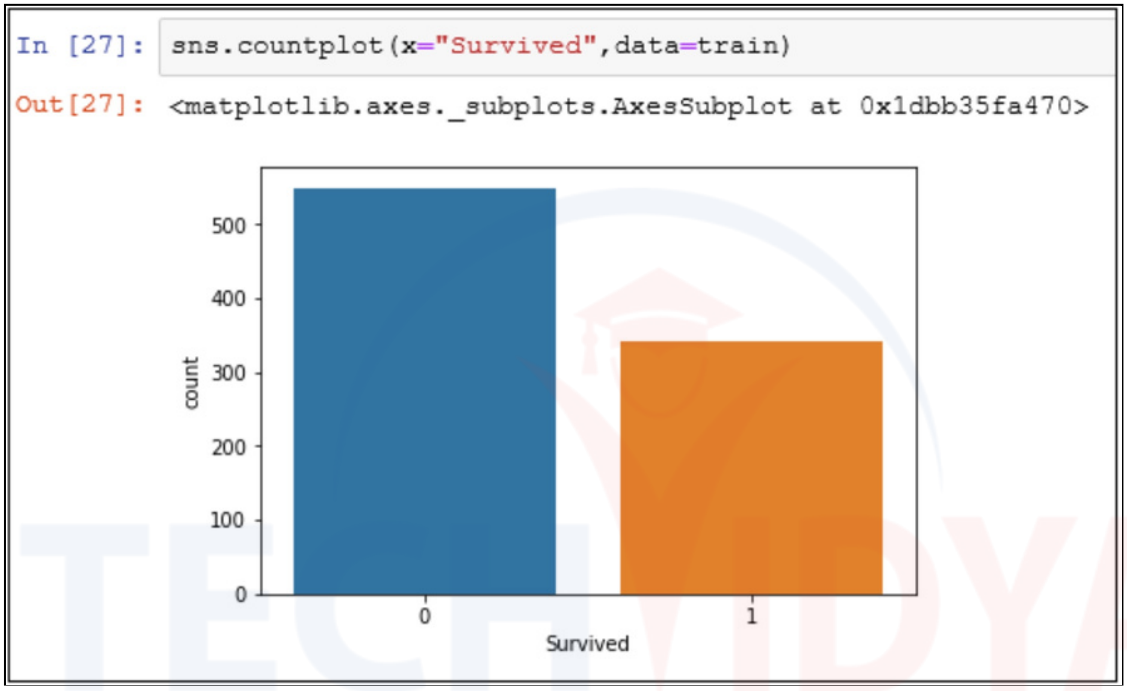
Logistic regression is emphatically not a classification a lgorithm on its own. It is only a classification algorithm in combination with a decision rule that makes dichotomous the predicted probabilities of the outcome. Logistic regression is a regression model because it estimates the probability of class membership as a (transformation of a) multilinear function of the features.

18. What is data.shape?

data.shape is the comman d which gets you the number of rows and columns in the dataset.

19. EDA using seaborn's countplot

The countplot function of seaborn lib rary is a very useful function which is used to plot the count of a categorical variable. Let's suppose we have the titanic dataset which have a column "Survived" and it is a binary variable with 0 denoting not survived and 1 denoting survived. Plotting this using countplot is very easy.



20. Types of classifiers?

There are different types of classifiers:
a. Perceptron b. Naïve Baye s
c. Decision Tre e
d. Logistic Regre ssion
e. K-Nearest Neighbor
f. Artificial Neural Net work
g. Support Vector Machine

21. What is knn?

You will get 100s of definitions of KNN on internet, but to keep it simple. The basic idea of K-Nearest Neighbor is to predict the label for any item by looking at the value of k. Okay, let's understand it, you have a sample of animals, few have 4 legs and others have 2 legs. Now you put the value of k as 2 that means you need to make 2 buckets. The algorithm will pick the first item and will place it in a bucket. It will again take up another item and will place it somewhere on the co-ordinate. Now from the third sample it will start placing the item near to the one of these buckets. Sooner all the items will be put in one or the other bracket

22. How does knn works?

KNN works by analogy. Th e idea is that you are what you resemble.
So when we want to classify a point we look at its K-closest (most si milar) neighbors and we classify the point as the majority class in those neighbors.
KNN depends on two things: A metric used to compute the distance between two points and the value of "k" the number of neighbors to consider.
When "k" is a very small number KNN can over fit, it will classify just based on the closest neighbors instead of learning a good separating frontier between classes. But if "k" is a very big number KNN will under fit, in the limit if k=n KNN will think every point belongs to the class that has more samples.
KNN can be used for regression, just average the value for the k nearest neighbors or a point to predict the value for a new point.
One nice advantage of KNN is that it can work fine if yo u only have a few samples for some of the classes.

23. What is fitting the model?

Fitting the model is training a m odel i.e. you take two arguments, independent variable and dependent variable and train your model on these data points.
model.fit(X _train,y_train)

Let’s build some supervised models

24. Let’s create our dataset first. We will create Thyroid data set with attributes as Weight, Blood Sugar, and Sex(M=1,F=0)

```
#Weight,Blood Sugar, and gender Male = 1, Female = 0
X = [[80, 150,0], [90, 200, 0], [95, 160, 1], [110, 200, 1] , [70, 110, 0],
 [60,100,1], [70, 300, 0],
     [100,200,1], [140, 300, 0 ], [60, 100, 1], [70,100,0], [100,300,1], [70,
110, 1]]

y =
['Thyroid','Thyroid','Normal','Normal','Normal','Normal','Thyroid','Normal','
Thyroid',
'Normal', 'Normal','Thyroid','Normal']
```

25. What are X and y?

The list X contains the a ttributes and y contains the classification

26. How to build the testing dataset?

We have to create a test data set also. L et’s take some values which are not identical to the above values but are close to the labels. So we know that [90,200,0] is a thyroid patient , so we will test our model on [100,250,0] and we have already labeled it to Thyroid

X_test = [[100,250,0],[70,110,1],[60,100,0],[120,300,1]]

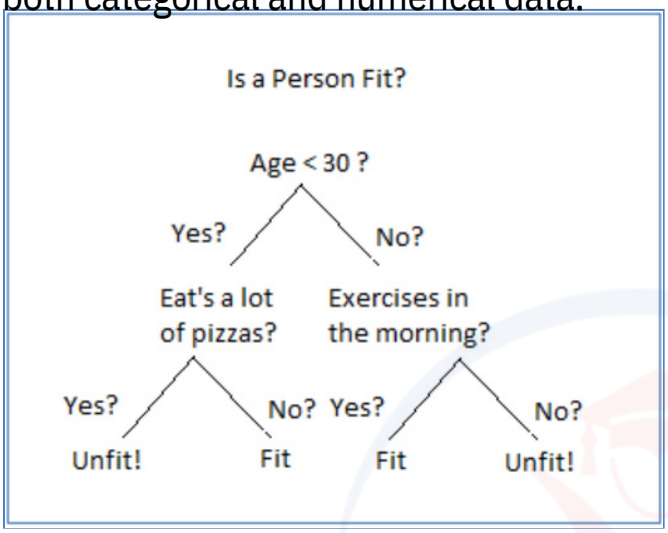
y_test = ['Thyroid','Normal','Normal','Thyroid']

27. Let’s understand Decision Tree first. Define Decision Tree.

Decision trees learn from data to approximate a sine curve with a s et of if-then-else decision rules. The deeper the tree, the more complex the decision rules and the fitter the model. Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

28. Define Leaf and Node of a Decision Tree

A decision node has two or more branches. Lea f node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.



29. How does a Decision Tree works? What is splitting?

Decision tree follows three steps – Splitting, Pruning, and T ree Selection. Decision Tree starts with Splitting which is a process of partitioning the data into subsets. Splits are formed on a particular variable. In the above Decision tree the split on the first level happened on the variable which is Age. Then further split happened on Pizza and exercise in morning.

30. What is pruning?

The shortening of bran ches of the tree. Pruning is the process of reducing the size of the tree by turning some branch nodes into leaf nodes, and removing the leaf nodes under the original branch. Pruning is useful because classification trees may fit the training data well, but may do a poor job of classifying new values. A simpler tree often avoids over-fitting.

31. What is tree selection?

The process of finding the s mallest tree that fits the data. Usually this is the tree that yields the lowest cross-validated error.

32. What is entropy?

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogeneous). If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.

33. Let's create a Decision Tree Classifier

from sklearn import tree

DecisionT = tree.DecisionTreeClassifier()

DecisionT = DecisionT.fit(X,y)
DecisionT_prediction = Decisio nT.predict(X_test)
print (DecisionT_prediction)

```
In [20]: #DecisionTreeClassifier
         DecisionT = tree.DecisionTreeClassifier()
         DecisionT = DecisionT.fit(X,y)
         DecisionT_prediction = DecisionT.predict(X_test)
         print (DecisionT_prediction)

         ['Thyroid' 'Normal' 'Normal' 'Thyroid']
```

34. Explain the above code

First we imported tree packa ge from sklearn library. The function DecisionTreeClassifier in the tree package holds the model so we initialize our model 'DecisionT' with the above function. This will create a decision tree model. Now we need to fit the model on our training data set i.e. X and y. So we have used DecisionT.fit(X,y).
Predict function takes up your test data set and predicts it on the basis of values

35. What is a Random forest?

Random forests is a supervised learning algorithm. It can be used both for classification and regression. It is also the most flexible and easy to use algorithm. A forest is comprised of trees. It is said that the more trees it has, the more robust a forest is. Random forests creates decision trees on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance.

36. What are the applications of Random Forest?

andom forests has a variety of applications, such as r ecommendation engines, image classification and feature selection. It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the base of the Boruta algorithm, which selects important features in a dataset.

37. What is feature importance in Random Forest?

A great quality of the random forest algorithm is that i t is very easy to measure the relative importance of each feature on the prediction. Sklearn provides a great tool for this, that measures a features importance by looking at how much the tree nodes, which use that feature, reduce impurity across all trees in the forest. It computes this score automatically for each feature after training and scales the results, so that the sum of all importance is equal to 1.

38. How does the Random Forest Algorithm works?

Step 1 – The algorithm will select a random sample from the given dataset
Step 2 – Construct a complete Decision Tree and get a prediction result
Step 3 – Get a vote from each predicted result
Step 4 -Select the prediction result with the mo st votes as the final prediction.

39. What are the advantages of Random Forest?

The one main advantage of Random Forest is that it considers almost all the combination of results so the accuracy on the training dataset is very high. It does not suffer from over fitting problem.
One more advantage is that it can be used in Regression and Classification problem

40. What are the disadvantages of Random Forest?

It fails to provide the same level of accuracy on the tes t data set because the algorithm is not trained on unseen values, so it loses accuracy there.
The model is made up of multiple trees, so it is hard to interpret the backend algorithm

41. Do a Random Forest vs Decision Tree.

Many decision trees make up a forest
Decision trees are computationally faster
Random Forest is difficult to interpret

42. We already have a Decision Tree model at place, now let's create a Random Forest Classifier?

from sklearn.ensemble impor t RandomForestClassifier

RandomF = RandomForestClassifier()
RandomF.fit(X,y)
RandomF_predict ion = RandomF.predict(X_test)
print (RandomF_prediction)

```
In [21]: from sklearn.ensemble import RandomForestClassifier
         #RandomForestClassifier
         RandomF = RandomForestClassifier()
         RandomF.fit(X,y)
         RandomF_prediction = RandomF.predict(X_test)
         print (RandomF_prediction)
         ['Thyroid' 'Normal' 'Normal' 'Normal']
```

43. Explain the code above.

The process of building the m odel is same as Decision Tree. Import the Random Forest Classifier package, fit the model on the training dataset, use predict() function to predict values for test data

Let's create one more model, a basic but highly effective model i.e. Logistic Regression model

44. What is Logis tic Regression?

Logistic Regression is a Machine L earning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). In other words, the logistic regression model predicts P(Y=1) as a function of X.

What are the assumptions of Logistic Regression?
Binary logistic regression requires the dependent v ariable to be binary.

45. What are the types of questions that Logistic regression can examine?

How does the probability of getting lung cancer (yes vs. no) change for every additional pound a person is overweight and for every pack of cigarettes smoked per day?
Do body weight, calorie int ake, fat intake, and age have an influence on the probability of having a heart attack (yes vs. no)?

46. What are the major assumptions in Logistic regression?

The dependent variable should be dichotomous in nature (e.g., p resence vs. absent).

There should be no outliers in the data, which can be assessed by converting the continuous predictors to standardized scores, and removing values below -3.29 or greater than 3.29.

There should be no high correlations (multicollinearity) among the predictors. This can be assessed by a correlation matrix among the predictors. Tabachnick and Fidell (2013) suggest that as long correlation coefficients among independent variables are less than 0.90 the assumption is met.

47. What is overfitting?

When selecting the mode l for the logistic regression analysis, another important consideration is the model fit. Adding independent variables to a logistic regression model will always increase the amount of variance explained in the log odds (typically expressed as R^2). However, adding more and more variables to the model can result in overfitting, which reduces the generalizability of the model beyond the data on which the model is fit.

48. Let's build a Logistic Regression Classifier

from sklearn.linear_model import LogisticRegres sion
#LogisticRegression
LogisticR = Logistic Regression()
LogisticR.fit(X,y)
LogisticR_prediction = LogisticR.predict(X_test)
print (LogisticR_prediction)

```
In [22]: from sklearn.linear_model import LogisticRegression
         #LogisticRegression
         LogisticR = LogisticRegression()
         LogisticR.fit(X,y)
         LogisticR_prediction = LogisticR.predict(X_test)
         print (LogisticR_prediction)

         ['Thyroid' 'Normal' 'Normal' 'Thyroid']
```

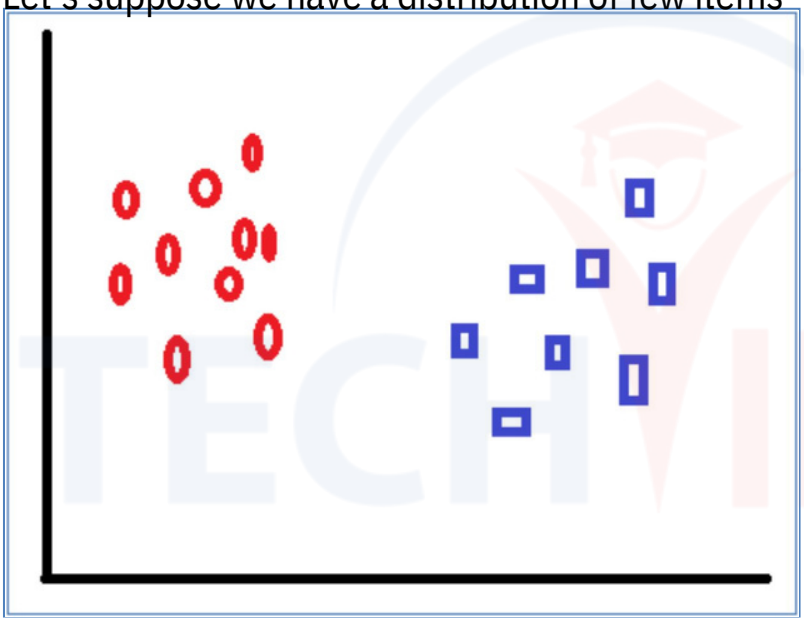
Let's explore one more model i.e. Support Vector Classifier

49. What is Support Vector Classifier?

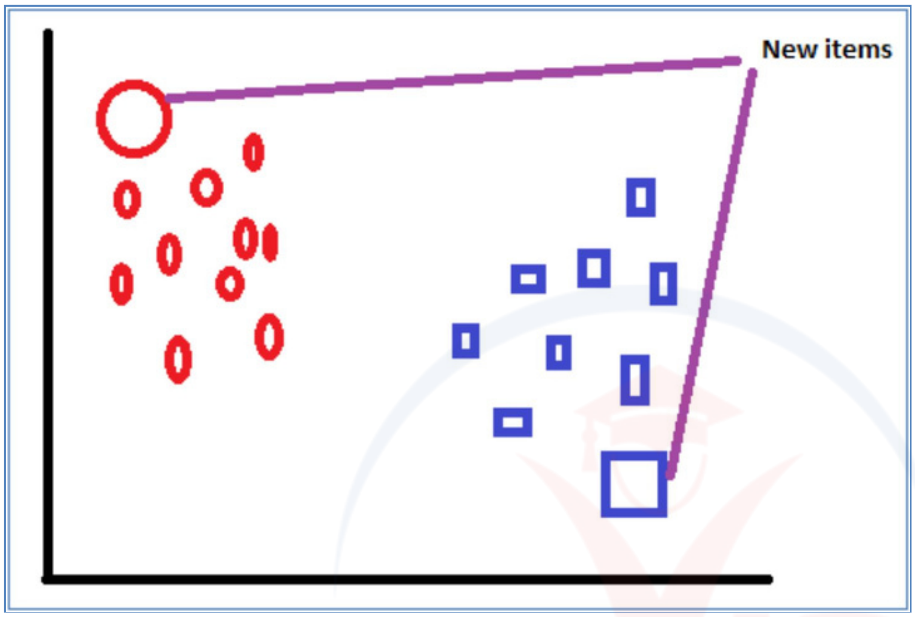
A Support Vector Machine (SVM) is a di scriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. In two dimentional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side.

50. Give an example to explain SVM.

Let's suppose we have a distribution of few items



What an SVM does is that it makes a line of separation between the types of objects i.e. circle and rectangle. The attribute of test dataset is then examined on various parameters and are then placed in one of the two buckets like give below



51. What will happen if data points overlaps, i.e. circles and rectangles are on the same point?

In real world application , finding perfect class for millions of training data set takes lot of time. As you will see in coding. This is called regularization parameter. In next section, we define two terms regularization parameter and gamma. These are tuning parameters in SVM classifier. Varying those we can achive considerable non linear classification line with more accuracy in reasonable amount of time.

52. What is a kernel?

SVM algorithms use a set of mathematical functions that are defined as the kernel. The function of kernel is to take data as input and transform it into the required form.

For linear kernel th e equation for prediction for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows:

f(x) = B(0) + sum(ai * (x,xi))

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

53. What is the equation for polynomial kernel? Leave the question for now if you don't want to go deep into Mathematics.

The polynomial kernel can be written as

K(x,xi) = 1 + sum(x * xi)^d

exponential as

K(x,xi) = exp(- gamma * sum((x — xi²))

54. What is Regularization?

The Regularization parameter (often termed as C parameter in python's sklearn library) tells the SVM optimization how much you want to avoid misclassifying each training example.

55. What is the impact of a large and a small value of c?

For large values of C, the optimization will choose a smalle r-margin hyper plane if that hyper plane does a better job of getting all the training points classified correctly. Conversely, a very small value of C will cause the optimizer to look for a larger-margin separating hyper plane, even if that hyper plane misclassifies more points.

56. What is Gamma?

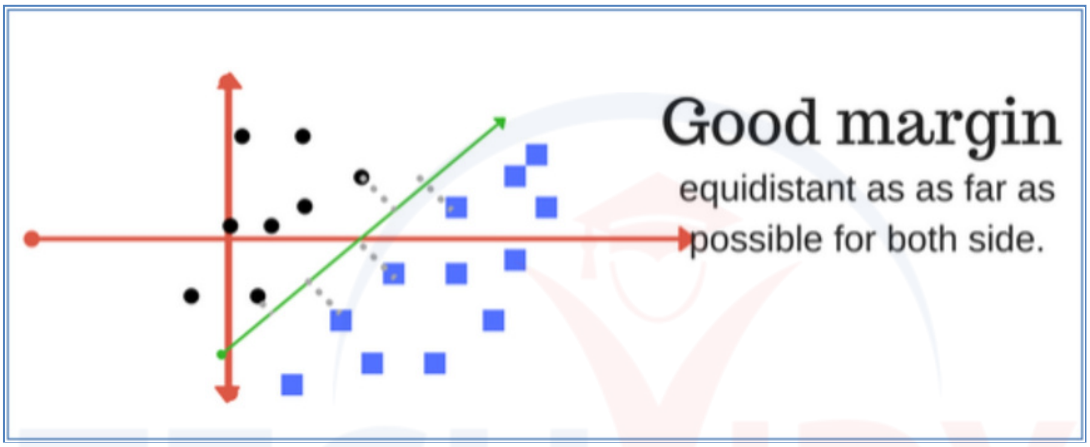
The gamma parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'. In other words, with low gamma, points far away from plausible separation line are considered in calculation for the separation line. Whereas high gamma means the points close to plausible line are considered in calculation.

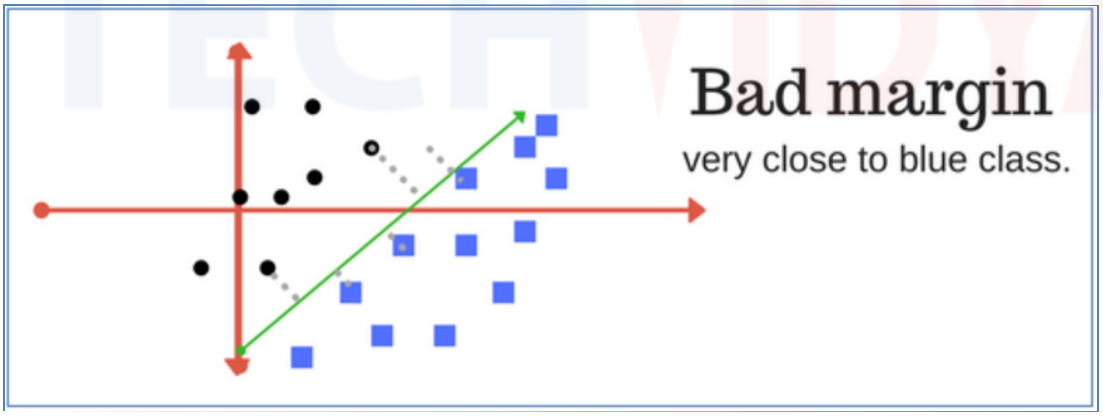
57. What is a margin?

A margin is a separation of line to the closest class points.

58. What is a good and a bad margin?

A good margin is one where this separati on is larger for both the classes. Images below gives to visual example of good and bad margin. A good margin allows the points to be in their respective classes without crossing to other class.





59. Let's build the Support Vector Classifier model on our dataset

from sklearn.svm import SVC
#Support Vector Classifier
SupportV = SVC()
SupportV.fit(X,y)
SupportV_predict ion = SupportV.predict(X_test)
print (SupportV_prediction)

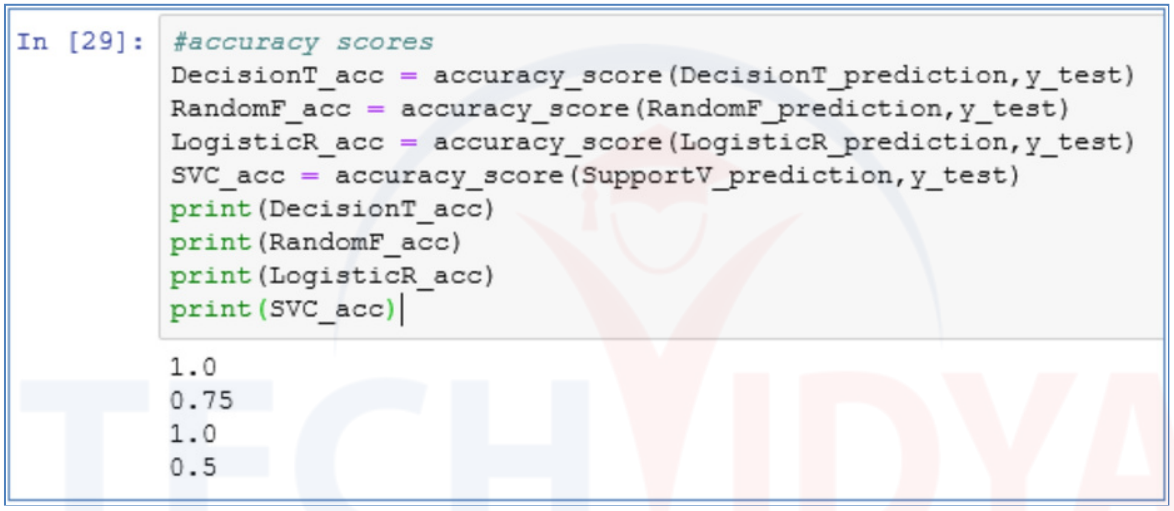
```
In [23]: from sklearn.svm import SVC
         #Support Vector Classifier
         SupportV = SVC()
         SupportV.fit(X,y)
         SupportV_prediction = SupportV.predict(X_test)
         print (SupportV_prediction)

         ['Normal' 'Normal' 'Normal' 'Normal']
```

60. How to measure the accuracy of models ?

We can use the accuracy_score() function which takes two parameters which are prediction values and real output i.e. y_test. These are present in the following packages
import numpy as np
from sklearn.metrics import accuracy_score

61. Print the accuracy of each model

DecisionT_acc = accuracy_score(Decis ionT_prediction,y_test)
RandomF_acc = accuracy_score(RandomF_prediction,y_test)
LogisticR_acc = accuracy_score(LogisticR_prediction,y_test)
SVC_acc = accuracy_score(SupportV_prediction,y_test)
print(DecisionT_acc)
print(RandomF_acc)
print(LogisticR_acc)
print(SVC_acc)



K-Nearest Neighbor is one such algorithm which is very useful in classification problem. It is a very basic algorithm which gives a good accuracy.

62. What is KNN Algorithm?

The intuition behind the KNN a lgorithm is one of the simplest of all the supervised machine learning algorithms. It simply calculates the distance of a new data point to all other training data points. The distance can be of any type e.g Euclidean or Manhattan etc. It then selects the K-nearest data points, where K can be any integer. Finally it assigns the data point to the class to which the majority of the K data points belong.

63. What are the pros of KNN model?

a. It is extremely easy to implement
b. The KNN algorithm much faster than other algorithms that require training e.g. SVM, linear regression, etc.
c. Since the algorithm requires no trainin g before making predictions, new data can be added seamlessly.
d. There are only two paramet ers required to implement KNN i.e. the value of K and the distance function (e.g. Euclidean or Manhattan etc.)

64. What are the cons of KNN model?

a. The KNN algorithm doesn't work well with high dimensional data because with large number of dimensions, it becomes difficult for the algorithm to calculate distance in each dimension

b. The KNN algorithm has a high prediction cost for large datasets. This is because in large datasets the cost of calculating distance between new point and each existing point becomes higher

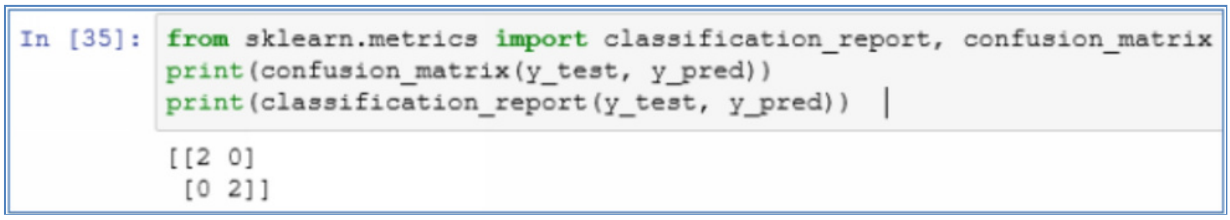
c. Finally, the KNN algorithm doesn't w ork well with categorical features since it is difficult to find the distance between dimensions with categorical features

65. We will try to build a basic KNN model with 5 neighbors. Write the code for the same.

```
from sklearn.neighb ors import KNeighborsClassifier
classifier = KNeighborsClassifier(n_neighbors=5)
classifier.fit(X, y)
y_pred = classifier.predict(X_test)
```

66. Let's create a confusion matrix to see how did the model perform?

from sklearn.metrics import classification_report, confusion_matrix
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))



As you can see the accuracy of the model is 100%. This is because the number of training and testing dataset are very less. Once you try your hands with larger dataset, then only you will be able to check the performance of other models.

67. Don't get confused between KNN and K-means algorithm. What is the difference between the two?

K-Nearest Neighbors is a supervis ed classification algorithm, while k-means clustering is an unsupervised clustering algorithm. While the mechanisms may seem similar at first, what this really means is that in order for K-Nearest Neighbors to work, you need labeled data you want to classify an unlabeled point into (thus the nearest neighbor part). K-means clustering requires only a set of unlabeled points and a threshold: the algorithm will take unlabeled points and gradually learn how to cluster them into groups by computing the mean of the distance between different points.

The critical difference here is that KNN needs labeled points and is thus supervised learning, while k-means doesn't — and is thus unsupervised learning.

After going through the above algorithms, you must have got a good idea about implementing these algorithms on different datasets provided in various Hackathons. Go through Kaggle or Analytics Vidhya to participate

in Live and Past Hackathons.

Apart from implementation of these 5 algorithms, we also need to know the evaluation metrics, some definitions used in supervised learning, etc. All of these will be covered below

68. What is ROC curve?

The ROC curve is a graph ical representation of the contrast between true positive rates and the false positive rate at various thresholds. It's often used as a proxy for the trade-off between the sensitivity of the model (true positives) vs the fall-out or the probability it will trigger a false alarm (false positives).

69. Define Precision and Recall.

Recall is also known as the true po sitive rate: the amount of positives your model claims compared to the actual number of positives there are throughout the data. Precision is also known as the positive predictive value, and it is a measure of the amount of accurate positives your model claims compared to the number of positives it actually claims. It can be easier to think of recall and precision in the context of a case where you've predicted that there were 10 apples and 5 oranges in a case of 10 apples. You'd have perfect recall (there are actually 10 apples, and you predicted there would be 10) but 66.7% precision because out of the 15 events you predicted, only 10 (the apples) are correct.

70. What is the difference between L1 and L2 Regularization?

L2 regularization tends to spread error among all the terms, while L1 is more binary/sparse, with many variables either being assigned a 1 or 0 in weighting. L1 corresponds to setting a Laplacean prior on the terms, while L2 corresponds to a Gaussian prior.

71. What's the difference between Type I and Type II error?

Type I error is a false positive, while Type II error is a false nega tive. Briefly stated, Type I error means claiming something has happened when it hasn't, while Type II error means that you claim nothing is happening when in fact something is.

A clever way to think about this is to think of Type I error as telling a man he is pregnant, while Type II error means you tell a pregnant woman she isn't carrying a baby.

72. What's the trade-off between bias and variance?

Bias is error due to erroneous or overly simplistic assum ptions in the learning algorithm you're using. This can lead to the model underfitting your data, making it hard for it to have high predictive accuracy and for you to generalize your knowledge from the training set to the test set.
Variance is error due to too much complexity in the learning algo rithm you're using. This leads to the algorithm being highly sensitive to high degrees of variation in your training data, which can lead your model to overfit the data. You'll be carrying too much noise from your training data for your model to be very useful for your test data.
The bias-variance decomposition essentially decom poses the learning error from any algorithm by adding the bias, the variance and a bit of irreducible error due to noise in the underlying dataset. Essentially, if you make the model more complex and add more variables, you'll lose bias but gain some variance — in order to get the optimally reduced amount of error, you'll have to tradeoff bias and variance. You don't want either high bias or high variance in your model.

73. How is a decision tree pruned?

Pruning is what happens in decision trees when branches that have weak predictive power are removed in order to reduce the complexity of the model and increase the predictive accuracy of a decision tree model. Pruning can happen bottom-up and top-down, with approaches such as reduced error pruning and cost complexity pruning.

Reduced error pruning is perhaps the simplest versio n: replace each node. If it doesn't decrease predictive accuracy, keep it pruned. While simple, this heuristic actually comes pretty close to an approach that would optimize for maximum accuracy

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

Model accuracy is a very misleading parameter to judge a model. A model could be useless even after having 99% accuracy. Suppose you are creating a model to classify a very rare disease as whether a patient is infected by that disease. Then even if you tag every patient as "not infected" then the model will have more than 99%. But this model is not at all useful. So, the model performance is the best matrix to judge the working of a model.

75. What's the F1 score? How would you use it?

The F1 score is a measure of a model's performanc e. It is a weighted average of the precision and recall of a model, with results tending to 1 being the best, and those tending to 0 being the worst. You would use it in classification tests where true negatives don't matter much.

76. How would you handle an imbalanced dataset?

An imbalanced dataset is when you have, for example , a classification test and 90% of the data is in one class. That leads to problems: an accuracy of 90% can be skewed if you have no predictive power on the other category of data! Here are a few tactics to get over the hump:

1- Collect more data to even the imbalances in the d ataset.

2- Resample the dataset to correct for imbalances

3- Try a different algorithm altogether on your da taset

77. When should you use classification over regression?

Classification produces discrete values and dataset to strict categories, while regression gives you continuous results that allow you to better distinguish differences between individual points. You would use classification over regression if you wanted your results to reflect the belongingness of data points in your dataset to certain explicit categories (ex: If you wanted to know whether a name was male or female rather than just how correlated they were with male and female names.)

78. Name an example where ensemble techniques might be useful.

Ensemble techniques use a combination of learning algorithms to optim ize better predictive performance. They typically reduce over fitting in models and make the model more robust (unlikely to be influenced by small changes in the training data).

You could list some examples of ensemble methods, from bagging to boosting to a “bucket of models” method and demonstrate how they could increase predictive power.

What’s important here is th at you have a keen sense for what damage an unbalanced dataset can cause, and how to balance that.

79. How do you ensure you're not over fitting with a model?

This is a simple restatement of a fundamental problem in machin e learning: the possibility of over fitting training data and carrying the noise of that data through to the test set, thereby providing inaccurate generalizations.
There are three main methods to avoid over fitting:
1- Keep the model simpler: reduce variance by taki ng into account fewer variables and parameters, thereby removing some of the noise in the training data.
2- Use cross- validation techniques such as k-folds cross-validation.
3- Use regularization techniques such as LASSO that penalize certa in model parameters if they're likely to cause over fitting.

80. What evaluation approaches would you work to gauge the effectiveness of a machine learning model?

You would first split the dataset into training a nd test sets, or perhaps use cross-validation techniques to further segment the dataset into composite sets of training and test sets within the data.
You could use measures such as the F1 scor e, the accuracy, and the confusion matrix. What's important here is to demonstrate that you understand the nuances of how a model is measured and how to choose the right performance measures for the right situations

81. How do you handle missing or corrupted data in a dataset?

You could find missing/corrupted data in a dataset and either drop t hose rows or columns, or decide to replace them with another value.

In Pandas, there are two very useful methods: isnull() and dropna() that will help you find columns of data with missing or corrupted data and drop those values. If you want to fill the invalid values with a placeholder value (for example, 0), you could use the fillna() method.

82. What's the trade-off between bias and variance?

Bias is error due to erroneous or overly simplistic assum ptions in the learning algorithm you're using. This can lead to the model underfitting your data, making it hard for it to have high predictive accuracy and for you to generalize your knowledge from the training set to the test set.
Variance is error due to too much complexity in the learning algo rithm you're using. This leads to the algorithm being highly sensitive to high degrees of variation in your training data, which can lead your model to overfit the data. You'll be carrying too much noise from your training data for your model to be very useful for your test data.
The bias-variance decomposition essentially decom poses the learning error from any algorithm by adding the bias, the variance and a bit of irreducible error due to noise in the underlying dataset. Essentially, if you make the model more complex and add more variables, you'll lose bias but gain some variance — in order to get the optimally reduced amount of error, you'll have to tradeoff bias and variance. You don't want either high bias or high variance in your model.

83. What happens when you take large value for K in KNN algorithm?

A large value of K in KNN algorithm makes it completely expensive. It means that you are creating large clusters.

84. What happens when you take smaller value for K?

A small value of k means that noise will have a higher infl uence on the result

85. How to select the optimum value of k in knn?

There after methods which are used to identify the c orrect or optimal value of k in knn algorithm. The methods are :-

a. Elbow method

b. Cross Validatio n method

86. What is a cross validation method?

Cross-validation can be used to estimate t he test error associated with a learning method in order to evaluate its performance, or to select the appropriate level of flexibility.

87. How does KNN algorithm works?

KNN works by analogy. The idea is that you are what you resemble.
So when we want to classify a point we look at its K-closest (most si milar) neighbors and we classify the point as the majority class in those neighbors.
KNN depends on two things: A metric used to compute the distance between two points and the value of "k" the number of neighbors to consider.
When "k" is a very small number KNN can over fit, it will classify just based on the closest neighbors instead of learning a good separating frontier between classes. But if "k" is a very big number KNN will under fit, in the limit if k=n KNN will think every point belongs to the class that has more samples.
KNN can be used for regression, just average the value for the k nearest neighbors or a point to predict the value for a new point.
One nice advantage of KNN is that it can work fine if yo u only have a few samples for some of the classes.

88. What is the difference between binary classification and multi-class classification?

In a binary clas sification model we need to classify the output in only two types Like Typhoid or normal, Male or Female, Survived or Not, etc.
In multi class classification we need to classify the output in more th an two types. Like, Types of flowers or types of animals, etc.

89. Write a program to do cross validation for knn

```
# creating odd list of K for KNN
from sklearn.model_selection im port cross_val_score
myList = list(range(1,50))
# subsetting just the odd o nes
neighbors = filter(lambda x: x % 2 != 0, myList)
# empty list that will hold cv scores
cv_scores = []
# perform 10-f old cross validation
for k in neighbors:
knn = KNeighbo rsClassifier(n_neighbors=k)
 scores = cross_val_score(knn, X, y, cv=10, sc oring='accuracy')
 cv_scores.append(scores.mean())
```

90. Get the case here:-

a. If a person will purch ase a new home or not – Classification
b. Number of bikes rented in a month – Linear
c. Whether a person has diabetes – Classification

91. What is reshape function?

NumPy provides the reshape() f unction on the NumPy array object that can be used to reshape the data. The reshape() function takes a single argument that specifies the new shape of the array. It is common to need to reshape a one-dimensional array into a two-dimensionalarray with one column and multiple arrays. It gives a new shape to an array without changing its data (123) becomes (123,1) if we use the code y.reshape(-1,1)

92. What is correlation? How can you find the correlation of variables on a dataframe?

Correlation is a st atistical measure that indicates the extent to which two or more variables fluctuate together. A positive correlation indicates the extent to which those variables increase or decrease in parallel; a negative correlation indicates the extent to which one variable increases as the other decreases.

Df.corr()

93. How to explore a dataset in Python?

There are multiple commands which can he lp you in exploring a data set. Following are a few commands:
.info()
.descri be()
.head()

94. What is loss function?

At its core, a loss function i s incredibly simple: it’s a method of evaluating how well your algorithm models your dataset. If your predictions are totally off, your loss function will output a higher number. If they’re pretty good, it’ll output a lower number. As you change pieces of your algorithm to try and improve your model, your loss function will tell you if you’re getting anywhere.

95.Why can’t we use 1000 fold to get the best accuracy ?

More folds results in more computational expense. The samp le code for cross_val_score is given below:

```
from sklearn.model_selection im port cross_val_score
reg = linear_model.LinearRegression()
cv_results = cross_val_score(reg,X,y,cv =5)
```

96. What are the constraints of small and large alpha value in SVM?

Small alpha value will lead to over fitting because it will not penalize hig h coefficient

large alpha will lead to under fitting

97. What is Lasso regression?

Lasso performs regularization b y adding to the loss function a penalty term of the absolute value of each coefficient multiplied by some alpha. This is known as L1 Regularization because the regularization term is the L1 norm of the coefficients

98. What is L2 Regularization?

Always give more priority to L2 r egularization as compared to L1 i.e. Lasso Regression. It takes squared value of the coefficients

63. Why accuracy is not always the best metric to jud ge the performance of a model?

email spa m. It will be horrible at actually classifying a spam. Therefore fails at its original purpose

99. How to build your confusion metrics in Python?

```
# Import necessary modules
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

# Create training and test set

X_train, X_test, y_train, y_te st =
train_test_split(X,y,test_size=0.4,random_state=42)


# Instantiate a k-NN classifier: knn
knn = KNeighborsClassifier()

# Fit the classifier to the training data

knn.fit(X_train,y_train)

# Predict the labels of the test data: y_pred

y_pred = knn.predict(X_test)

# Generate the confusion matrix and classification report

print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred ))
```

100. What is hold-out set?

Hold-out set is the set of dat a which have been separated from the training data set. This helps in understanding which hyper parameter value is working the best with an unseen data set called hold-out set

TECHVIDYA

TECHVIDYA

ISO 9001:2015 Accredited Company

"Stay Updated, Stay Ahead"