**WINTER OF CODE 7.0**

**REPORT**

* **ALGORITHMS USED:**
* Linear regression for multiple features
* Logistic regression for multiple features
* Classifier model: one vs one
* K means on both supervised and unsupervised learning
* Softmax regression

1. **LINEAR REGRESSION TRAINING SET:**

* Applied linear regression on the model
* Plotted the cost vs iteration graph to determine if the model converges and reaches the global minimum
* Determined the cost on training set and cross validation set to determine if the model has high bias or high variance or is the best fit model
* Computed R squared value of the model to determine its performance

1. **POLYNOMIAL REGRESSION TRAINING SET:**

* Computed the different combinations of the powers of features that are possible and determined the weights for all possible combinations
* Determined the best fit model based on the r squared value of each model

1. **BINARY CLASSIFICATION TRAINING SET:**

* Applied logistic regression on the training set
* Regularized the model to optimize it for overfitting
* Set a threshold based on accuracy to assign binary labels to the dataset
* Plot a graph of cost vs iteration to make sure that the model converges and isn’t overshooting
* Calculated the accuracy, precision, recall and f1 score to determine if the model is performing as expected

1. **MULTICLASS CLASSIFICATION TRAINING SET:**
2. **ONE VS ONE:**

* Used one hot encoding to create columns for each unique class that reported 1 for the respective class and zero for all the other classes.
* Trained all the classes on logistic regression to determine the respective weights and biases so obtained.
* Calculated the probability of a given training example of belonging to the given class and reported the one with maximum probability as the predicted class
* Calculated the f1 score and accuracy to determine the functionality of the given model

1. **SOFTMAX:**

* Used one hot encoding to create columns for each unique class that reported 1 for the respective class and zero for all the other classes
* Rescaled the features to optimize the data to prevent overflow
* Defined the cost function and the gradient function to compute the weight matrix w.r.t all classes and features.
* Trained the model using gradient descent and computed accuracy and F1 score to determine its functionality

1. **K MEANS:**

* Initialized a no(equal to the total no of classes) of random centroids and running k means on those centroids.
* Recomputing the centroids till the recomputed centroids become equal to the one computed previously and computing the cost .
* Repeating the same for over a 100 iterations and determining the final centroids based on whichever one has the lowest cost.
* Repeating this process around 20 times to get a variety of centroids with very low cost
* Iterating through the obtained set of centroids to get the one with highest accuracy and f1 score

1. **UNSUPERVISED LEARNING TRAINING SET:**

* Used k means