ML Assignment I

Shreyank Buddhadev (MT2021130)

October 2021

1 Classification Problem

Given a dataset of extracted features of bank notes images task is to classify whether a bank note is authentic or not.

1.1 Dataset Overview

Feature	**Type**	**Summary**
variance	numerical	variance of Wavelet Transformed image
skewness	numerical	skewness of Wavelet Transformed image
curtosis	numerical	curtosis of Wavelet Transformed image
entropy	numerical	entropy of image
target	categorical	**1** is for authentic note and **0** for not.

1.2 Data Visualization

- Imbalanced dataset pose a challenge of poor modeling. To better visualize imbalance in dataset fig 1 depicts count plot for feature *target*.

  As depicted in figure 1 data has slight imbalance.

- Figure 2a and 2b depicts boxplot and distribution plot for feature *variance*

- Figure 3a and 3b depicts boxplot and distribution plot for feature *skewness*

- Figure 4a and 4b depicts boxplot and distribution plot for feature *curtosis*

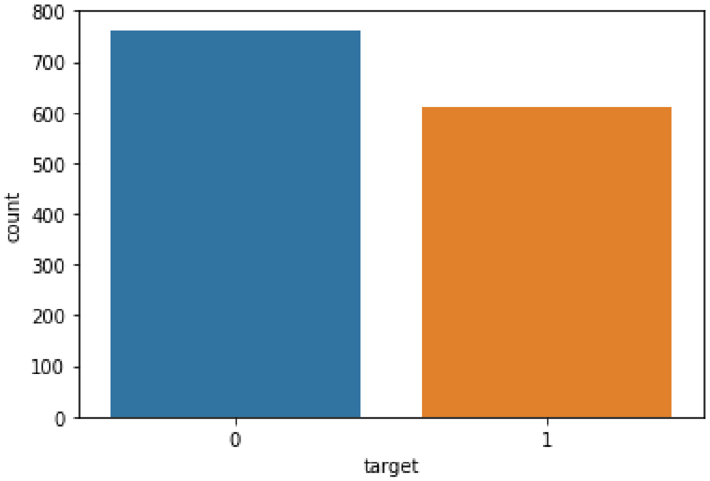
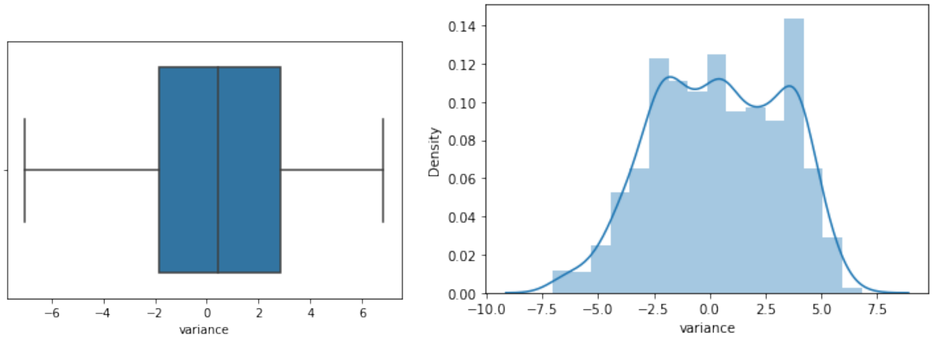


Figure 1: count plot for target features



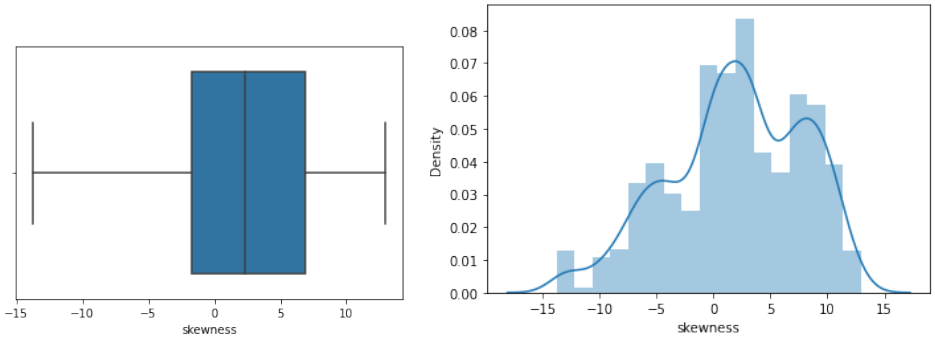
(a) box plot for variance

(b) dist plot for variance

Feature *curtosis* has right skewed data. Also, from the plots we can infer that there is possibility of outlier in the feature *curtosis*.

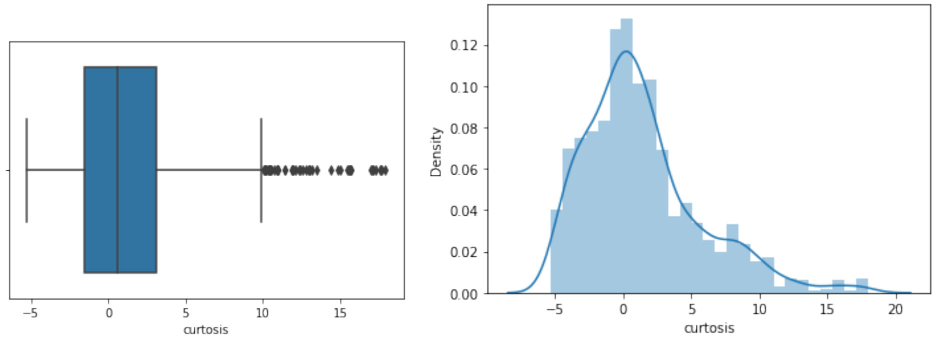
- Figure 5a and 5b depicts boxplot and distribution plot for feature *entropy*

  Feature *entropy* has left skewed data. Also, from the plots we can infer that there is possibility of outlier in the feature *entropy*.



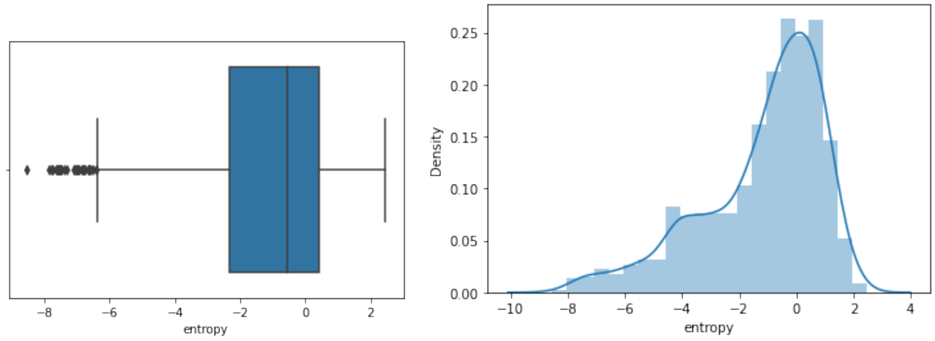
(a) box plot for skewness

(b) dist plot for skewness



(a) box plot for curtosis

(b) dist plot for curtosis



(a) box plot for entropy

(b) dist plot for entropy

1.3 Data Pre processing

Provided data had several inconsistencies to overcome those below mentioned steps are applied on data set. To handle inconsistencies there were several possible alternatives, this section contains approach which were best suited to achieve high accuracy.

1.3.1 Filling NaN values

Features *curtosis* and *entropy* had some outliers which contributed to higher standard deviation. Hence, to replace data with central values NaN values for all features were replaced by corresponding feature's median value.

1.3.2 Outlier removal

Feature *curtosis* and *entropy* have few data points as outliers. But, since logistic regression does sigmoid squshing it doesn't had much impact. Analysis for effect of outliers on accuracy is added in analysis section.

1.3.3 Normalizing or Standardizing data

The goal of normalization is to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values. Similarly, Standardizing the features around the center mean 0 and a standard deviation of 1 is important when we compare measurements that have different units.

- Normalization process $x_i = (x_i - min)/(max - min)$
- Standardization process $x_i = (x_i - \mu)/\sigma$

here x_i = data point, μ = mean, σ = standard deviation

Here, feature data is around similar scale. Yet, to be computationally efficient normalization was applied.

1.3.4 Linearity and Multi collinearity

Below heatmap shows linear independence between columns. Here, skewness and curtosis data is highly corelated.

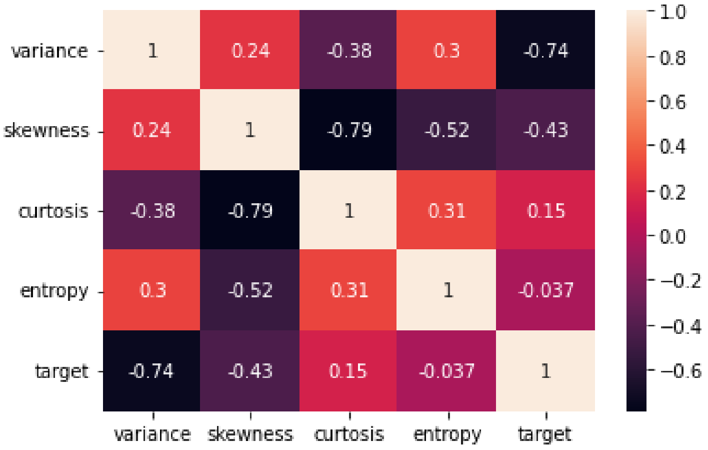


Figure 6: heat map for features

1.4 Multivariate Gaussian Naive Bayes

From fig 6 we can infer that since curtosis and skewness column has high corelation we can choose *variance, skewness* and *target* column for multivariate gaussian.

1.5 Logistic Regression loss function plots

- Loss function graph for logistic regression using gradient descent

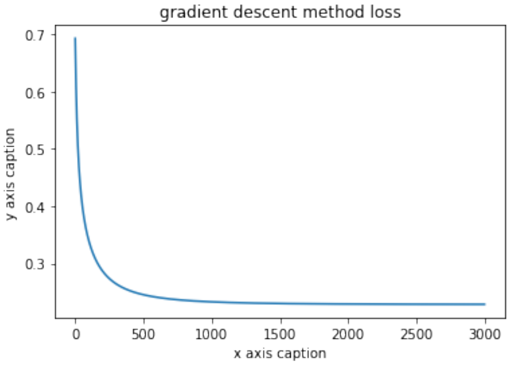


Figure 7: loss graph per iteration for gradient descent

- Loss function graph for logistic regression using newton's method

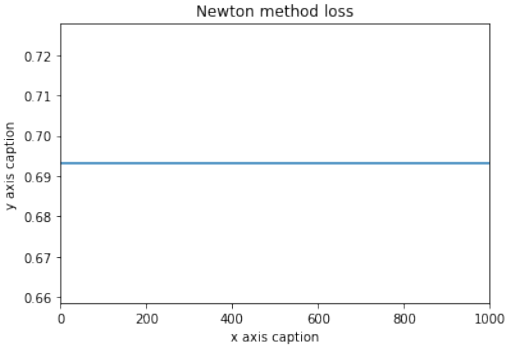


Figure 8: loss graph per iteration for newton's method

1.6 Accuracy Analysis

For logistic regression

type	**accuracy**
newton's method - training	89.79033728350045
newton's method - testing	90.54545454545455
gradient descent - training	89.88149498632635
gradient descent - testing	89.45454545454545

For gaussian

features taken	**accuracy**
all-training	81.03919781221514
all - testing	77.818181818181815
variance - training	81.13035551504102
variance - testing	79.63636363636364
skewness - training	61.16681859617138
skewness - testing	62.909090909090914
curtosis - training	49.68094804010939
curtosis - testing	56.36363636363636
entropy - training	55.51504102096627
entropy - testing	47.63636363636364

2 Regression Problem

Given dataset of garment workers productivity predict actual productivity of workers.

2.1 Dataset Overview

Below table depicts feature names and type of dataset features.

feature	**type**
quarter	categorical
department	categorical
day	categorical
team	categorical
targeted_productivity	numerical
smv	numerical
wip	numerical
over_time	numerical
incentive	numerical
idle_time	numerical
idle_men	numerical
no_of_style_change	numerical
no_of_workers	numerical
actual_productivity	numerical

Provided dataset contains **14** features and **1160** data points.

2.2 Data Preprocessing

Provided data had several inconsistencies to overcome those below mentioned steps are applied on data set. To handle inconsistencies there were several possible alternatives, this section contains approach which were best suited to achieve least error.

2.2.1 Filling NaN values

Below depicted features has provided NaN values.

feature	**Number of NaN**
targeted_productivity	384
wip	703
over_time	371
quarter	353
day	375

- *wip(work in progress)* column has 61% NaN values. But further analyzing dataset below were the findings:
  - *wip* values for rows having *department* ***finishing*** are NaN and that accounts for 71% of total missing values
  - Assuming that, *department* ***finishing*** might have some waiting time before work in progress. All NaN's with *department* ***finishing*** are replaced with zeros.
  - for remaining 21% of missing values has *department* ***sweing***, statistical analysis for which is given below,

    mean 1150.280088

    min 7.000000

    median 1035.000000

    max 23122.000000

    There are some outliers as depicted in the figure 9. Also, standard deviation is high therefore mean might be deviated from central value because of outliers.So, remaining NaN's are replaced with median value of *wip* with *department* ***sweing***.

- *targeted_productivity* feature has numerical data. To handle NaN values in it we've statistically analyzed it and found below results.
  - mean: 0.732668
  - median: 0.750000
  - min: 0.350000
  - max: 0.800000
  - Since, mean and median has not much difference between them for *targeted_productivity* column. All Na N's for feature *targeted _productivity* are replaced with mean value of available *targeted _productivity* values.

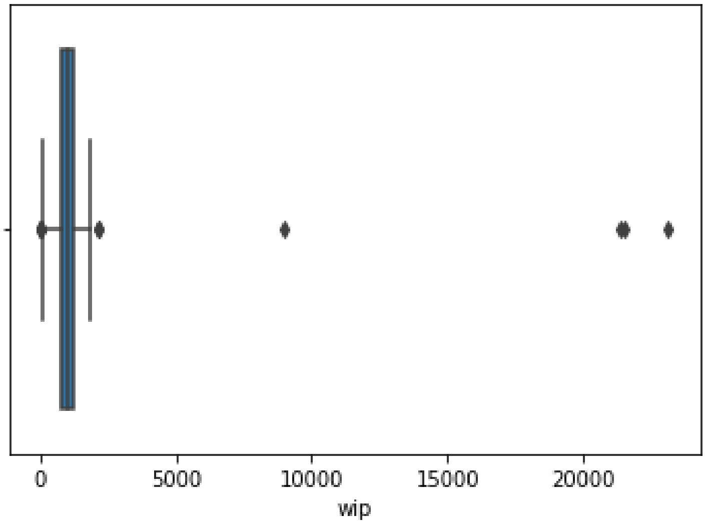


Figure 9: wip box plot

- Feature *overtime* has numerical data. Below is statistical analysis for the feature
  - count 789.000000
  - mean 4612.053232
  - std 3390.363700
  - min 0.000000
  - median 4080.000000
  - max 25920.000000
  - *overtime* feature's data has some large values which contributes to higher deviation. But, common sensibly we can think that all the NaN's corresponds to 0 which translates that each department workers might not be required to do overtime.
- Feature *quarter* and Feature *day* has categorical data. To fill NaN's several possible approaches can be
  - to fill with most frequent value
  - to simply ignore NaN's

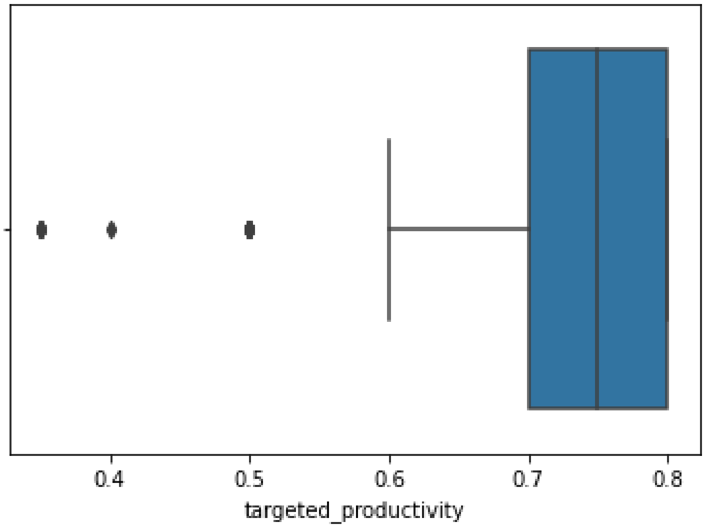


Figure 10: targeted productivity box plot

- to assign a new categorical value

- Analyzing above possibilities, NaN Values for Feature *quarter* and *day* were replaced by new categorical values.

2.2.2 Encoding features

To fit categorical data into numerical model encoding has been applied.

- Feature *quarter, department* and *day* has categorical data which were encoded.
- Feature *quarter, day* had multiple unique values. So, to avoid adverse impact of it on model it was encoded using one hot encoding.
- Feature *department* had only 2 values. So, it was encoded using label encoder.

2.2.3 Normalizing or Standardizing data

- Normalization process $x_i = (x_i - min)/(max - min)$

- Standardization process $x_i = (x_i - \mu)/\sigma$

  here x_i = data point, μ = mean, σ = standard deviation

- Among above two approach, least square error difference was negligible.

2.2.4 Linearity and Multi colinearity

Below heatmap shows linear independence between columns. From the heatmap we can observe that *no_of_workers* has high co relation with *smv*.

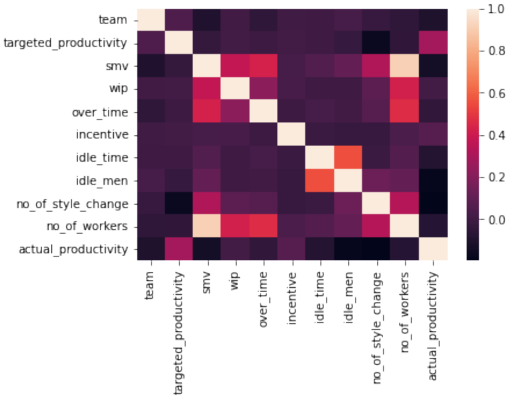


Figure 11: heatmap for features of given dataset

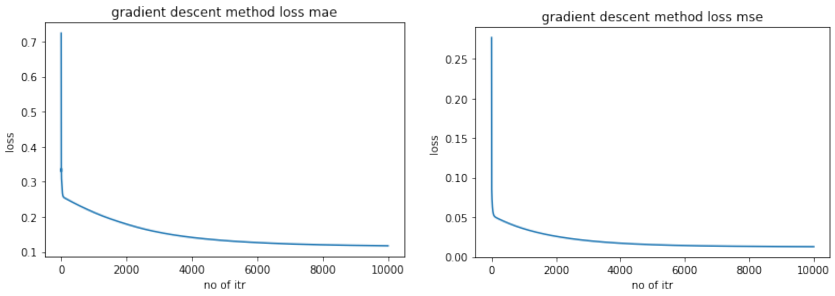
2.3 Linear Regression loss function plots

figure 12b, 12a, 13a and 13b contains graph for gradient descent mean sqaure error, gradient descent mean absolute error, newton method mean absolute error and newton method mean square error respectively.

2.4 Accuracy Analysis

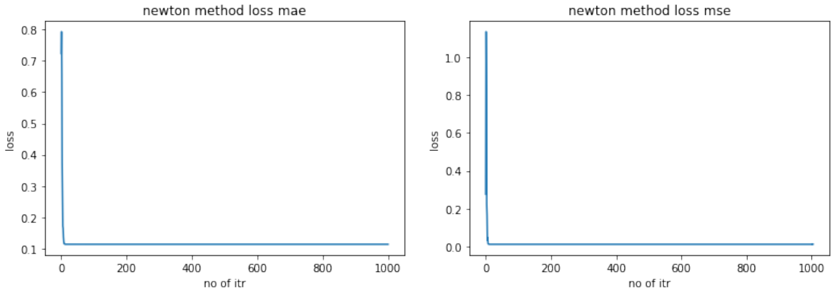
For linear regression

type	mean square error	mean absolute error
gradient descent-training	0.01283256	0.11742528590710881
gradient descent-testing	0.01108155	0.1157965155900566
newton's method-training	0.01201337	0.11387367756711504
newton's method-testing	0.01004012	0.11076702506485948



(a) gradient descent mae

(b) gradient descent mse



(a) newton method mae

(b) newton method mse

For newton method 1000 iteration were taken while for gradient descent 10000 were taken.