EAS 508 Exam 2

Name: Venkata Satya Surya Sai Vineet Atyam

UB Person number: 50419767

UBIT Name: vatyam

PART A:

1.) A single layer perceptron is a basic Neural Network with only one layer. Perceptron (feed forward neural) single layer computation is the sum of input vectors with the value multiplied by the appropriate vector weight. The output value presented will be the input to an activation function.

The input and output layers of a single layer perceptron are only two. Because it just has one layer, it is known as a single layer perceptron. It lacks the Hidden Layers found in Multilayer Perceptron. A multilayer perceptron is a feed-forward artificial neural network that produces a collection of outputs from a set of inputs. An MLP is a neural network that connects many layers in a directed graph, meaning that the signal route via the nodes is only one way. The MLP network is made up of three layers: input, output, and hidden. Each hidden layer is made up of several perceptron's, which are referred to as hidden layers or hidden units.

2.) The activation function determines whether or not a neuron should be triggered by generating a weighted sum and then adding bias to it. The activation function's objective is to induce non-linearity into a neuron's output. Because the gradients are given simultaneously with the error to update the weights and biases, activation functions enable back-propagation. Without an activation function, a neural network is just a linear regression model.

The activation function applies a non-linear change to the input, allowing it to learn and accomplish increasingly difficult tasks. It receives the prior cell's output signal and changes it into a form that can be used as input to the next cell.

Types:

1. Sigmoid Function 2. Tanh Function 3. ReLU (Rectified Linear Unit) Function

4.) Softmax Function 5.) Maxout etc

3.) Back-propagation is the process of fine-tuning a neural net's weights depending on the error rate (i.e., loss) achieved in the previous epoch (i.e., iteration). Proper weight tuning ensures decreased error rates, boosting the model's reliability by increasing its generalization. Backpropagation is used to train a multi-layered neural network to learn the right internal representations that will allow it to learn any arbitrary mapping of input to output.

The learning rate indicates how big of a step in the direction of the gradient we should take. If the learning rate is too low, it will only move a little amount in the opposite direction of the gradient or in small increments, and this might cause it to become stuck in local minima since it is not being as aggressive as it should be to escape them.

If the learning rate is too high, it may entirely overshoot and diverge, which is much worse. As a result, it's critical to practice choosing optimum learning rate.

4.) CNNs are typically used for image processing. The 'convolutional' part of the term refers to the way filters divide square patches of pixels in a picture. As a consequence, the model can mathematically represent important visual signals like textures and edges that aid in class differentiation. Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs.

A convolutional network's initial layer is the convolutional layer. While further convolutional layers or pooling layers can be added after convolutional layers, the fully-connected layer is the last layer. The CNN becomes more detailed with each layer, detecting larger areas of the picture. Earlier layers concentrate on basic elements like colors and borders. As the visual data travels through the CNN layers, it begins to distinguish bigger components or features of the item, eventually identifying the target object.

5.) If our model performs significantly better on the training set than on the test set, it is likely that it is overfitting. We can ensure the prevention of overfitting by any of the following methods:

1. Cross-validation: Cross-validation is an effective tool for avoiding overfitting. Create several mini train-test splits using the initial training data. To fine-tune the model, use these splits. The data is partitioned into k subsets, or folds, in standard k-fold cross-validation. The method is then repeatedly trained on k-1 folds, with the remaining fold serving as the test set. With cross-validation, hyperparameters may be fine-tuned using only the original training set. This permits the test set to remain a completely unknown dataset for picking the final model.

2. Train with more data: More data can help algorithms discern the signal more accurately and can prevent overfitting.

3. Feature selection and Dimensionality reduction: Simple models that generalize effectively, as well as input data with few input variables, are preferred. This is especially true for linear models, where the number of inputs and the model's degrees of freedom are frequently linked.

4. Early Stopping: New iterations refine the model until a specified number of iterations have been completed. The model's capacity to generalize declines beyond that point, as it begins to overfit the training data. Halting the training process before the learner reaches that stage is referred to as early stopping.

6.) If the dataset contains high positive or negative attributes, the model's performance is likely to be harmed by a problem known as Multicollinearity. When one predictor variable in a model can be linearly predicted from the others with a high degree of accuracy, this is known as multicollinearity. They introduce redundant information as a result, the results may be distorted or erroneous. The structure and operation of a classifier are unaffected by correlation phenomena. They do, however, have a detrimental impact on individual predictions, which can affect the overall quality of the findings.

By removing the correlated features, it improves the model's accuracy and speed since the dimensions of the dataset is being reduced (Curse of Dimensionality) and hence can provide better results with generalised models and reducing overfitting.

7.) Sampling and fitting several models many times is one of the simplest ways to quantify the uncertainty of your measurements. This won't account for any sample disparities in our data when compared to the actual value but it will reduce the possibility of a "lucky split" when splitting our training and validation data.

In case of classification, rather than just calculating the accuracy of the models, it is preferred to see how many of the limited data points are measured in the true value, that is the portion of results in true positives and true negative and also the fraction of data in the false positives and false negatives range, in case there are more points in the false regions, it indicates high uncertainty and hence the models are not performing well.

8.) The models prefer to classify results into the class with the most instances, the majority class, due to the disparity of classes in the variables, providing the misleading impression of a highly accurate model. The prediction models are harmed by their failure to forecast unusual occurrences, the minority class, and their deceptive accuracy. When models are trained on imbalanced datasets, they frequently perform poorly when they are forced to generalize (predict a class or classify unseen observations)

The algorithm is biased towards one class since it gets disproportionately more samples from that class. It fails to comprehend the fundamental patterns that allow us to discern classes and does not learn what makes the other class "different. "The algorithm learns that a certain class is more prevalent, therefore there is a "natural" inclination to gravitate toward it. When this happens, the algorithm is prone to overfitting the majority class. Models would score well on their loss functions only by guessing the majority class.

Methods to solve this problem:

1. Resampling: By under sampling the majority data or by oversampling the less prevalent classes can help reduce the problem since it reduces the bias and avoids skewing the data towards the more prevalent class.

2. Collecting more data of the lower class can prevent this problem.

3. Using penalized models: There are penalized versions of many algorithms. Typically, algorithms treat all misclassifications equally, therefore the goal is to penalize minority-class misclassifications more severely than majority-class misclassifications. Mistakes committed during training incur an additional penalty, but these penalties should theoretically aid the model in improving the attention provided to the minority class.

PART B:

A.) Property1 Regression Analysis

MODEL	TRAIN R-SQUARED	TEST R-SQUARED	RMSE TRAIN	RMSE TEST
GAUSSIAN PROCESS REGRESSION	0.371628	0.1524	58.2	65.32
SUPPORT VECTOR REGRESSION	0.78675	0.1354	59.5013	68.8155
RANDOM FOREST REGRESSION	0.810173	0.34028	34.037	55.0178
MULTIPLE LINEAR REGRESSION	0.1773	0.10026	64.802	74.2824
LASSO REGRESSION	0.44826	0.1485	67.0675	66.2624
RIDGE REGRESSION	0.3952	0.0365	68.723	65.9984

Model	Train Accuracy	Test Accuracy
Artificial Neural Network	0.8625	0.6167

Accuracy:

Models like Support Vector Regression, Random Forest regression models have a very good training R2 score as compared to other models which have only about 0.2 Training R2-score.

In terms of testing the data, the random forest model works better than the other models though the R2 score is still less. This is due to the fact that the dataset is very erratic with very high correlation between many features and also there are same columns data with just the change in column name (Feature5 and Feature12 are exactly the same columns). This is causing the models to perform with a lot of bias and since there is very less data, the models are not working as expected.

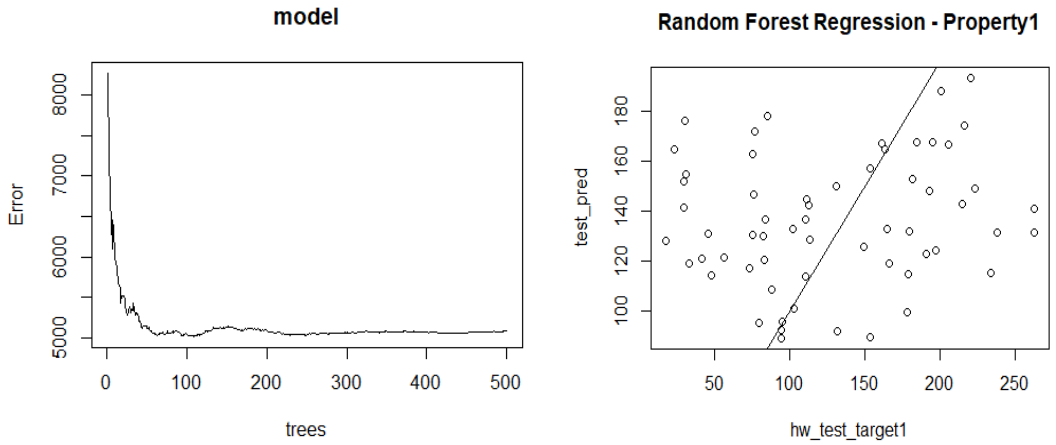
But going with just the raw accuracy metric, models like Artificial Neural Network modelling on the median data as a classifier has obtained good training accuracy and testing accuracy of 86.25% and 61.67% respectively. It is good considering the fact that the model has less data points and hence it is difficult to train the data.

Maximum Accuracy Models:

1. Random Forest Regression

2. Artificial Neural Network

Random Forest Image



Random Forest feature importance

Feature1 3.8562141 Feature2 3.2860379 Feature3 3.8827092 Feature4 7.4926959

Feature5 8.4729035 Feature6 5.6260191 Feature7 13.1388972 Feature8 3.8289728

Feature9 1.5197815 Feature10 2.3358746 Feature11 4.3435231 Feature12 7.4429508

Feature13 8.2253487 Feature14 4.0891350 Feature15 5.9895144 Feature16 4.7952726

Feature17 6.4845298 Feature18 0.9635018 Feature19 0.5929602 Feature20 -0.7651406

Interpretability:

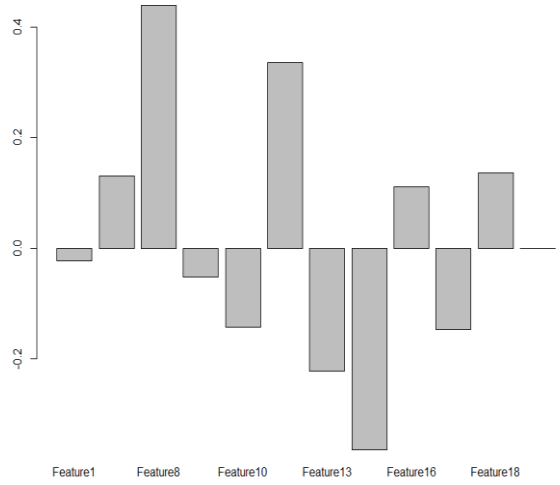
Linear Regression models like Multiple Linear Regression and Lasso Regression are good for interpretability though the accuracy is less, they show the features necessary for prediction in an easy way and we can identify the coefficients easily.

In the given dataset there are many similar columns with very high correlation and hence these features need to be eliminated during the analysis, thus choosing the features based on Principal Component Analysis and Lasso Regression techniques is very efficient.

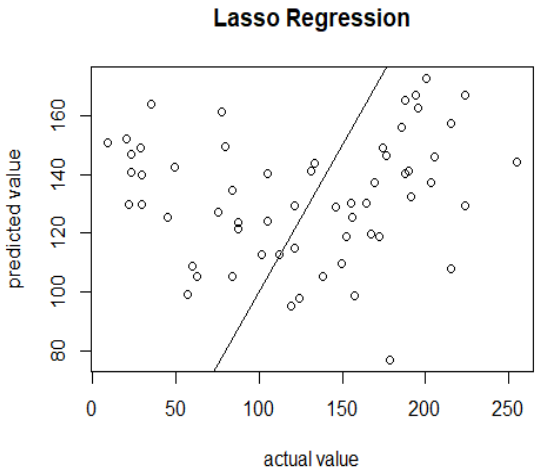
Models For Interpretability:

1.) Lasso Regression

2.) Principal Component Analysis



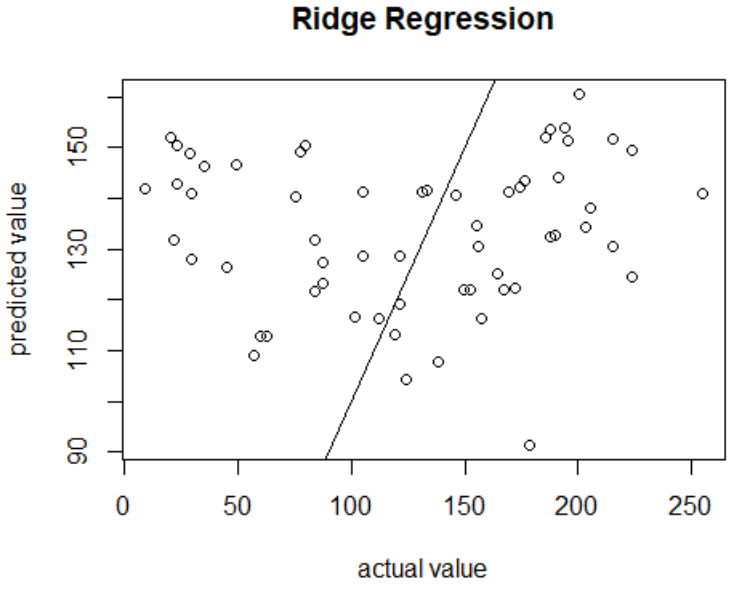
VIP Barplot using PCA



Lasso Regression Prediction plot

s0 (Intercept) 2.379519e+02

Feature1	-1.862222e+02	Feature6	-1.895967e-01	Feature7	-9.093734e+00
Feature8	-1.070836e-03	Feature9	-1.277729e+01	Feature13	7.469493e+01
Feature14	3.476105e-02	Feature15	3.682733e+00	Feature16	1.463444e+01
Feature17	1.315495e-01	Feature18	5.356973e+00	Feature20	-6.919143e-01



Prediction Using Ridge Regression

Robustness:

The model that is more robust compared to others is mostly the random forest regression model as it did not completely overfit the data and has better metrics compared to other regression models. The model works better in view of unseen data effectively and hence can be used for model predictions for predicting real world data.

Model for Robustness:

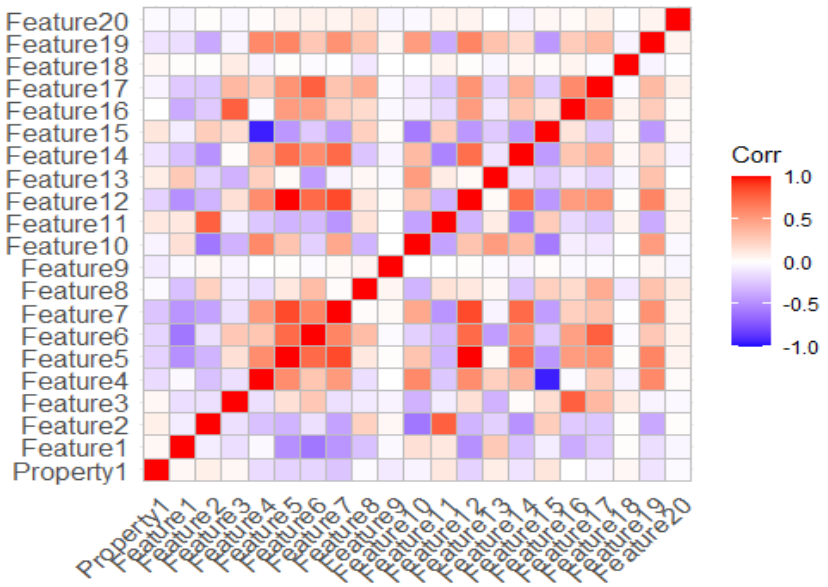
1.) Random Forest Regression

B.) The interpretability here can be made by using both the non linear models like random forest and support vector regression models and linear models like Lasso regression, though these models do not have better accuracies, we see that feature reduction has been made through Lasso model and also regression analysis could be done through Random forest with fair amount of training accuracy and hence its feature importance can be used for interpreting the important features in the model.

If the aim is just to identify important features, we can use the linear models whereas if we want a model with better accuracy, non linear models can be used but at the cost of interpretability.

C.) From the analysis through Principal Component Analysis and through Lasso regression, we can select lesser features with less correlation between them.

The features I've selected through PCA are Feature1, Feature6, Feature8, Feature9, Feature10, Feature13, Feature15, Feature17. This is due to the fact that the model is highly correlated with columns also being the same(column **Feature12 and Feature5** , **Feature4 and Feature 15**).



Hence there is a need to reduce the features and predict them else they will skew the results, and also many features seem to be related to each other by a ratio and also contain main 0 values and hence this causes the models to bring in bias and will not be able to predict the results as required.