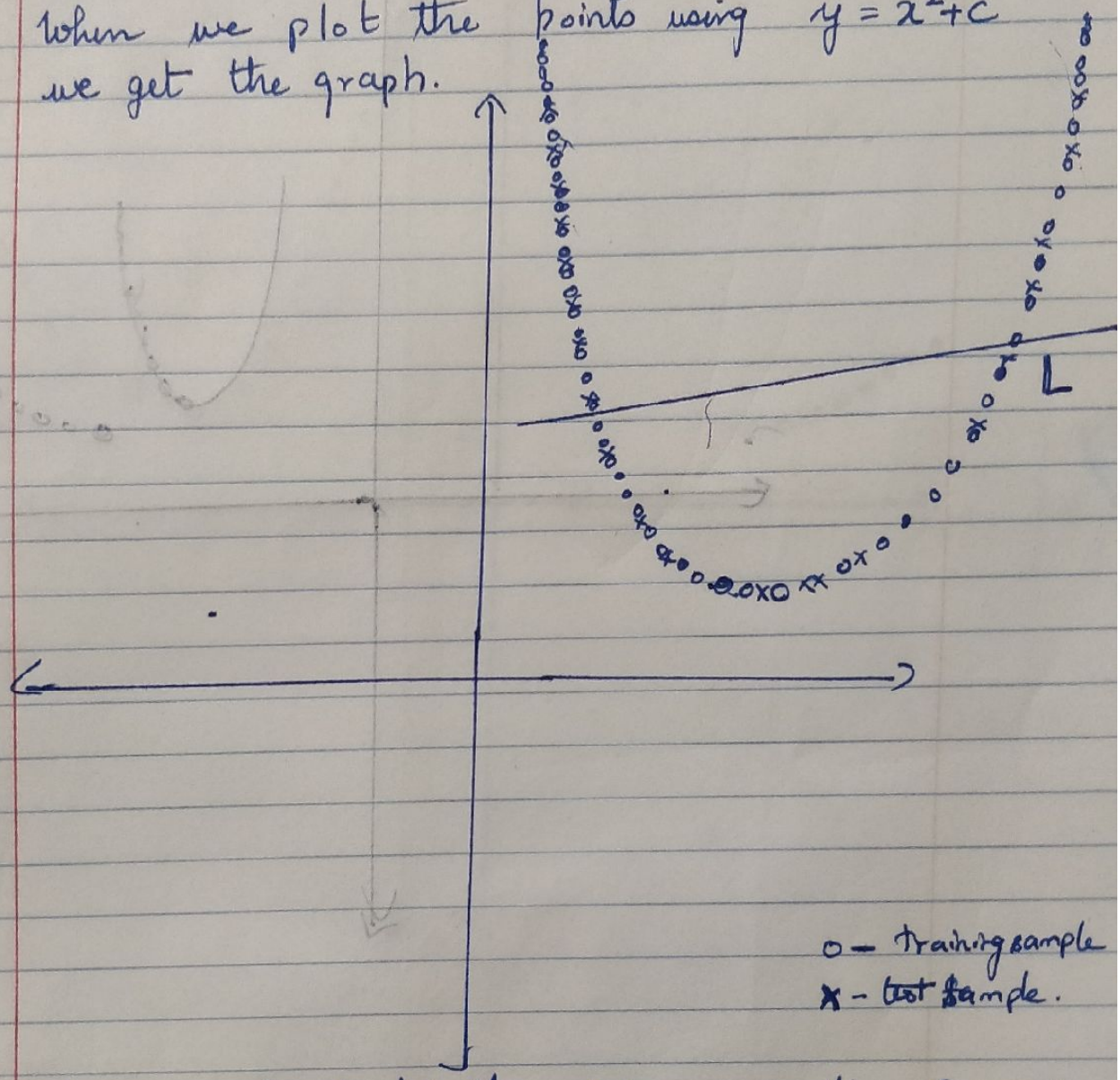
IST 597 - HomeWork - 5.

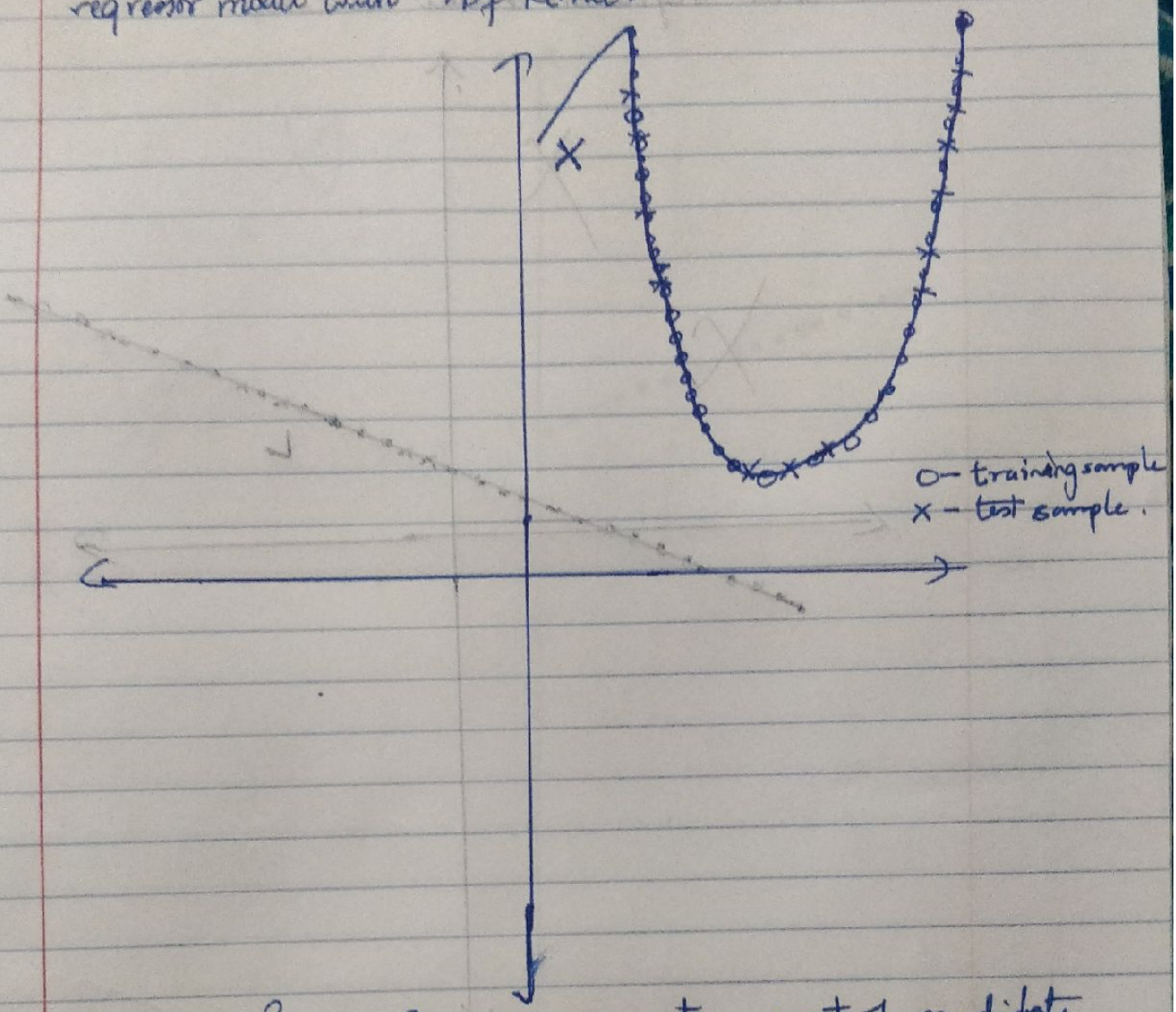
Problem 1:-

a. When we plot the points using $y = x^2 + c$ we get the graph.



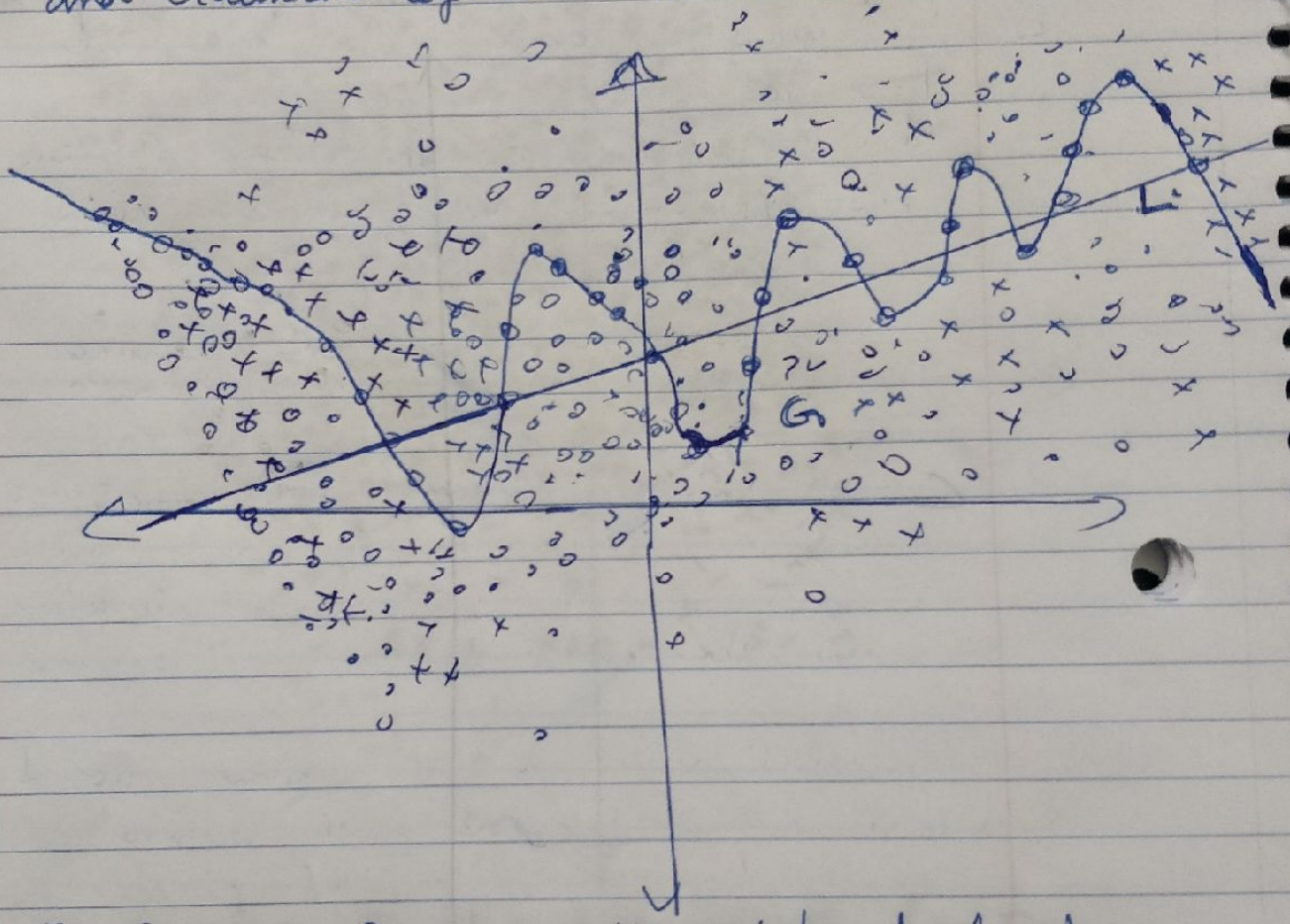
When we try to fit the data using Linear Regression model, the model does poorly. As the linear assumption does not hold, that the linear Regression assumes data - is linear. The model using the line data 'L' which can be seen as inaccurate to model the original data.

Now, use the same equation, to draw the data points on the graph and model the data using gaussian regressor model with rbf kernel.



Gaussian Process Regressor generates a set of candidate formulas that could have generated the observed data, and attempts to find the best match. It selects the best among them and proceeds to use that formula for predicting our data. We get a better model that is able to fit the data and predict values accurately Non-linear like X represents the model.

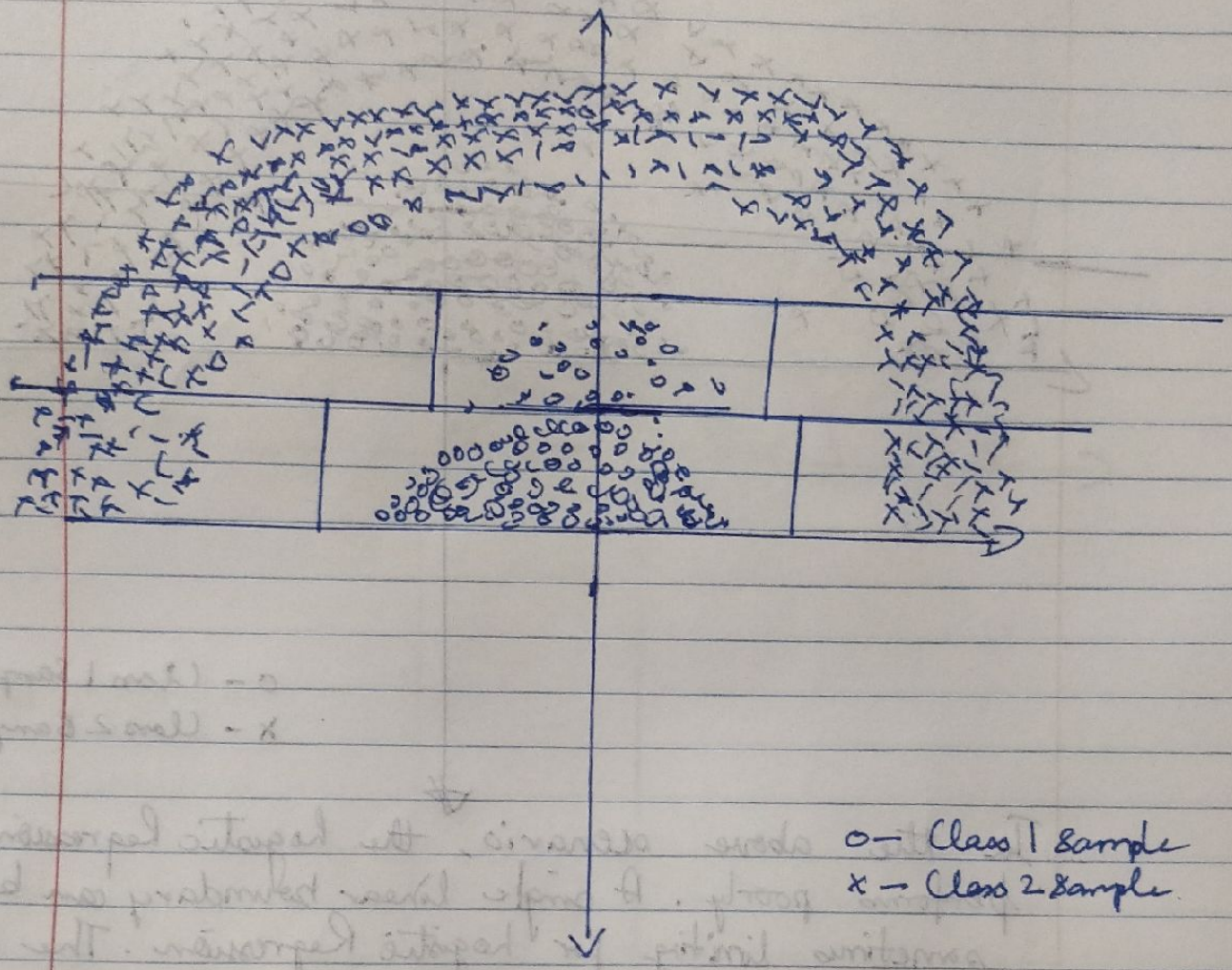
1b) Let us consider plotting the training samples 'o', test samples 'x' by combination of functions and use the models that is linear Regression and Gaussian Regressor to model the data :-



The Gaussian Regressor with rbf kernel clearly overfits the data and is unable to predict the relationship after the curvature it has drawn using only training data samples.

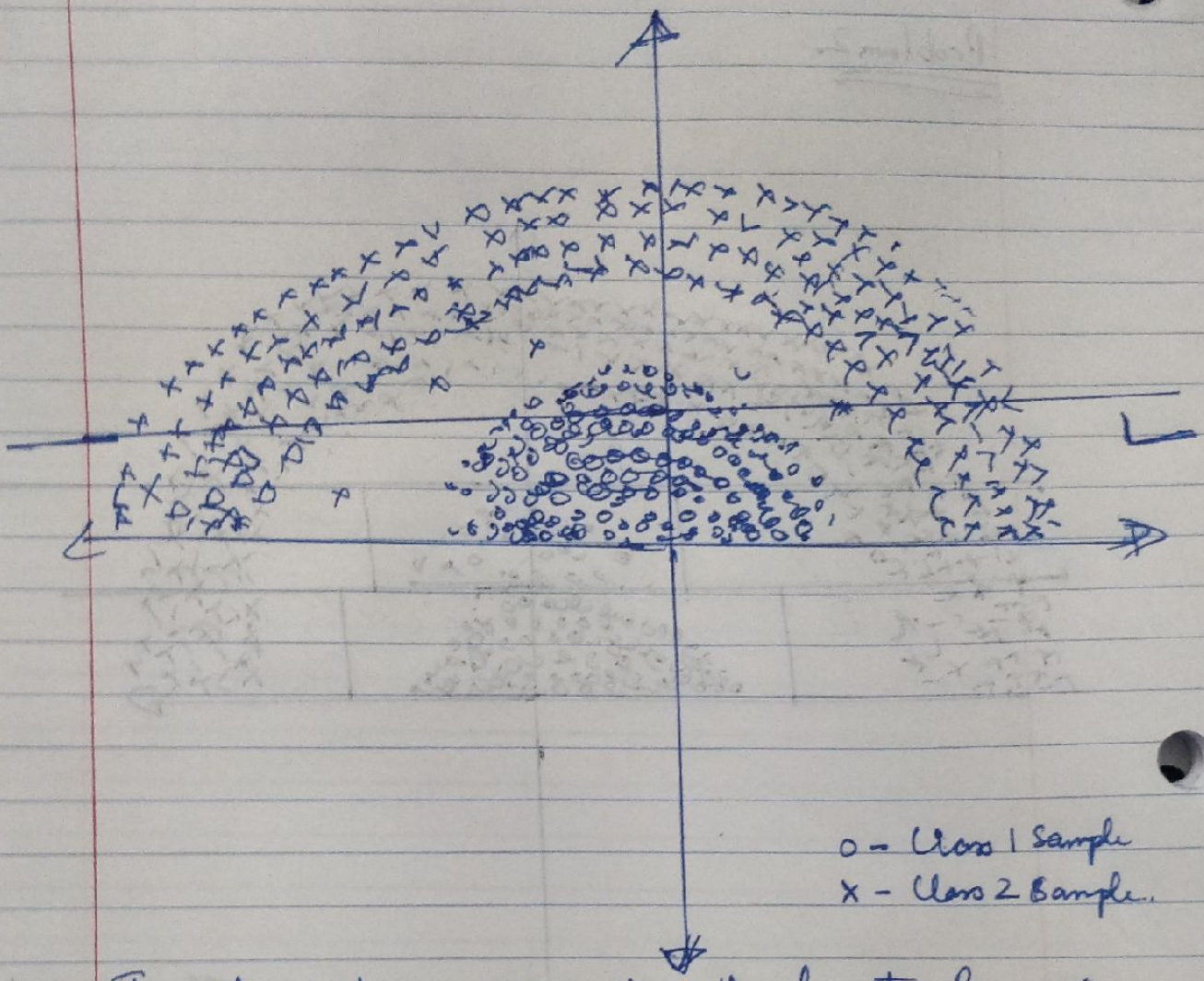
The linear Regression generalizes the data and assumes the data to be linear and provides more accurate results by not overfitting the data given. It is able to provide accurate results in predicting relationship in test samples as well. In case function like $y = \sin x$. Same thing might happen. The GP Regressor does not work out of the bounds of training data and overfits the training samples.

Problem 2



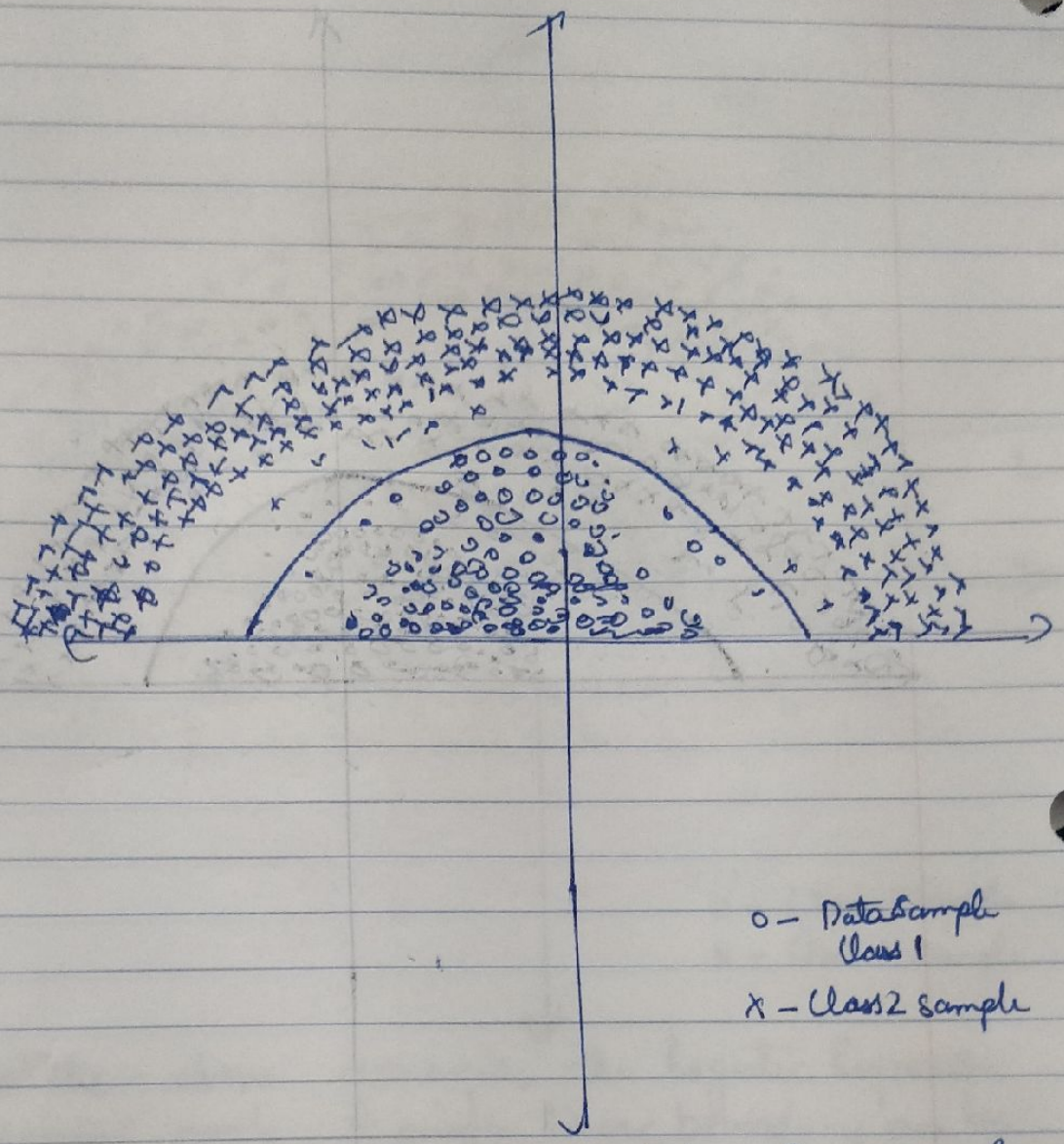
o – Class 1 Sample
x – Class 2 Sample

For the above dataset with two classes marked o and x, the Decision Trees bisect the space into smaller and smaller regions. The two classes are separated by a decidedly non-linear boundary. where data points within the boundary is attributed to particular class and outside the boundary are attributed to the other class. Hence Decision Tree Models the data quite well.



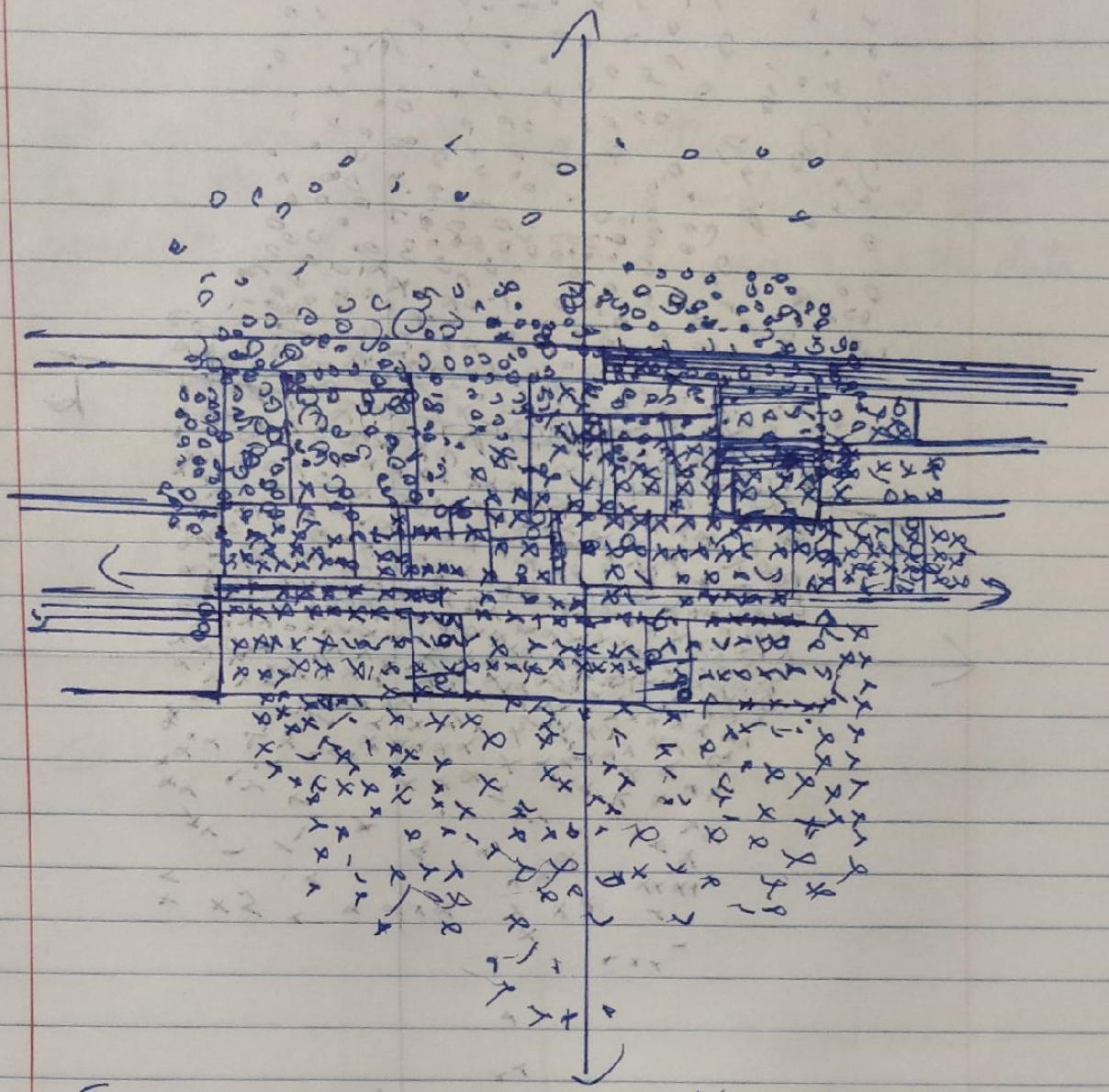
o - Class 1 Sample
x - Class 2 Sample.

In the above scenario, the logistic Regression performs poorly. A single linear boundary can be sometimes limiting for logistic Regression. The same dataset was used to model the logistic Regression based model.

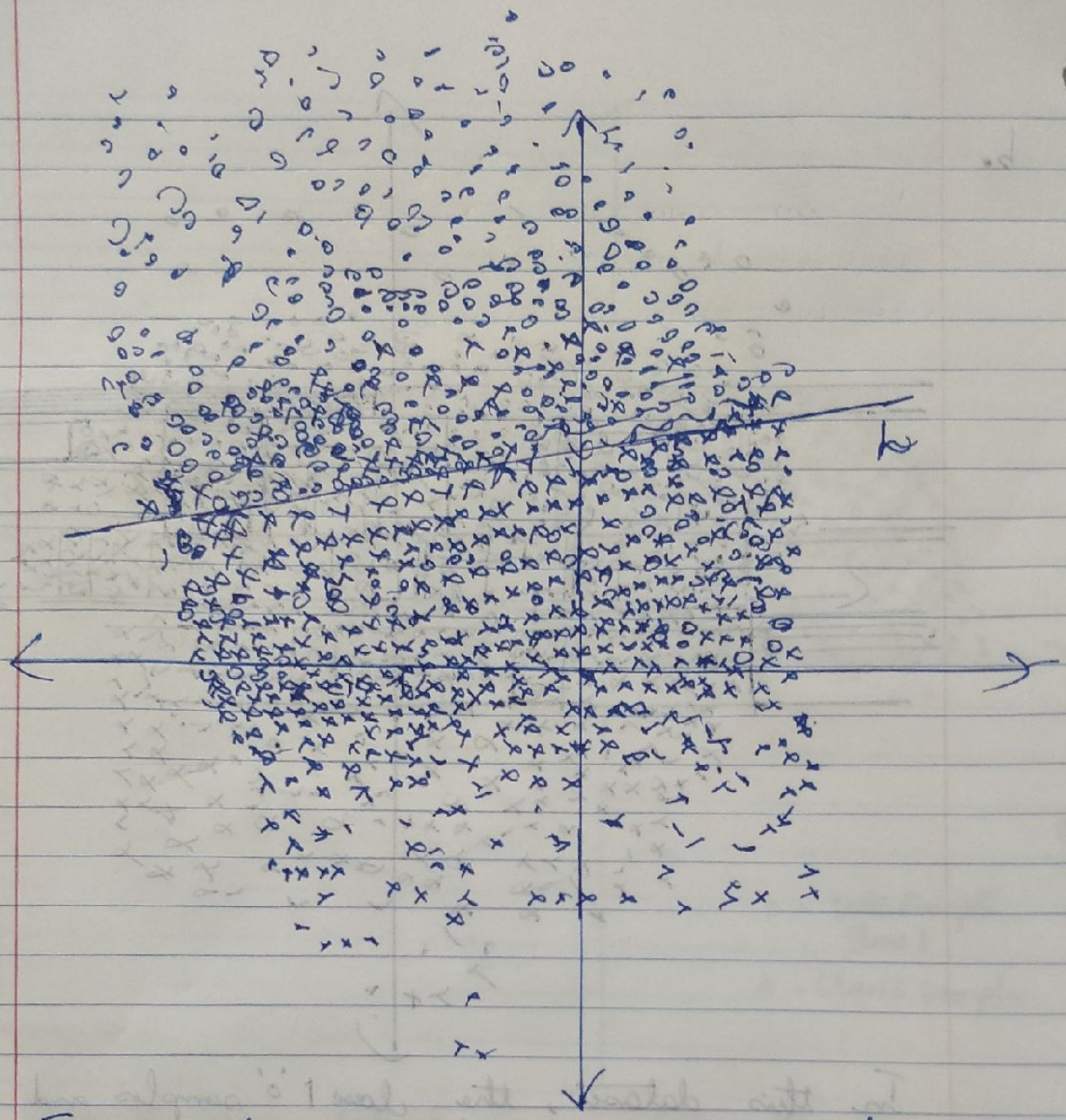


Using the same dataset again, if the Gaussian Regressor model is used it draws a clear decision boundary between Class 1 labels, Class 2 labels which is a non-linear boundary unlike Logistic Regression hence able to perform well in classification of samples.

b.



In this dataset, the class 1 'o' samples and class 2 'x' samples. there is no clear separation b/w samples, the decision tree succumbs to overfitting the training data. Hence, in this scenario we can mention that Decision Tree Classifier performs poorly.



For sample dataset used before, the Logistic Regression model is used to model the data and classify the Class 1 - 'O' and Class 2 - 'X' samples. The Logistic Regression draws a linear decision boundary and this approach generalizes better. Hence we can conclude that for this scenario the Logistic Regression model perform better because prediction confidence of this model can be computed in closed form for any co-ordinate values and can be more confident in prediction confidence values. This model does not overfit on the given data.

Problem 3

a. We have 11 features to predict the quality and type labels.
All the features are numeric values and the quality and type are categorical variables. of multi class type.

b. The Ordinal classification of multi class varicbles can be done using an Ordinal Classifier

Each ordinal class variable that has k multi class outcomes that are more than two can be transformed into k-1 binary classification problem where the first ordinal value, first class

$$V1 = 1 - P_r(y > V1)$$

where $P_r(y > V1)$ is the probability of the classes after first class. For the further classes

$$v_i = P_r(y > v_i) - P_r(y > v_{i+1})$$

where $P_r(y > v_i)$ is the probability of the classes. after v_i and $P_r(y > v_{i+1})$ is the probability of the classes v_{i+1} has after it.
For the last class $v_k = P_r(y > v_{k+1})$
where $P_r(y > v_{k-1})$ is the probability of classes after v_{k-1}.
Therefore the Ordinal nature of data is considered. As each class is predicted using the probabilities of other classes in consideration.
Two dimensional prediction task can be done by separately predicting each class of y value using a RandomForest Classifier to predict 'type' variable

The RandomForest Classifier builds the model on the data to predict 'type' variable apart from using the Ordinal Classifier to predict the 'quality' variable based on assumption that quality and type variables are mutually independent of each other. That is quality can be predicted
b. without knowing type or using type variable related signals and vice versa, thereby predicting them independently. Ordinal Classifier cannot be trained to predict variable with only two class labels.

c. Inequality of class labels can be dealt with by using the UnderSampling and OverSampling of data. In UnderSampling, the data points of the majority class that has more instances are removed and in OverSampling, the data points of the minority class labels are increased.

I have used the TomekLinks UnderSampling where datapoints of the majority class label are removed where only the data points that are near the datapoints of the other class labels are removed. Thereby the classifier can easily generate boundaries among the class labels as a more clearer boundaries are created in the data.

Bagging Classifier can be used to deal with Class-Imbalance in data as it perform better on class imbalance based data, which in turn uses RandomForest Classifier or Decision Tree Classifier. By default it uses Decision Tree Classifier.

d. The proposed scheme was implemented and the result has been submitted. Ordinal Classifier uses RandomForest Classifier. I also used BaggingClassifier object to build Ordinal Classifier.

e. I have used the feature_importances of the model to get the importances of the features based on which we can specify or can judge what features are important.

Apart from the importances, we can use the hypothesis test methods of Bon-Ferroni and Holm-Bon-Ferroni and Benjamini-Hochberg correction methods to predict the important features based on feature_importances as p-values, and $\alpha = 0.05$ or $\alpha = 0.5$.
Here, for the ordinal Classifier that is used to predict the quality, since there 3 class labels, two models (RandomForest) are generated.
Using Holm-Bon-Ferroni Correction method, at $\alpha = 0.50$ I got, as first, third, nineth, eleventh feature as important for the first classifier.
That is fixed acid, citric acid, ph, alcohol.
For the second classifier, I got first, third, sixth, seventh features as important.
fixed acid, citric acid, free sulphates, total sulphates features are important.
If I chose the common among them, fixed acid, citric acid is important features.
According to a ~~research paper~~ book however the alcohol content determines the quality of the wine. Book Name :- Nature and Origins of Wine Quality chpt, Ronald S. Jackson. "Wine Tasting", 2017.

Hence the 11th feature alcohol might be important. The methodes of hypothesis testing did not provide results at $\alpha = 0.05$ but gave results at $\alpha = 0.5$ only. (Holm-Bonferroni Method).

The feature_importances of each feature was around 0.05 - 0.15 hence just by looking at importances we could not get any clear indications as to which feature is important.

Problem -4

Overfitting in Machine Learning occurs due to training of the model to the data where noise is mistaken for signal and the model interprets noise as signal and model is unable predict the output for test data and Overfitting has occured.

In multihypothesis testing, the same problem occurs when multi hypothesis testing is conducted many times on the data and the model interprets the signal as noise and the noise is mistaken for signal. Hence multi-hypothesis testing should not conducted multiple times on the data.

Hence, intuitively, there exists a relationship between overfitting and multihypothesis problem where signal is interpreted as noise.