MCQ

Q1 a

Q2 b

Q3 -

Q4 A

Q5 c, ~~d~~

Q6 d

Q7 b

Q8 a

Q9 b

Q10 b

Q11 a

Q12 c

Q13 a, b, c

Q14 a, b

Q15 a

Q16 = C

Q17 = a

Q18 = a

Q19 = a

Q20 = b

Q21 = b

Q22 =

Q23 = a

Q24 = d

Q25 = d

Q26 = b

Q27 = a

Q28

SBQ

Q1 For missing values we can use multiple steps like we can fill the missing values by using mean, median, mode (depends on the context), and also we can also use the random method. i.e for avoiding maintaining the randomness of the data we can use the to fill the missing value by using random filling method i.e

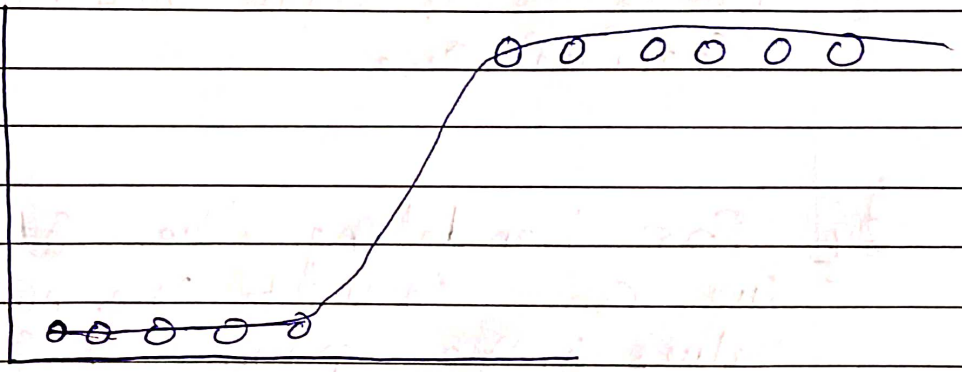
np.random.randint(mean - stdv, mean + stdv)

where stdv is the standard deviation

and if the numbers of the missing value is very low as compared to the shape of the dataset then we can simply drop that dataset.

And if the number of the missing value is very high then we can simply drop that column.

Q½ for making churn prediction model in telecommunications company we will use logistic regression model because our target column will be in classification i.e in "Yes" or "NO" because we will have only 2 values in our target column. whether the client will churn i.e "Yes" or it will remain on our platform i.e "NO".



logistic regression is a classification algorithem which helps to predict a binary outcome.

Date:
Page: Anand

Q3 For segmenting the Customers into different groups based on the purchasing behaviour we can use group by method so that we can differ different customes based on the purchasing behaviour like we can group by our Customes on the basis of salery, age, need etc so, the benefit we will that we can target the right people with so that our Company or Organization can grow, it also help us for getting the right audition from the Crowded One.

Q4 For handeling the Categorical variables we can convert the new Categorical values in the numerical one like if the dataset Contains a Column named Sex i.e male or female then we can Convert this Column into numerical by using 1 for male and 0 for female for this we can do manually by with the help of rename function or we can use the LABEL ENCODER from the SKlearn library. the syntax will be

```
from Sklearn.prepracessing import Label Encoder
LE = labelEncoder()
df["Colname"] = LE.fit_transform(df["Col-name"])
```

Date:
Page:

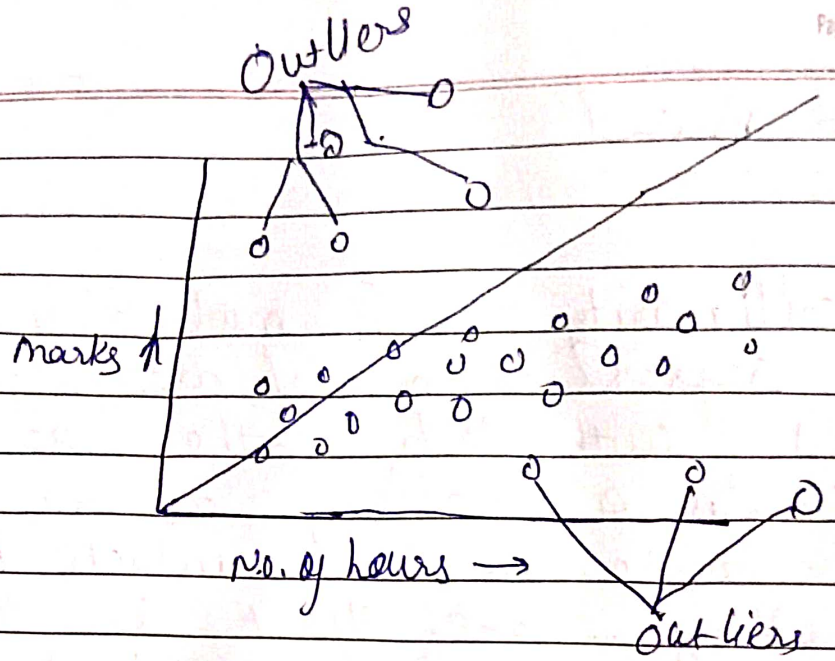
Concept Based

Q1 multicollinearity = In multicollinearity two regression model are highly Corelated with each other. so this will creates a problem as we will not be able to find which feature is making more effect on the target. So it will lead to reduce the accuracy of our model so, we can use correlation matrix for handling multicollinearity.

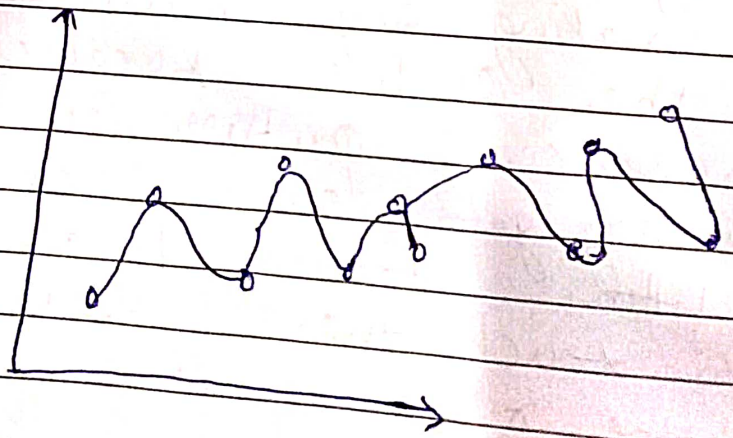
Q2 Outlier play a major role in making a effective prediction model because before making any model we should handle the outlier. because due to the presence of outlier our model's accuracy gets down because in training phase our model tries to touch these outliers so our the regression line gets deviated from its original path that leads to a bad model

we can detect outliers by using the
1. Scatterplot
2. box plot
3. Z-score method.

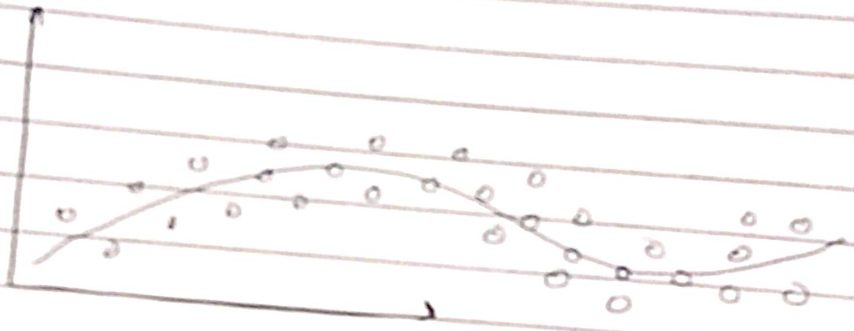
Q2



Q3 Overfitting a phenomenon in which our regression line tries to touch each and every datapoint present in dataset, it occurs by increasing the degree of the model.



Q4 For dealing with the non-linear relationship between predictors and target variable we can use polynomial regression as it give when we have (n-) graph non-linear relationship in our dataset then polynomial regression can be fitted



~~syntax~~ :-

~~from sklearn.linear_model import polynomialFeatures~~

~~poly = PolynomialFeatures(degree=n)~~

~~poly.fit(X_train, y_train)~~

It captures the non-linear pattern with the help of degree we provide in the polynomial feature.