PROBLEM STATEMENT

Explore whether Winsorizing (replacing extremely high values by predetermined upper/lower bounds) can improve the accuracy or computational effort of a single-node classification algorithm (e.g., perceptron), experimenting with any non-trivial two-class data set.

ANALYSIS

1) DATASET

The dataset “Breast Cancer Wisconsin (Original) Data Set” selected for this experiment can be found on UCI repository using the below source link where a detailed description is available.

Source: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29

This cancer data set was created by the physician Dr. William H. Wolberg at University of Wisconsin Hospitals, Madison. The data set consists of 10 attributes and a class label (with value 2 for benign and 4 for malignant). Of the 10 attributes the first attribute is a sample code id number hence, it is not fed into the algorithm either during training or testing. Remaining 9 attributes have integer values in the range of 1 to 10.

For this experiment the dataset with 699 instances is divided randomly into training and testing sets with 70% (490) and 30% (209) of the instances.

2) CODE

Libraries used in this experiment: psych (for winsorization), nnet (for single-node perceptron).

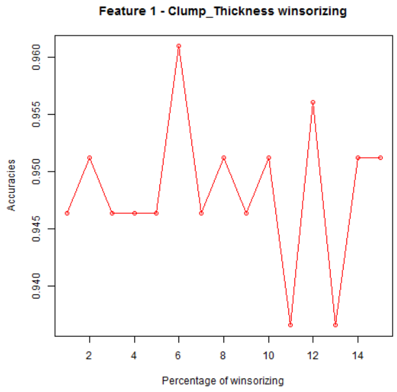
A single-node classifier is fed with the training data to build the model and then the model is tested with the testing data. Every time the data is fed to the algorithm either a feature by feature winsorizing is achieved. Hence for 9 features and 1 to 15 percentage of winsorizing at the end of the experiment we receive 9*15 = 135 accuracies which are plotted and saved.

3) EXPERIMENT AND RESULTS

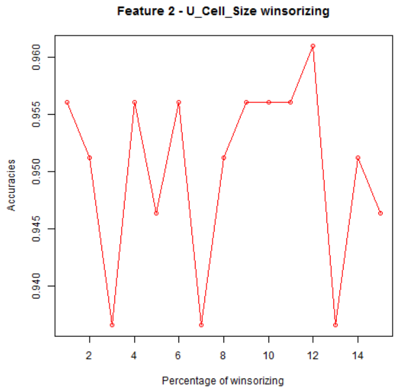
A single layer neural network with only one neuron which randomly selects weights and bias is used to compute the output for the given input vector. The perceptron is trained with 70% of the dataset to learn to classify the instances into benign(class-2) or malignant(class-4). Then the single layer perceptron model is tested with the testing data which is 30% of the original data not used for training. While testing the model doesn’t know to which class an instance belongs and the model classifies the instance either into a class 2 or 4. Once the model completes classifying the test instances accuracy is calculated using below formula.

$$\text{Accuracy} = \frac{True\ Positive + True\ Negative}{True + False}$$

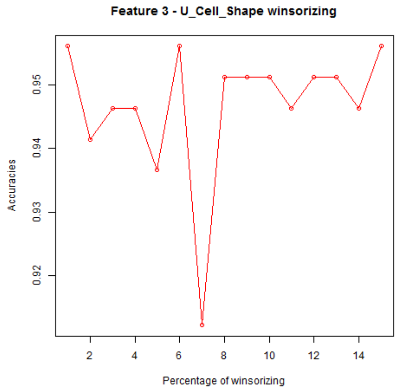
For winsorizing the winsor function from psych library is used and below strategy has been adopted. Winsorize each feature by a percentage of 1 to 15 and accuracies are calculated. Graph plots from 1 to 9 represent winsorizing features 1 to 9 respectively.



(1)

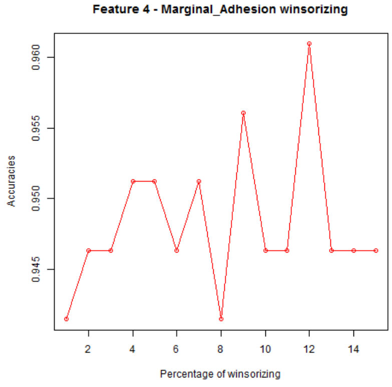


(2)

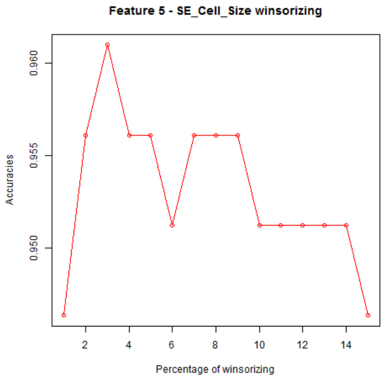


(3)

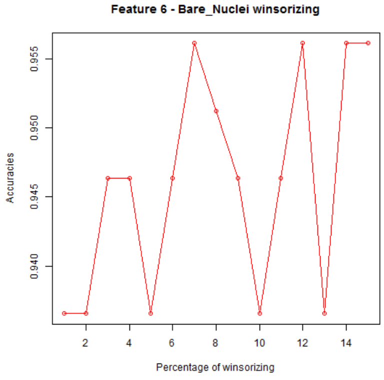
Features 1,3,5,8 and 9 show raise in accuracies during the early phase of the experiment i.e. even before 6 percent winsorization is into effect. Feature 6 periodically peaks up and comes back to original accuracy however, achieves highest accuracy at 15% winsorizing.



(4)

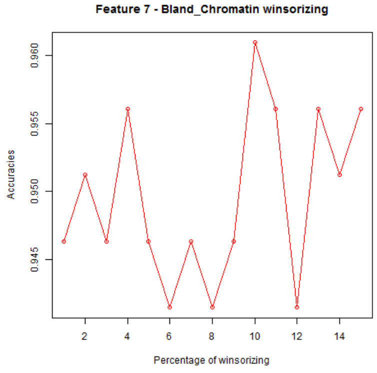


(5)

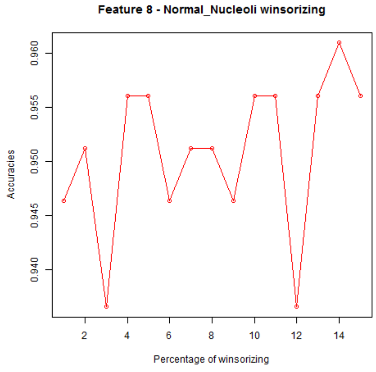


(6)

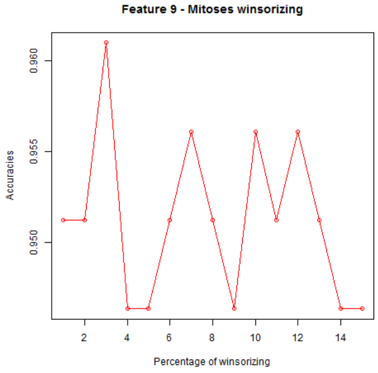
Features 3 and 6 are the only two features which do not cross the high accuracy mark of 0.96 within 1 to 15 percentage of winsorizing.



(7)



(8)

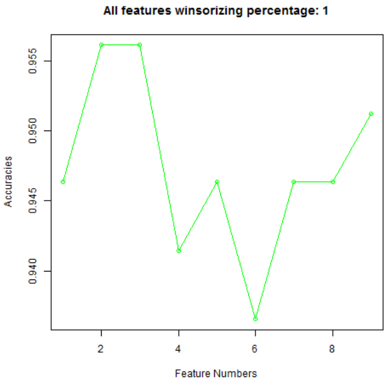


(9)

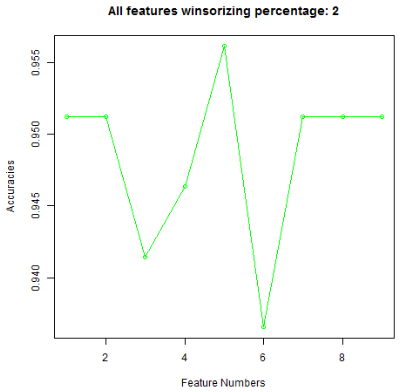
Feature-wise Winsorizing between the range of 1 to 15 percentage showed an increase in accuracies for features 1,4,6,7 and 8. Features 2 and 9 are the only two features with overall decrease in accuracies at the end of 15% percent winsorizing. Features 3 and 5 despite gaining and losing accuracies at various winsorizing percentages are surprisingly found to be neutral.

Similar results and inferences can be seen in the below list of figures where **winsorizing** (again with percentage ranging from 1 to 15) is done for each feature **separately** and accuracies are noted. Each of the below 15 figures for example say the first figure corresponds to change in accuracies as a result of 1% winsorization to feature 1, then 1% to feature 2, and so on till 1% to feature 8.

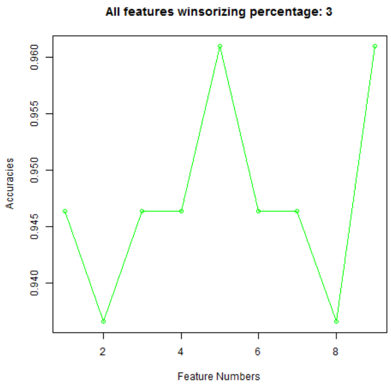
Between 1 to 3 % the accuracies vary in the range of 0.935 to 0.955 and only once rising above 0.96 for feature 5 and 9 with 3% winsorizing. No change in accuracy of feature 6 during 1 and 2 % winsorizing and a little rise is clear effect of 3% winsorizing having significance.



(1)

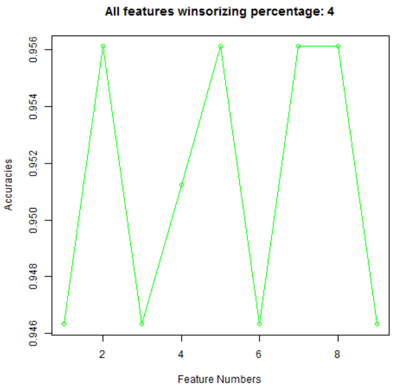


(2)

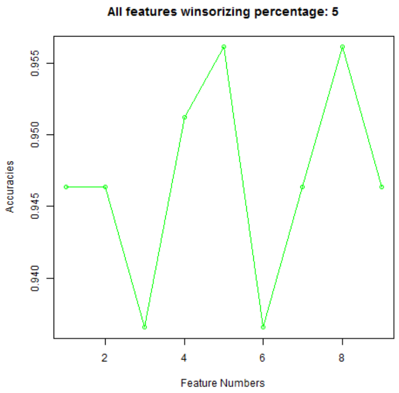


(3)

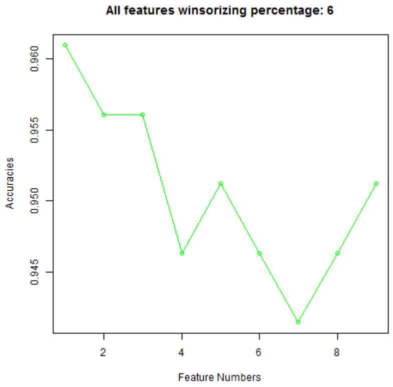
At 4% winsorizing of individual features there is rise in accuracies for features 2,4,5,7,8 from their previous values at 3% winsorizing attributing to "lesser the winsorizing percentage" before.



(4)

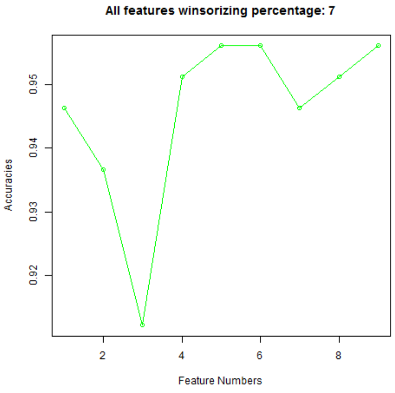


(5)

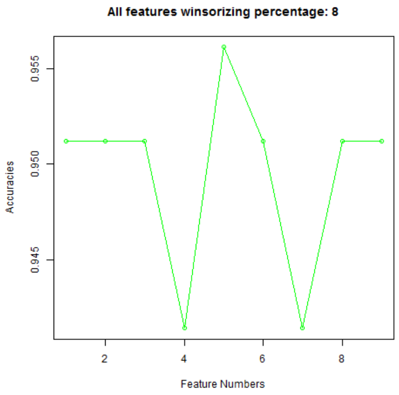


(6)

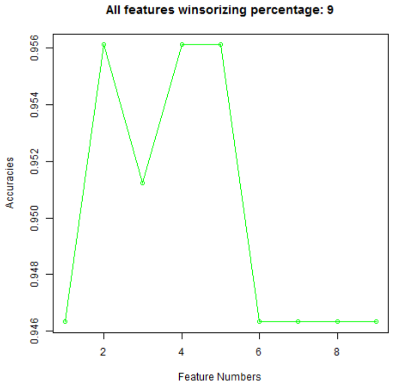
Yet again accuracies of most of the features decrease after winsorizing 5-7% which could be due to replacement of more and more values in the previously replaced ranges nullifying previous updates in weights.



(7)

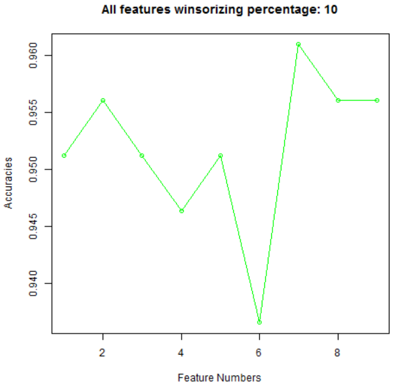


(8)

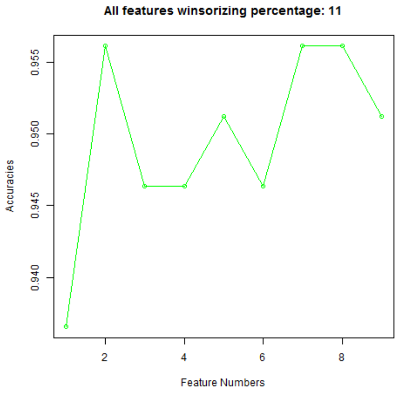


(9)

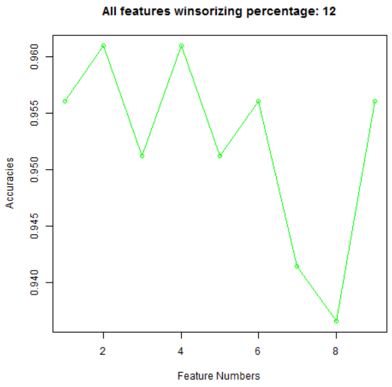
However, at 8-10 % winsorizing the accuracies again gain the lost accuracies and finally at 12% winsorizing couple of features even cross the 0.96 mark.



(10)

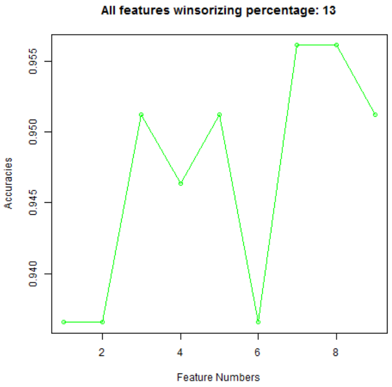


(11)

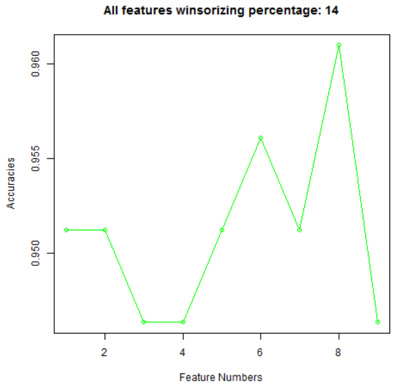


(12)

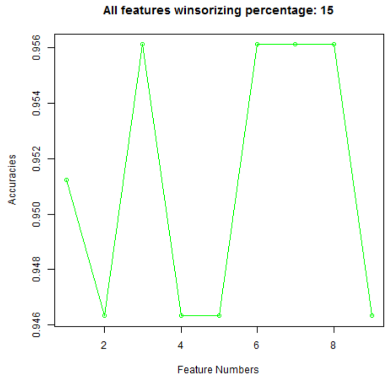
From a combined analysis of these 15 figures we can infer that feature 5 does really well by achieving higher accuracies at 2,3,4,5,7,8,9,10 percentages of winsorizing. Second in the race being feature 8 (at 4,5,11,13,14,15 percentages), third being feature 2 (at 1,4,9,11,12 percentages).



(13)



(14)



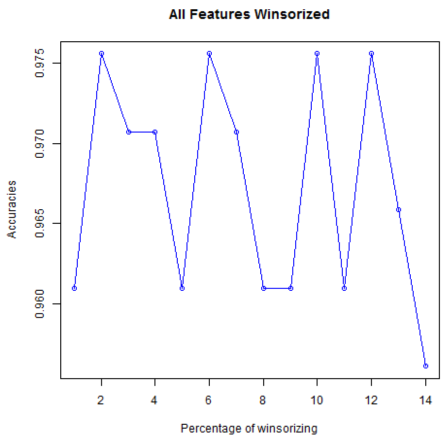
(15)

The effect of winsorizing on a dataset where attributes have

1) Only integer values: since resulting winsorizing values are also needed to be integers in order to make sense of the data and if there was very small variation (e.g. from 12th percentage to 13th percentage) of a given value for example if jth value of ith attribute changes say from 3 to 3.4 and returned again as 3 then for all such attribute the effect of winsorizing gets nullified.

2) Smaller range of values: Since all attributes have a common range between 1 to 10 and for features which have less instances with their values being 1, it would need more than 15% winsorizing for the significant number of such attributes to change their values to 2. For example, if feature 2 has only 20 instances of value 1 then even for 15% winsorizing only 3 of the instance will be affected. Hence significant impact of winsorizing on this dataset is less even with high percentage of winsorizing.

Combined Winsorizing: Below graph shows the change in accuracy when all the features are winsorized simultaneously from 1 to 15 percent. The periodic increase in the accuracy again infers that the effect of winsorizing is periodic as it takes a while for many the values to be replaced with their upper or lower bounds. In other words, for some specific percentages there is not much of impact though the weights are slightly updated which will pick up in the next higher winsorizing.



CONCLUSION

Winsorizing is and effective method to estimate accuracies and costs. However, deciding on the dataset with appropriate number of attributes and range of values these attributes have plays a vital role in application of winsorizing. If the range of the attribute values are large enough and also the type of attribute values are double instead of integers then more specific change in accuracies can be observed.

REFERENCES

My Code: https://github.com/sheshappanavar/BreastCancer_Wisconsin_Winsorizing.git

Code that helped me develop: https://github.com/UserIdentificationBTAS/btas-2016

Documentations to understand code: https://cran.r-project.org/web/packages/nnet/nnet.pdf