[[1]](#footnote-1)

DRESS ATTRIBUT SALE DATA CLASSIFICATION

Himanshu Pandey, Bhargav Pawar, Nishant Bhat, Shyam Kawale, Abhijeet Gawai

Department of Information Technology

***Abstract***— How many times does this happen you see what you were thinking to buy on online shopping sites. These are predictions not magic. Based on your previous data and of other users too, they predict what are you going to buy. These recommendations are necessary for their business and for us too. It gives better option and selection procedure. These data sets are very big in size. Dress\_Attribute\_Sales Data Set is also a kind of dataset whose target is whether a dress is to be recommended or not based on various factors like dresstype, sleeveline, waistline, price, rating, etc.

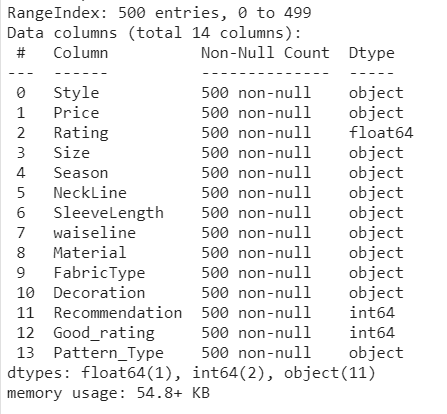
***INTRODUCTION***

While shopping online what we buy is what we see. So it is very important for online shopping companies to improve their recommendation. They require a lot of previous data to predict what is more recommended by the users. Dress\_Attribute\_Sales Data Set provides a lot of information about whether a dress should be recommended or not. Dress attribute dataset gives us an idea about the recommendation for a particular dress on the basis of various factors such as dresstype, sleeveline, waistline, price, rating. etc. So this projects aims to find correlation between these attributes of the data set. And classify the datapoints into two classes recommended or not.

# ***Problem Statement***

Take Dress\_Attribute\_Sales Data Set present in UCI repository and find the correlation of the input among them self and with the target. Find the different conclusions you got after data analysis of features. Implement any classification algorithm to do classification of records.

# ***Dataset Analysis***



|  |  |  |
| --- | --- | --- |
|  |  |  |

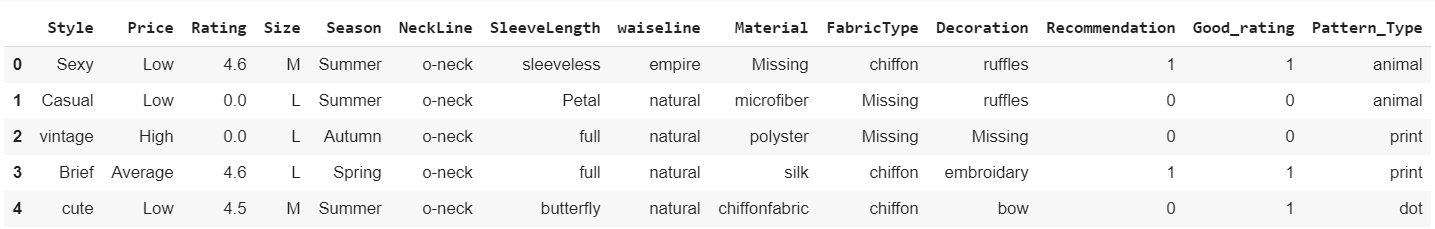
## **k nearest neighbors algorithm**

In knn classification all the attribute is plotted in multi-dimensional graph. Then when a new data point comes it checks it’s k number of nearest neighbors, the class which has the highest count it assigns that datapoint to that particular class. Since there is no training involved so this classification is very fast. Total time from training to prediction time required is very less.

# ***Methodology***

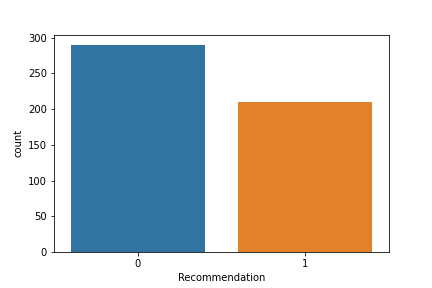
***A.Data Set Used***

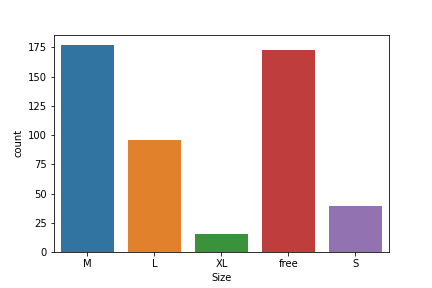
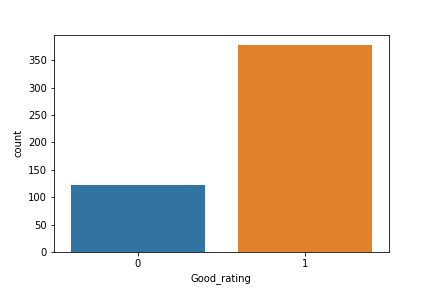
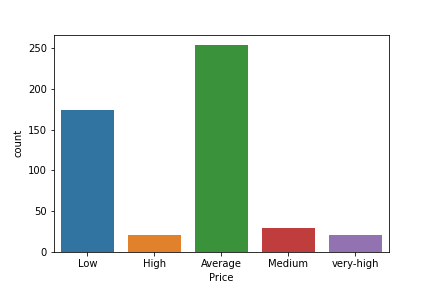
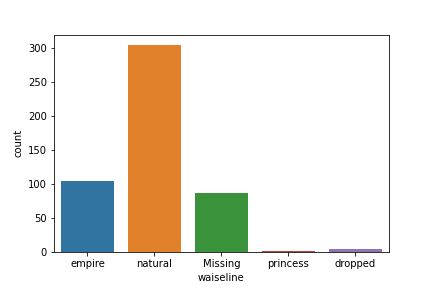
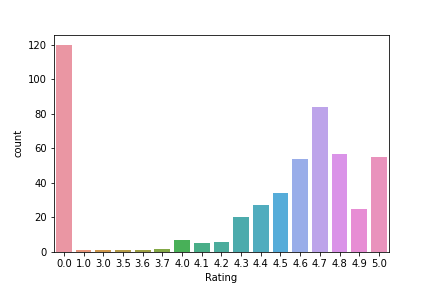
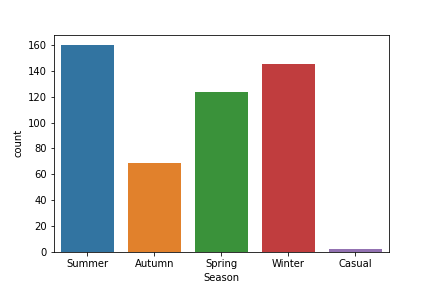
Dress\_Attribute\_Sales data set from UCI Machine learning repository [16] is utilized for the employment of the designed technique.



***B. Data Pre-Processing***

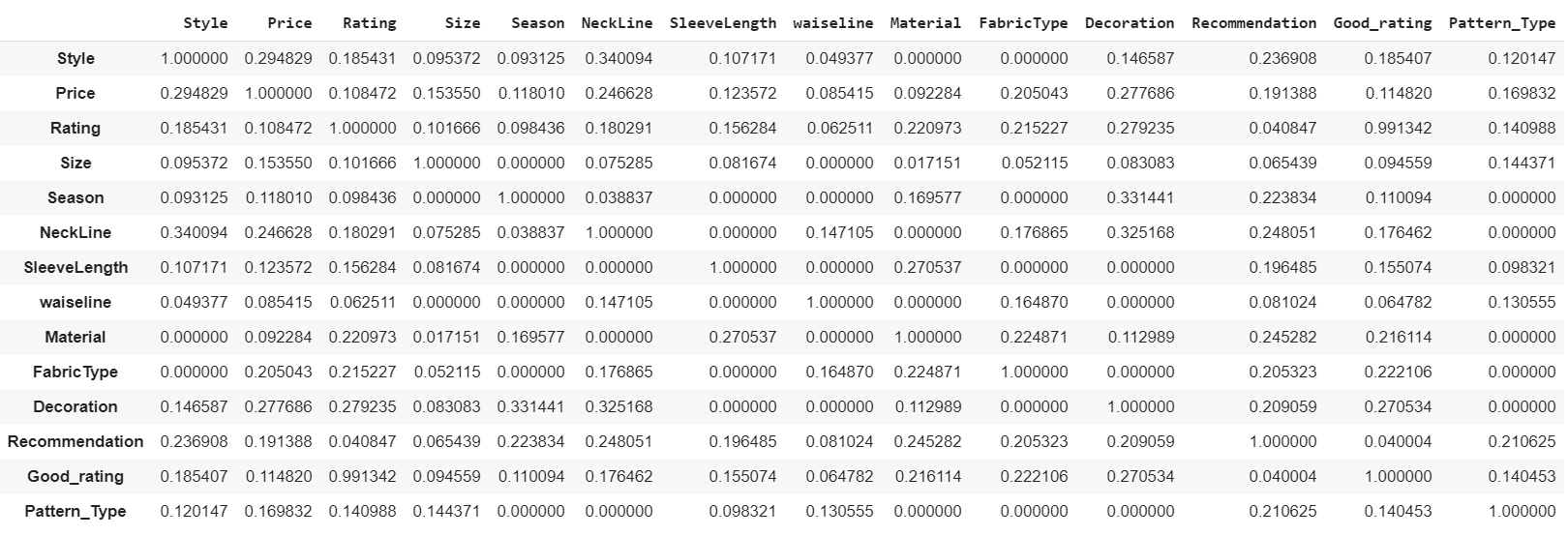
The initial step of the proposed techniques is data preprocessing by which raw data can be converted into the suitable format for which the providing the data set for training and testing and for further analyzing.

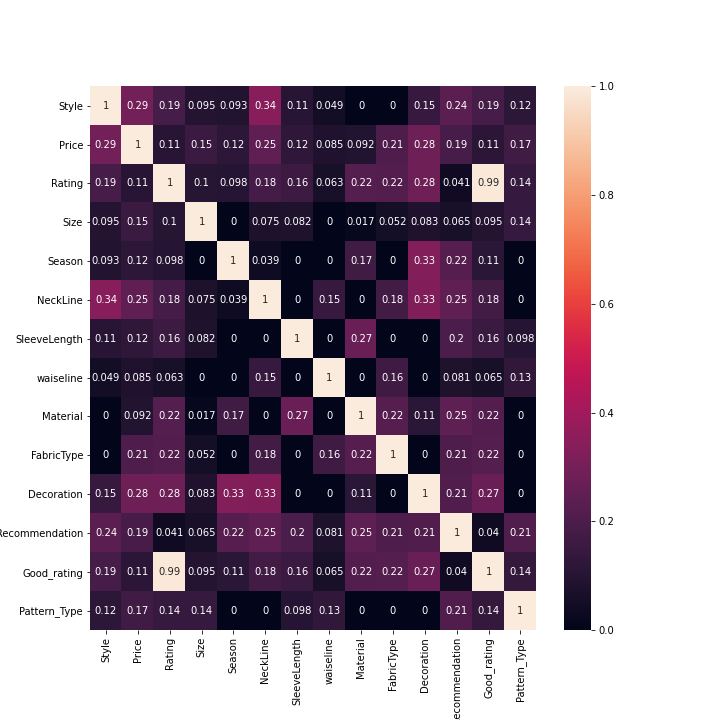




Correlation Between Input Features:

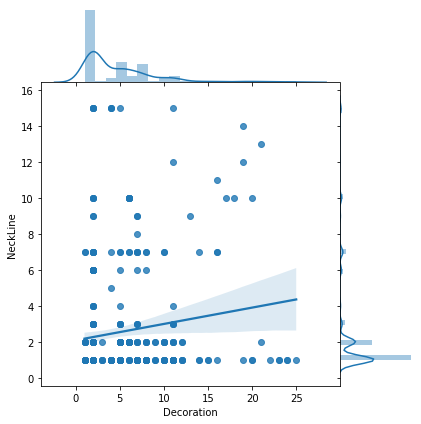
1. Since our data set have a lot of categorical features so for finding the correlation between categorical features there are various ways, for example, chi square test, cramers\_v, etc. Here we have used cramers\_V.
2. For correlation between categorical and numerical we have used theils\_u.
3. For correlation between numerical and numerical we have used pearson.



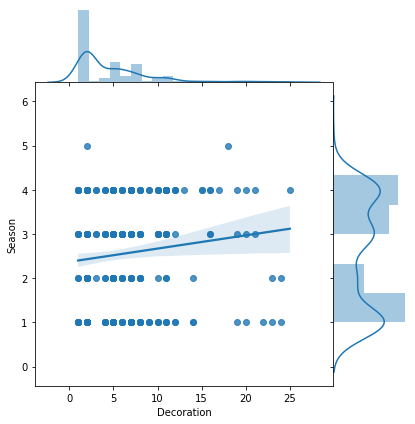
******

From the correlation heatmap we can get the most dependent pairs of attributes and plot them to view their correlation.

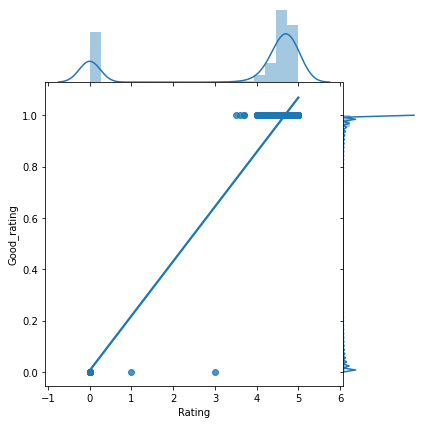
Here we have chosen ‘Decoration’ and ‘NeckLine’



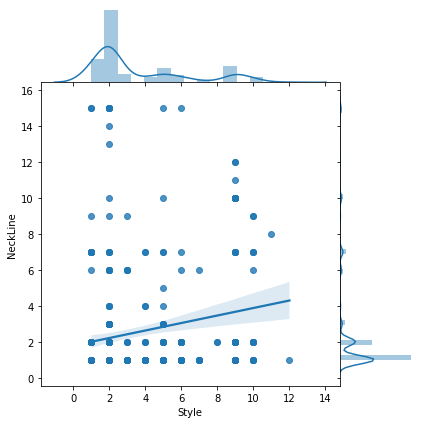
Here we have chosen ‘Decoration’ and ‘Season’



Here we have chosen ‘Rating’ and ‘Good\_rating’

****

Here we have chosen ‘Style’ and ‘NeckLine’

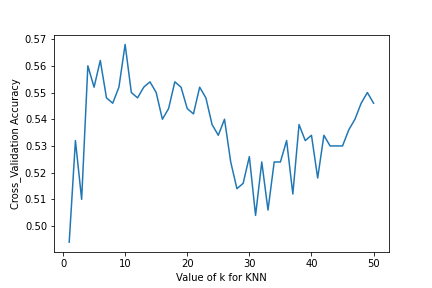


***C.Training and Testing Samples***

Out of the entire dress attribute data set obtained for investigations 80 % of data set is utilized in making the Classifiers to learn and the remaining 20% of dataset will be employed to verify the functionality of the classification process of the designed technique.

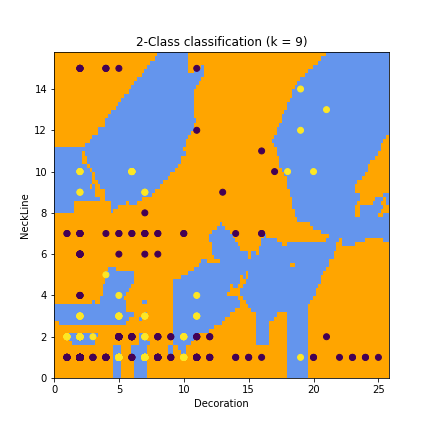
# ***EXPERIMENTAL ANALYSIS***

# we have implement knn algorithm for k =1 to k=50 and for 5 folds each for better classification and for improvement in accuracy. and here are the results for each classification:

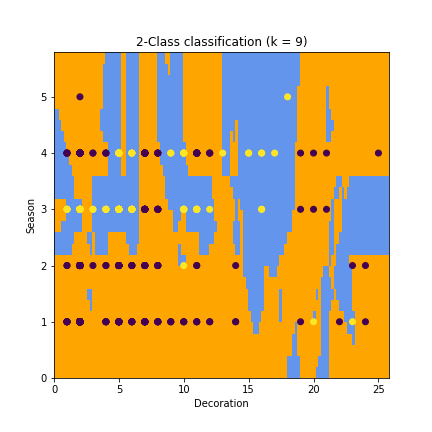


Max accuracy is 0.568 and is at k=9.

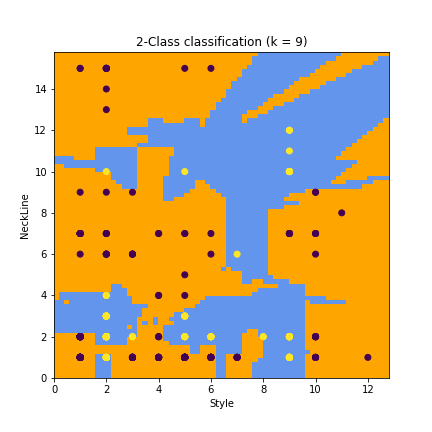
Then to see the classification process through graph we created a graph having classification of every point on it, and then plotting our data points. Attribute which we selected are on the basis of correlation, which has more dependencies.

Here we have chosen ‘Decoration’ and ‘NeckLine’.

##### Here we have chosen ‘Decoration’ and ‘Season’.



Here we have chosen ‘Style’ and ‘NeckLine’.



***Conclusion:***

Here we performed KNN classifier and we got the accuracy of 0.6.And we found the correlation between various attributes of the data set and found that some of the attributes were slightly dependent but otherwise it was totally independent.

***References:***

<https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>

<https://stackoverflow.com/questions/36417382/how-do-i-delete-or-close-a-graph-in-matplotlib>

<https://github.com/shakedzy/dython/blob/master/dython/nominal.py>

<https://seaborn.pydata.org/tutorial/distributions.html>

<https://scikit-learn.org/stable/auto_examples/neighbors/plot_classification.html>

<https://towardsdatascience.com/how-to-export-pandas-dataframe-to-csv-2038e43d9c03>

<https://towardsdatascience.com/the-art-of-effective-visualization-of-multi-dimensional-data-6c7202990c57>

<https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.scatter.html>

1. [↑](#footnote-ref-1)