Batch - 2

Rohith M N - 2018AB04514
Amritanshu Kumar - 2018AB04550
Basireddy Venkatasivateja Reddy - 2018AB04693
Anirban Ghorai - 2018AB04523
Om Prakash Yadav - 2018AB04566

DATA MINING Assignment-1
Chronic Kidney Disease

BITS Pilani
Pilani Campus

Agenda

- Problem Statement
- Understanding the data
- Pre-Processing techniques used
- Algorithm selection of building model
- Discussion on Results and Observations
- Conclusion

BITS Pilani
Pilani Campus

Problem Statement

Problem Statement

- **Chronic Kidney Disease** : Longstanding disease of the kidneys leading to renal failure. Often has no symptoms and is diagnosed by blood test.
- 30 million people in the United States are living with chronic kidney disease (CKD).
- The kidneys filter waste and excess fluid from the blood. As kidneys fail, waste builds up.
- **Causes:**
  - Diabetes, High BP(hypertension), Heart Disease
  - Having a family member with kidney disease
  - Being over 60 years old

Mine and Analyze CKD dataset

- Data Mining and Analytics plays a vital role to know the occurrence of the CKD at early stage in advance.
- Dataset Source:
  - Dr.P.Soundarapandian.M.D.,D.M
    (Senior Consultant Nephrologist),
    Apollo Hospitals,
    Managiri,
    Madurai Main Road,
    Karaikudi,
    Tamilnadu, India.

Language Used for Analysis: Python

innovate achieve lead

BITS Pilani

Pilani Campus

Understanding of Data

Understanding of Data

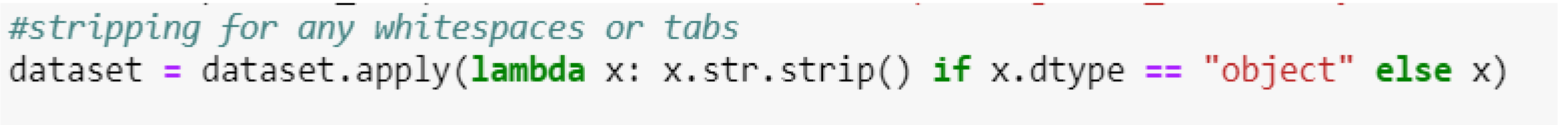
Attributes	Representation	Attribute Info	Description
age	age	numerical	years
blood pressure	bp	numerical	mm/Hg
specific gravity	sg	nominal	(1.005,1.010,1.015,1.020,1.025)
albumin	al	nominal	(0,1,2,3,4,5)
sugar	su	nominal	(0,1,2,3,4,5)
red blood cells	rbc	nominal	normal,abnormal
pus cell	pc	nominal	normal,abnormal
pus cell clumps	pcc	nominal	present,notpresent
bacteria	ba	nominal	present,notpresent
blood glucose random	bgr	numerical	mgs/dl
blood urea	bu	numerical	mgs/dl
serum creatinine	sc	numerical	mgs/dl
sodium	sod	numerical	mEq/L
potassium	pot	numerical	mEq/L
hemoglobin	hemo	numerical	gms
packed cell volume	pcv	numerical	numerical
white blood cell count	wc	numerical	cells/cumm
red blood cell count	rc	numerical	millions/cmm
hypertension	htn	nominal	yes,no
diabetes mellitus	dm	nominal	yes,no
coronary artery disease	cad	nominal	yes,no
appetite	appet	nominal	good,poor
pedal edema	pe	nominal	yes,no
anemia	ane	nominal	yes,no
class	class	nominal	ckd,notckd

Pre-Processing Techniques

Pre-Processing Techniques

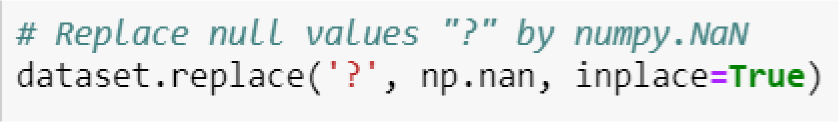
STEPS:

- Read dataset file
- Stripping for any whitespaces or tabs in csv cells



```
#stripping for any whitespaces or tabs
dataset = dataset.apply(lambda x: x.str.strip() if x.dtype == "object" else x)
```

- Replace null values “?” to numpy.NaN



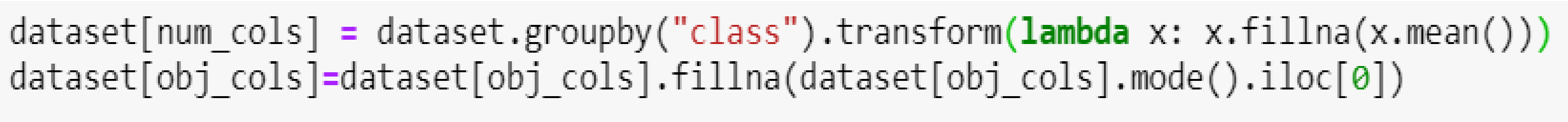
```
# Replace null values "?" by numpy.NaN
dataset.replace('?', np.nan, inplace=True)
```

Pre-Processing Techniques(contd..,)

- Imputing the missing values

  Num_cols : Fill the NULL values with a **groupby class mean**

  Obj_cols: Fill the NULL values with a **mode**

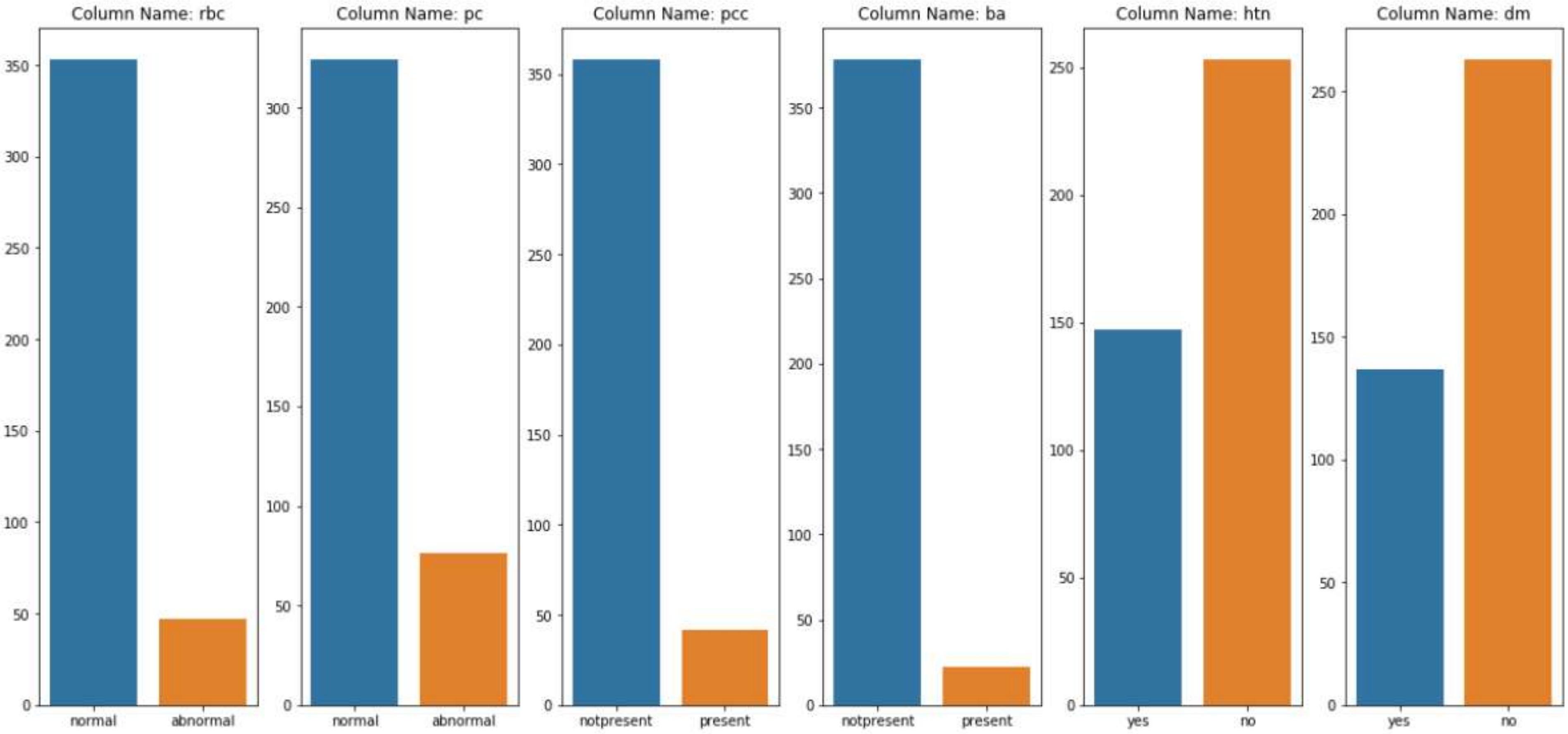


```
dataset[num_cols] = dataset.groupby("class").transform(lambda x: x.fillna(x.mean()))
dataset[obj_cols]=dataset[obj_cols].fillna(dataset[obj_cols].mode().iloc[0])
```

- Normalizing Features

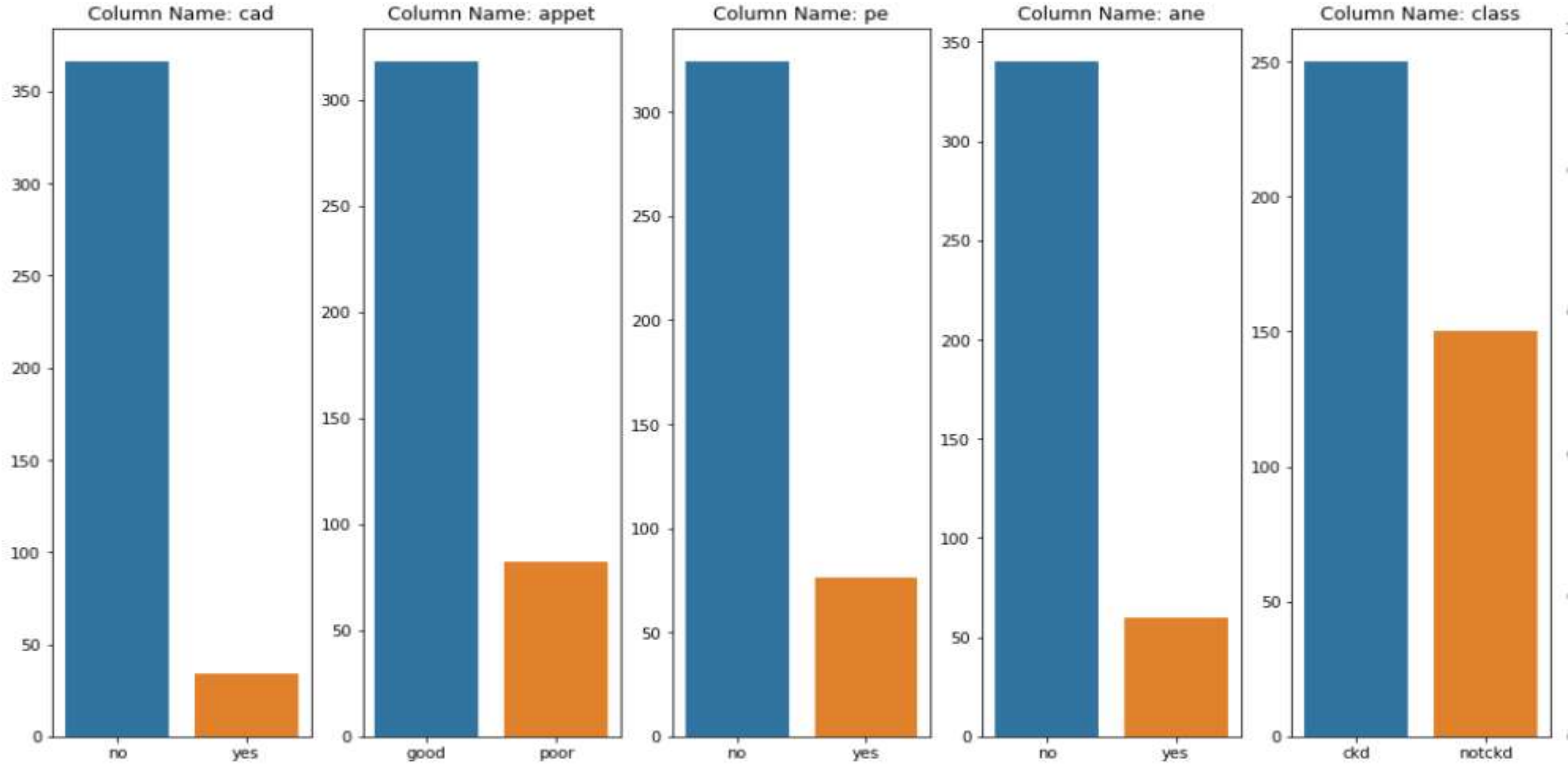
  Used MinMaxScaler preprocessing technique for normalizing features.

Exploratory Data Analysis(EDA)



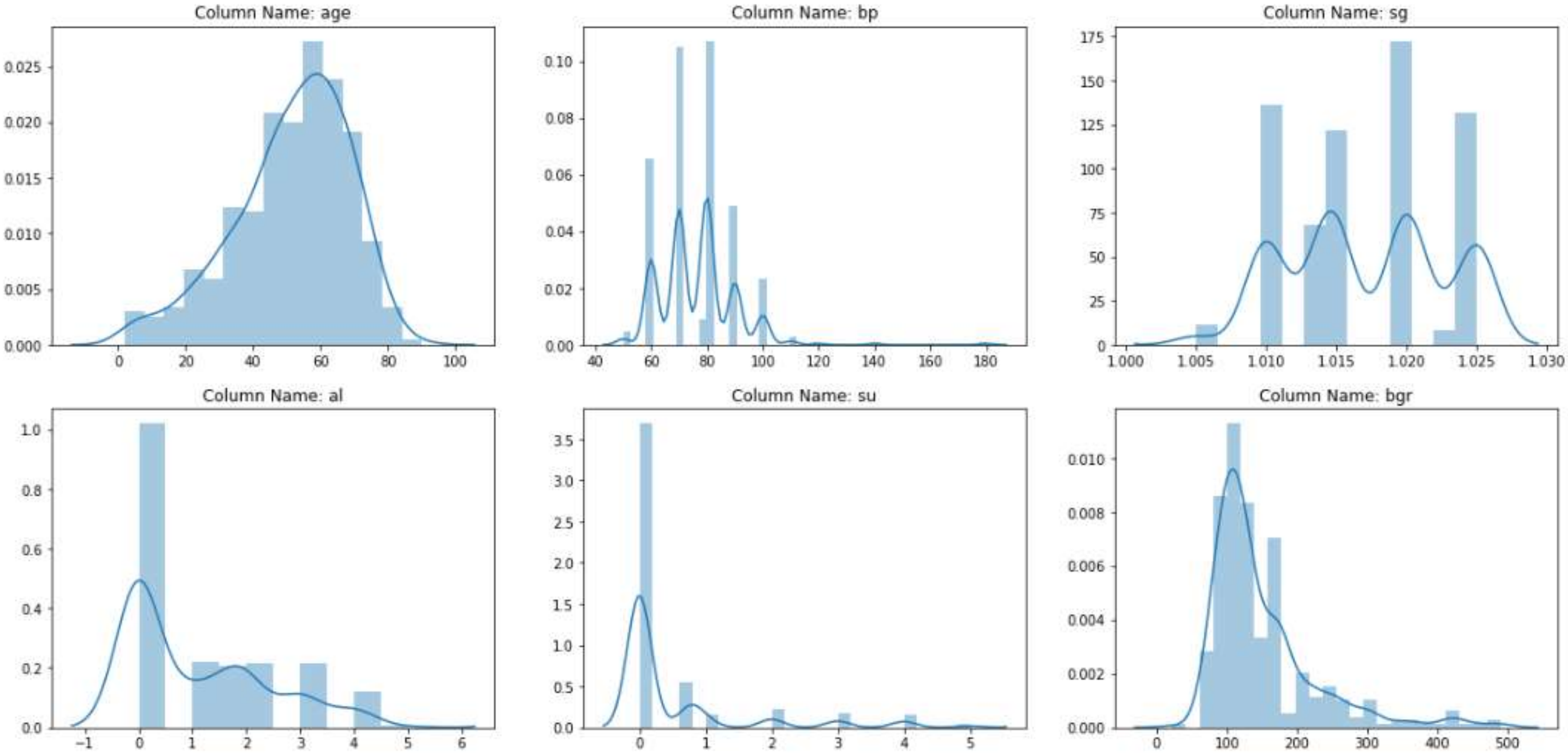
For Categorical Variables

Exploratory Data Analysis(Contd..,)



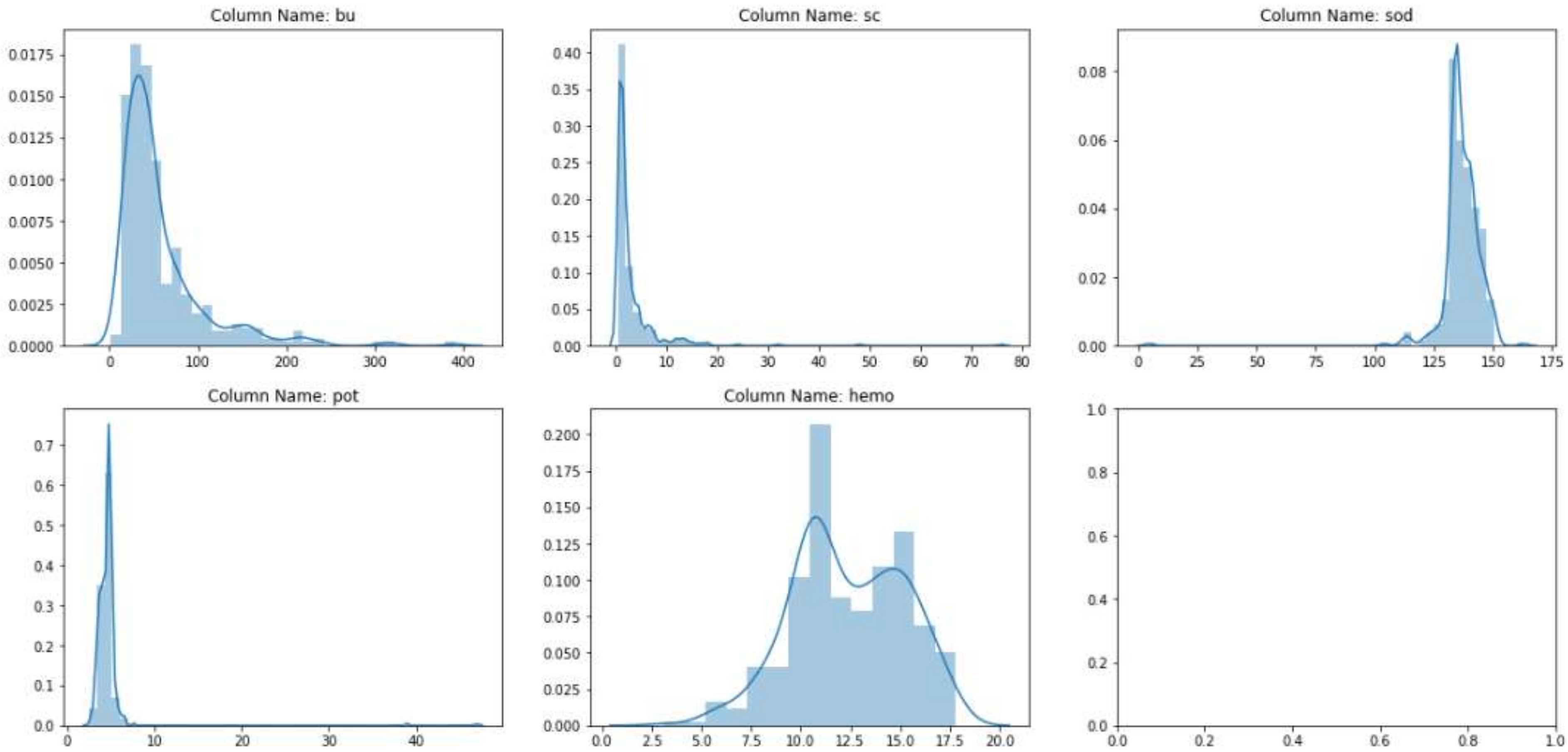
For Categorical Variables

Exploratory Data Analysis(Contd..,)



For Numerical Variables

Exploratory Data Analysis(Contd..,)



For Numerical Variables

innovate achieve lead

BITS Pilani
Pilani Campus

Dimensionality Reduction

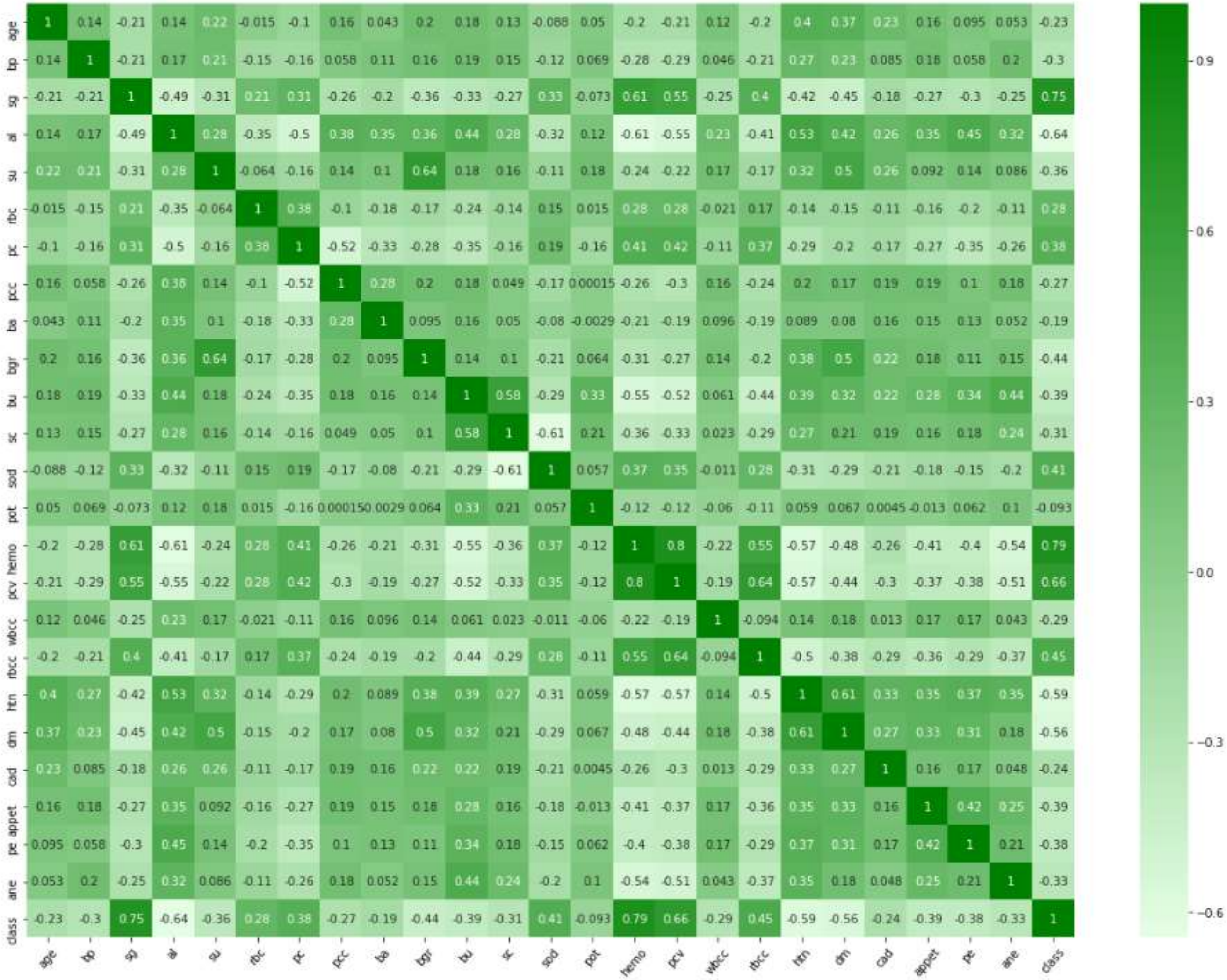
Dimensionality Reduction

Tried to find out whether we can reduce the data by analyzing it with correlation matrix.

correlation analysis shows us how to determine both the nature and strength of relationship between two variables.

correlation lies between -1 to 1 (0: No correlation; -1: perfect negative correlation; +1: perfect positive correlation)

Correlation Matrix



Correlation Matrix(Contd..,)

From the previous slide correlation matrix we could infer that all none of the attributes are very strongly correlated. (<0.85 , if we consider 85%)

The Max correlation we saw was 0.80, between the feature/attribute ‘hemo’ and ’pcv’.

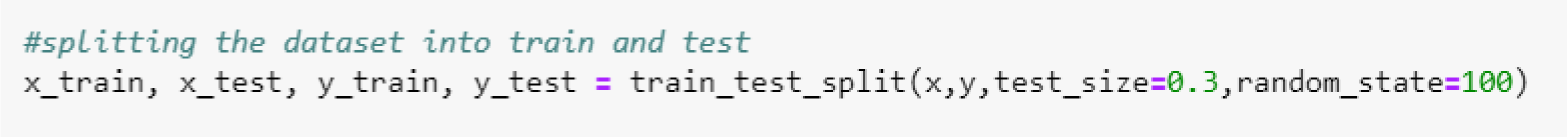
Hence we are not considering the option of Feature Selection.

Algorithm selection

Algorithm selection of building model

In order to build and test model the dataset has been spited into Training and Test data,.

- Training data – 70% (280 Records)
- Test data – 30% (120 Records)



```
#splitting the dataset into train and test
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.3,random_state=100)
```

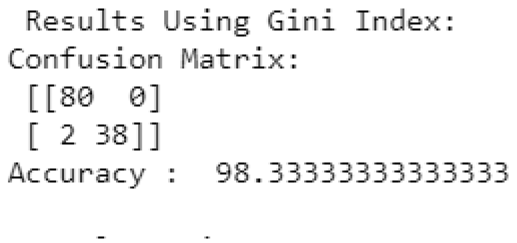
Different Models Used:

1. Decision Tree Classifier using gini
2. Decision Tree Classifier using entropy
3. Naïve Bayes Classifier
4. SVM
5. Random Forest

Different Models Used(Contd..,):

1. Decision Tree Classifier using gini

- The model created and Tested using “Decision Tree Classifier” using gini index.
- The accuracy we got for this is 98.33%
- Confusion Matrix:

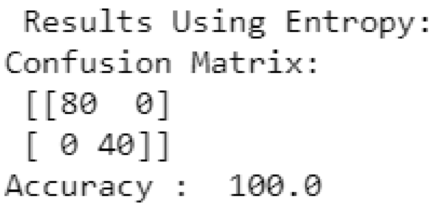


```
 Results Using Gini Index:
Confusion Matrix:
 [[80  0]
 [ 2 38]]
Accuracy :  98.33333333333333
```

Different Models Used(Contd..,):

2. Decision Tree Classifier using entropy

- The model created and Tested using “Decision Tree Classifier” using entropy.
- The accuracy we got for this is 100%
- Confusion Matrix:

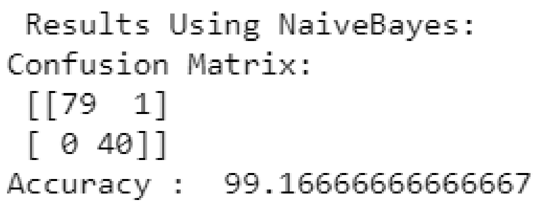


```
 Results Using Entropy:
Confusion Matrix:
 [[80  0]
 [ 0 40]]
Accuracy :  100.0
```

Different Models Used(Contd..,):

3. Naïve Bayes Classifier

- The model created and Tested using “Naïve Bayes Classifier
- The accuracy we got for this is 99.16%
- Confusion Matrix:

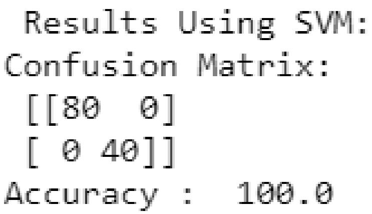


```
 Results Using NaiveBayes:
Confusion Matrix:
 [[79  1]
 [ 0 40]]
Accuracy :  99.16666666666667
```

Different Models Used(Contd..,):

4. Support Vector Machine

- The model created and Tested using “Support Vector Machine” Classifier.
- The accuracy we got for this is 100%
- Confusion Matrix:

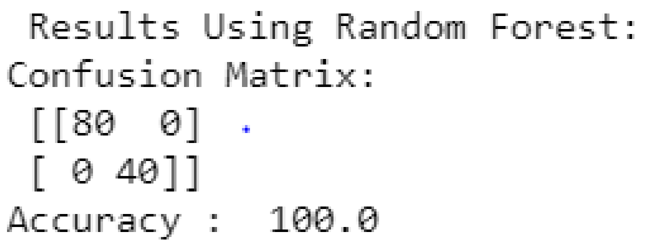


```
 Results Using SVM:
Confusion Matrix:
 [[80  0]
 [ 0 40]]
Accuracy :  100.0
```

Different Models Used(Contd..,):

5. Random Forest

- The model created and Tested using “Random Forest”
- The accuracy we got for this is 100%
- Confusion Matrix:

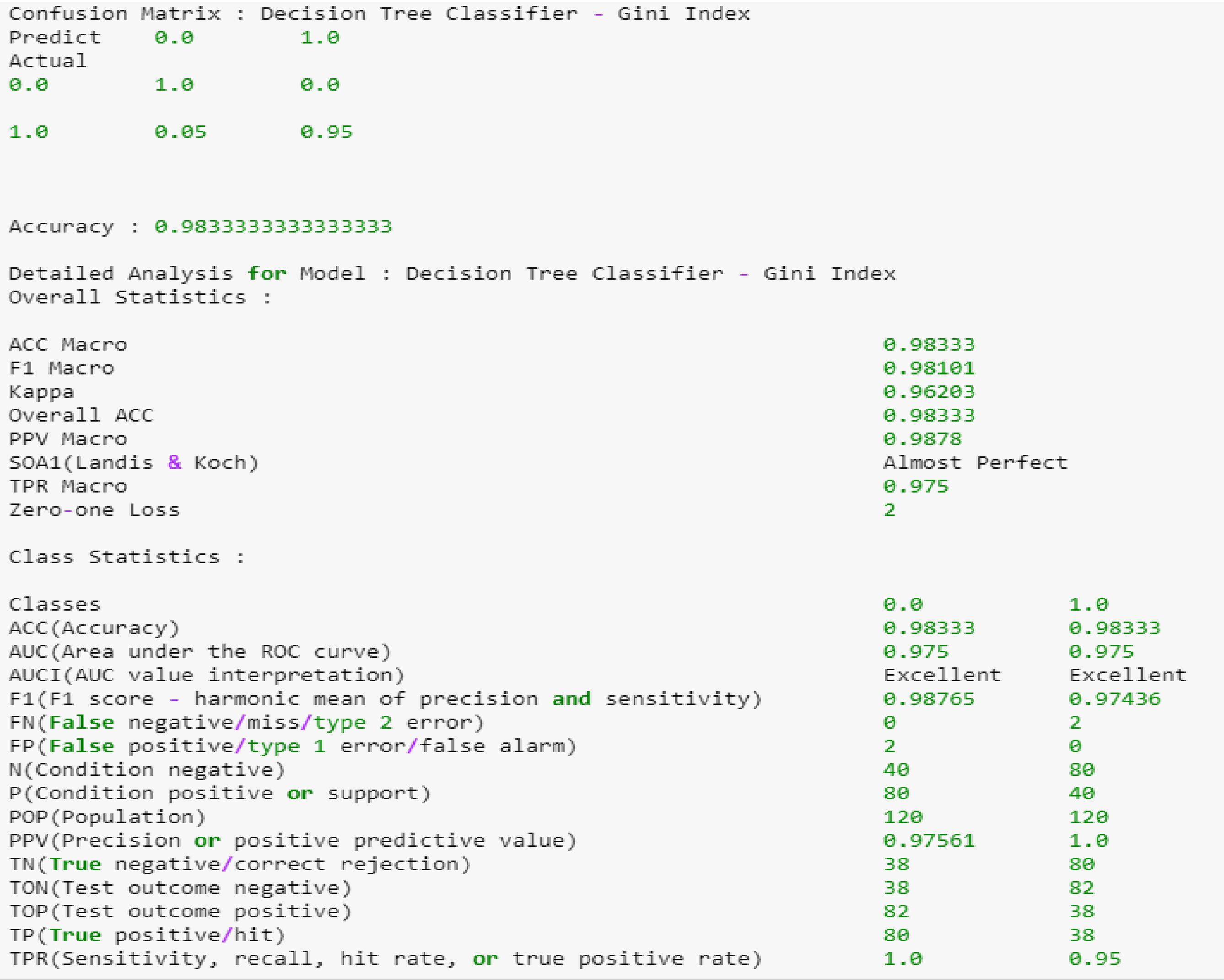


```
 Results Using Random Forest:
Confusion Matrix:
 [[80  0]
 [ 0 40]]
Accuracy :  100.0
```

BITS Pilani
Pilani Campus

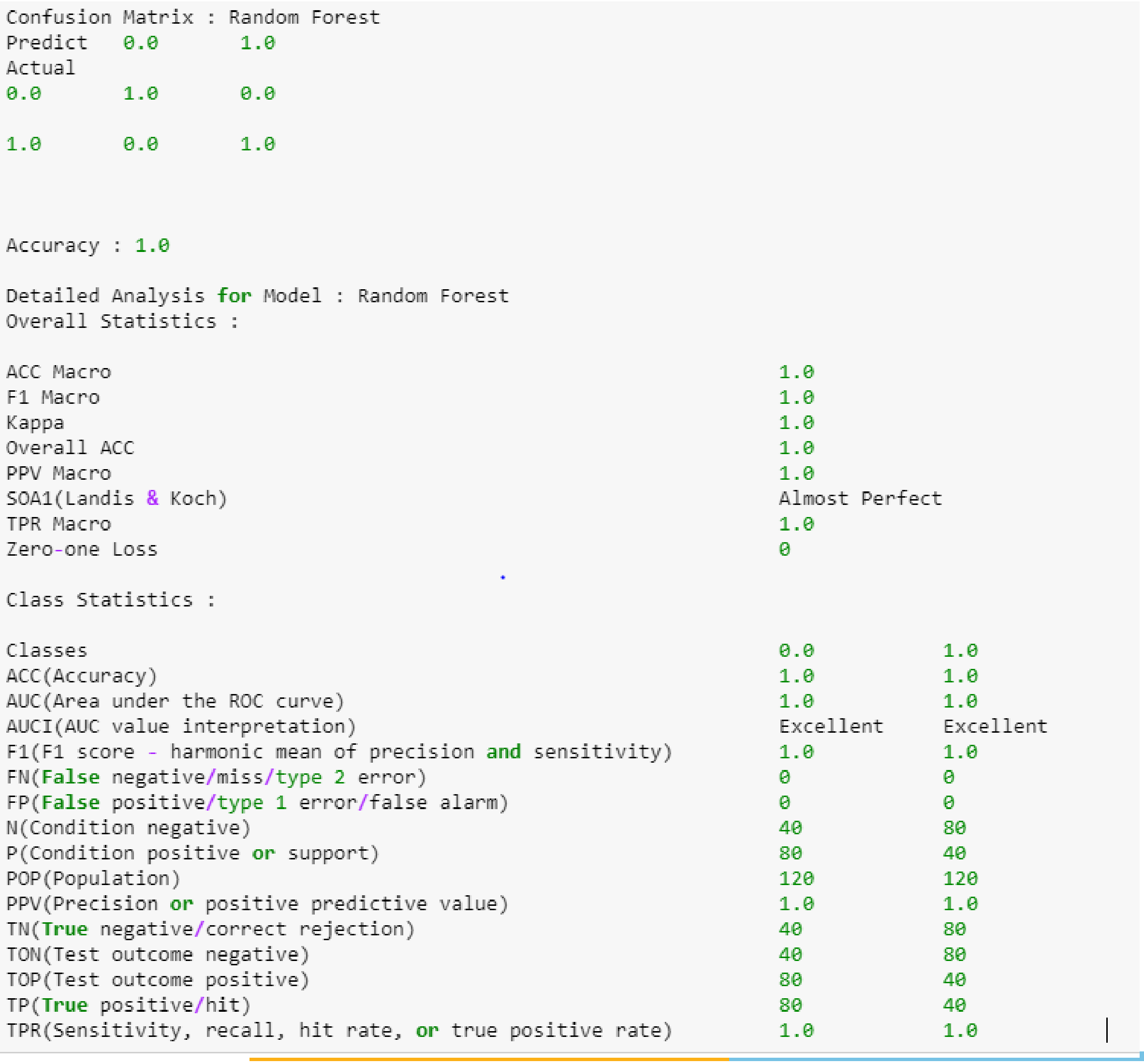
Discussion on the Results

Decision Tree gini - Detailed Report



```
Confusion Matrix : Decision Tree Classifier - Gini Index
Predict    0.0        1.0
Actual
0.0        1.0        0.0

1.0        0.05       0.95



Accuracy : 0.9833333333333333

Detailed Analysis for Model : Decision Tree Classifier - Gini Index
Overall Statistics :

ACC Macro                                                         0.98333
F1 Macro                                                          0.98101
Kappa                                                             0.96203
Overall ACC                                                       0.98333
PPV Macro                                                         0.9878
SOA1(Landis & Koch)                                               Almost Perfect
TPR Macro                                                         0.975
Zero-one Loss                                                     2

Class Statistics :

Classes                                                           0.0           1.0
ACC(Accuracy)                                                     0.98333       0.98333
AUC(Area under the ROC curve)                                     0.975         0.975
AUCI(AUC value interpretation)                                    Excellent     Excellent
F1(F1 score - harmonic mean of precision and sensitivity)         0.98765       0.97436
FN(False negative/miss/type 2 error)                              0             2
FP(False positive/type 1 error/false alarm)                       2             0
N(Condition negative)                                             40            80
P(Condition positive or support)                                  80            40
POP(Population)                                                   120           120
PPV(Precision or positive predictive value)                       0.97561       1.0
TN(True negative/correct rejection)                               38            80
TON(Test outcome negative)                                        38            82
TOP(Test outcome positive)                                        82            38
TP(True positive/hit)                                             80            38
TPR(Sensitivity, recall, hit rate, or true positive rate)         1.0           0.95
```

Random Forest Detailed Report



```
Confusion Matrix : Random Forest
Predict   0.0       1.0
Actual
0.0       1.0       0.0

1.0       0.0       1.0



Accuracy : 1.0

Detailed Analysis for Model : Random Forest
Overall Statistics :

ACC Macro                                                         1.0
F1 Macro                                                          1.0
Kappa                                                             1.0
Overall ACC                                                       1.0
PPV Macro                                                         1.0
SOA1(Landis & Koch)                                               Almost Perfect
TPR Macro                                                         1.0
Zero-one Loss                                                     0

Class Statistics :

Classes                                                           0.0           1.0
ACC(Accuracy)                                                     1.0           1.0
AUC(Area under the ROC curve)                                     1.0           1.0
AUCI(AUC value interpretation)                                    Excellent     Excellent
F1(F1 score - harmonic mean of precision and sensitivity)         1.0           1.0
FN(False negative/miss/type 2 error)                              0             0
FP(False positive/type 1 error/false alarm)                       0             0
N(Condition negative)                                             40            80
P(Condition positive or support)                                  80            40
POP(Population)                                                   120           120
PPV(Precision or positive predictive value)                       1.0           1.0
TN(True negative/correct rejection)                               40            80
TON(Test outcome negative)                                        40            80
TOP(Test outcome positive)                                        80            40
TP(True positive/hit)                                             80            40
TPR(Sensitivity, recall, hit rate, or true positive rate)         1.0           1.0
```

Discussion on the Results(Contd..,)

By Observing the confusion matrix and Accuracy of all the models we could infer that for the given data set we could achieve 100% accuracy by applying below models.

- Decision Tree Classifier using entropy
- SVM
- Random Forest

Extracted the detailed report involving Precession, Recall, F1 score, TPR etc..,of Decision Tree(gini) and Random Forest model (please see the previous slide)- Same can be done for all models.

Discussion on the Results(Contd..,)

Advantages of Random Forest:

- As we mentioned earlier a single decision tree tends to overfit the data. The process of averaging or combining the results of different decision trees helps to overcome the problem of overfitting.
- Random forests also have less variance than a single decision tree. It means that it works correctly for a large range of data items than single decision trees.

Discussion on the Results(Contd..,)

Disadvantages of Random Forest:

- The main disadvantage of Random forests is their complexity. They are much harder and time-consuming to construct than decision trees.
- In addition, the prediction process using random forests is time-consuming than other algorithms.

Conclusion

Conclusion

- The CKD(Chronic Kidney Disease) can be very well predicted using many classifiers in Data Mining. We in this assignment have used Decision Tree Classifier with gini and entropy, Naïve Bayes Classifier, SVM and Random Forest.

- As per our observations in the detailed report of all the models, the best models for the given dataset are SVM and Random Forest.

THANK YOU