**1. Unstructured Data Classification**

Implement classification of ‘labeled\_articles.zip’ data set.

1. Extract and load files, using below code snippet. Each line contains the label and the statement.

*Training\_Folder = '/labeled\_articles/'*

*Training\_Label = []*

*Training\_Matrix = []*

*for file in os.listdir(Training\_Folder):*

*print (file)*

*Text\_File = open(Training\_Folder+file,'r+',encoding='utf-8')*

*for line in Text\_File:*

*if (line.startswith('###')):*

*continue*

*#### Sentence preprocessing for cleaning*

*#line = line.replace('\n','');line = line.lower()*

*line = re.sub(r'[-\*+%$()\.,/?!><"&#\[\]\(\)\\]', ' ',line)*

*lines = line.split(' ')*

*#### Putting lines into the list for classification*

*Training\_Matrix.append(lines[1])*

*#### Making file name as label for classification*

*Training\_Label.append(lines[0])*

*Text\_File.close()*

2. Classify the corpus into given labels.

3. Incorporate the below:

a. CountVectoriser - http://scikit-learn.org/stable/modules/generated/sklearn.feature\_extraction.text.CountVectorizer.html

b. kfold crossvalidation: http://scikit-learn.org/stable/modules/generated/sklearn.model\_selection.KFold.html

c. Classifiers:

* DecisionTreeClassifier
* SGDClassifier
* SVC
* KNeighborsClassifier
* OneVsRestClassifier(svm.LinearSVC())
* RandomForestClassifier
* MultinomialNB - http://scikit-learn.org/stable/modules/generated/sklearn.naive\_bayes.MultinomialNB.html

d. classification\_report

e. Save the classifier model with highest accuracy – Predict using saved model - <http://scikit-learn.org/stable/tutorial/basic/tutorial.html>

**2. Regression**

1. Implement regression analysis for ‘train.csv’

2. Analyze the data and apply required pre-processing steps, select only the important variables for the model

3. Split data into train, test. Evaluate using test data, compute RMSE and plot the same

5. Find the best performing algorithm

**3. Clustering**

1. Implement clustering algorithm to identify the various customer segments hidden in the data. Find optimal number of clusters.

[**https://archive.ics.uci.edu/ml/datasets/Wholesale+customers**](https://archive.ics.uci.edu/ml/datasets/Wholesale+customers)

2. Evaluate using ‘Elbow’ method (<https://en.wikipedia.org/wiki/Elbow_method_(clustering>))

**4.** Anomaly detection

1. Implement an anomaly detection model to identify bad connections vs good connections.

<http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> – [kddcup.data\_10\_percent.gz](http://kdd.ics.uci.edu/databases/kddcup99/kddcup.data_10_percent.gz)

2. All ['back.', 'buffer\_overflow.', 'ftp\_write.', 'guess\_passwd.', 'imap.', 'ipsweep.', 'land.', 'loadmodule.', 'multihop.', 'neptune.', 'nmap.', 'perl.', 'phf.', 'pod.', 'portsweep.', 'rootkit.', 'satan.', 'smurf.', 'spy.', 'teardrop.', 'warezclient.', 'warezmaster.'] can be considered as ‘attack’.

3. You may stratify split the data and work on a smaller set, if any memory issues.

4. Evaluate the detection based on test data.