In [4]:

```
%matplotlib inline
```

In [1]:

```
!pip install data_helper_2
```

```
Requirement already satisfied: data_helper_2 in c:\users\lenovo\anaconda3
\lib\site-packages (0.3)
Requirement already satisfied: matplotlib in c:\users\lenovo\anaconda3\lib
\site-packages (from data_helper_2) (2.2.2)
Requirement already satisfied: sklearn in c:\users\lenovo\anaconda3\lib\si
te-packages (from data_helper_2) (0.0)
Requirement already satisfied: seaborn in c:\users\lenovo\anaconda3\lib\si
te-packages (from data_helper_2) (0.7.1)
Requirement already satisfied: statsmodels in c:\users\lenovo\anaconda3\li
b\site-packages (from data_helper_2) (0.8.0)
Requirement already satisfied: pandas in c:\users\lenovo\anaconda3\lib\sit
e-packages (from data_helper_2) (0.23.4)
Requirement already satisfied: numpy>=1.7.1 in c:\users\lenovo\appdata\roa
ming\python\python36\site-packages (from matplotlib->data_helper_2) (1.15.
0)
Requirement already satisfied: cycler>=0.10 in c:\users\lenovo\anaconda3\l
ib\site-packages (from matplotlib->data_helper_2) (0.10.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in
c:\users\lenovo\anaconda3\lib\site-packages (from matplotlib->data_helper_
2) (2.1.4)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\lenovo\ana
conda3\lib\site-packages (from matplotlib->data_helper_2) (2.7.3)
Requirement already satisfied: pytz in c:\users\lenovo\anaconda3\lib\site-
packages (from matplotlib->data_helper_2) (2017.2)
Requirement already satisfied: six>=1.10 in c:\users\lenovo\appdata\roamin
g\python\python36\site-packages (from matplotlib->data_helper_2) (1.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\lenovo\appdat
a\roaming\python\python36\site-packages (from matplotlib->data_helper_2)
(1.0.1)
Requirement already satisfied: scikit-learn in c:\users\lenovo\anaconda3\l
ib\site-packages (from sklearn->data_helper_2) (0.19.1)
Requirement already satisfied: setuptools in c:\users\lenovo\anaconda3\lib
\site-packages (from kiwisolver>=1.0.1->matplotlib->data_helper_2) (40.2.
0)
```

In [5]:

```
import warnings
warnings.filterwarnings("ignore")
```

In [3]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from data_helper_2.data_helper import completeanalysis
```

The package being used here., (data_helper_2) is developed and maintained by me. I have attached the essential parts of the code and can be installed by using "pip install data_helper_2"

In [6]:

```
##Importing the data

df = pd.read_csv("pageblock.csv")
df.head()
df.columns= ['height','length','area','eccen','p_black','p_and','mean_tr','blackpix','b
lackand','wb_trans','classification']
df.classification = df.classification-1
```

In [5]:

```
# Adding dataframe to the analysis class
df_ca = completeanalysis(df)
```

```
creating separate list of numerical and categorical variables

categorical variable in the data... []

Numerical varialbles in the data... ['height', 'length', 'area', 'eccen',
'p_black', 'p_and', 'mean_tr', 'blackpix', 'blackand', 'wb_trans']

Splitting the data for train and test purpose 80% and 20 percent respectiv
ely..

Creating stratified samples of 5 fold...

A simple stratified sample and K-fold startified sample is created.
 The same              sample is used for comparing performance of multipl
e models
```

Meta data information

```
In [6]:
```

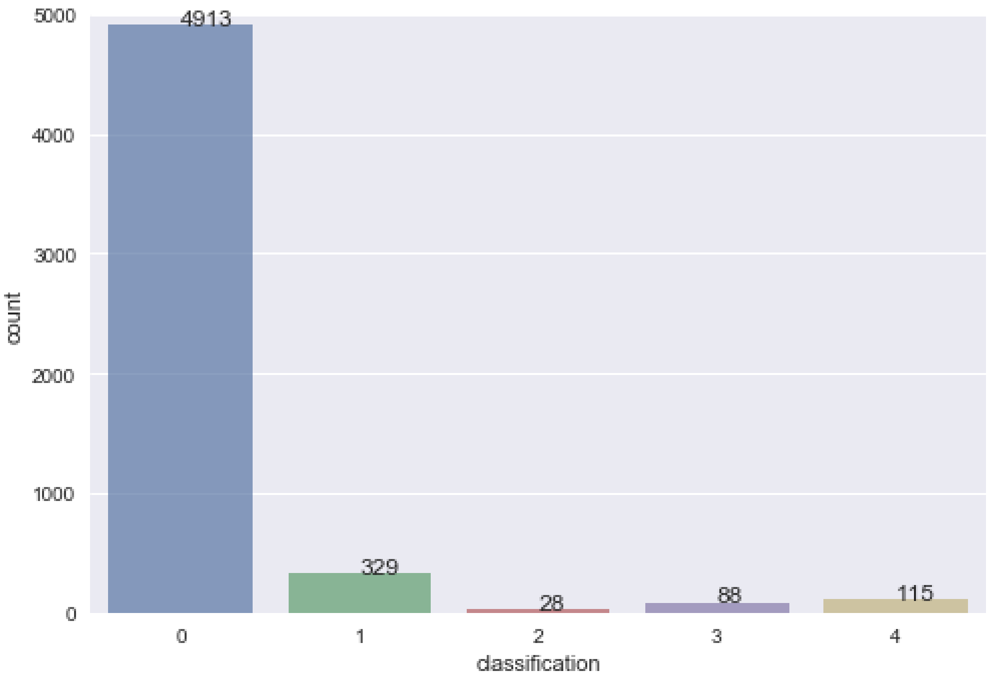
```
df_ca.col_meta_data()
```

```
Out[6]:
```

	Column Name	Number of NULL values	Number of Unique Values	Type
0	height	0.0	104	Numeric
1	length	0.0	452	Numeric
2	area	0.0	1395	Numeric
3	eccen	0.0	1511	Numeric
4	p_black	0.0	711	Numeric
5	p_and	0.0	700	Numeric
6	mean_tr	0.0	851	Numeric
7	blackpix	0.0	1069	Numeric
8	blackand	0.0	1718	Numeric
9	wb_trans	0.0	581	Numeric
10	classification	0.0	5	Numeric

```
In [7]:
```

```
df_ca.response_distribution()
```

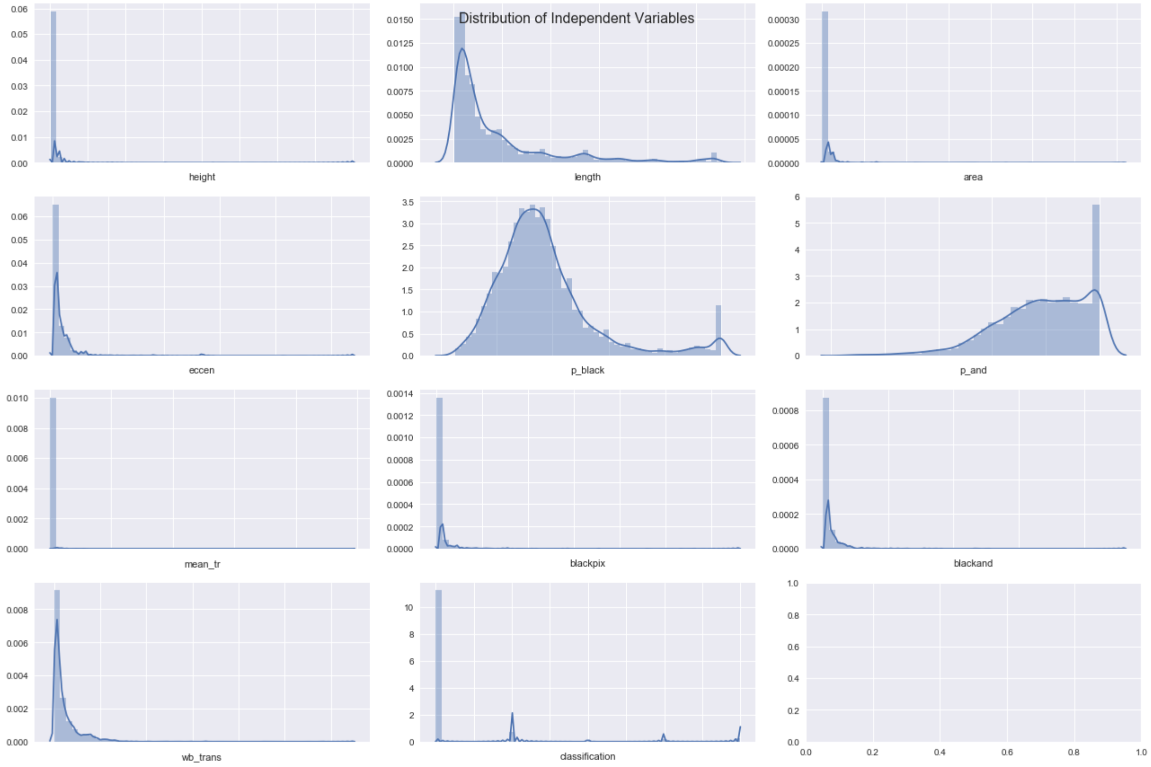


Visualisation

Univariate visualisation

In [8]:

```
# showing all the distributions plots
df_ca.distribution_plots()
```



Observation:

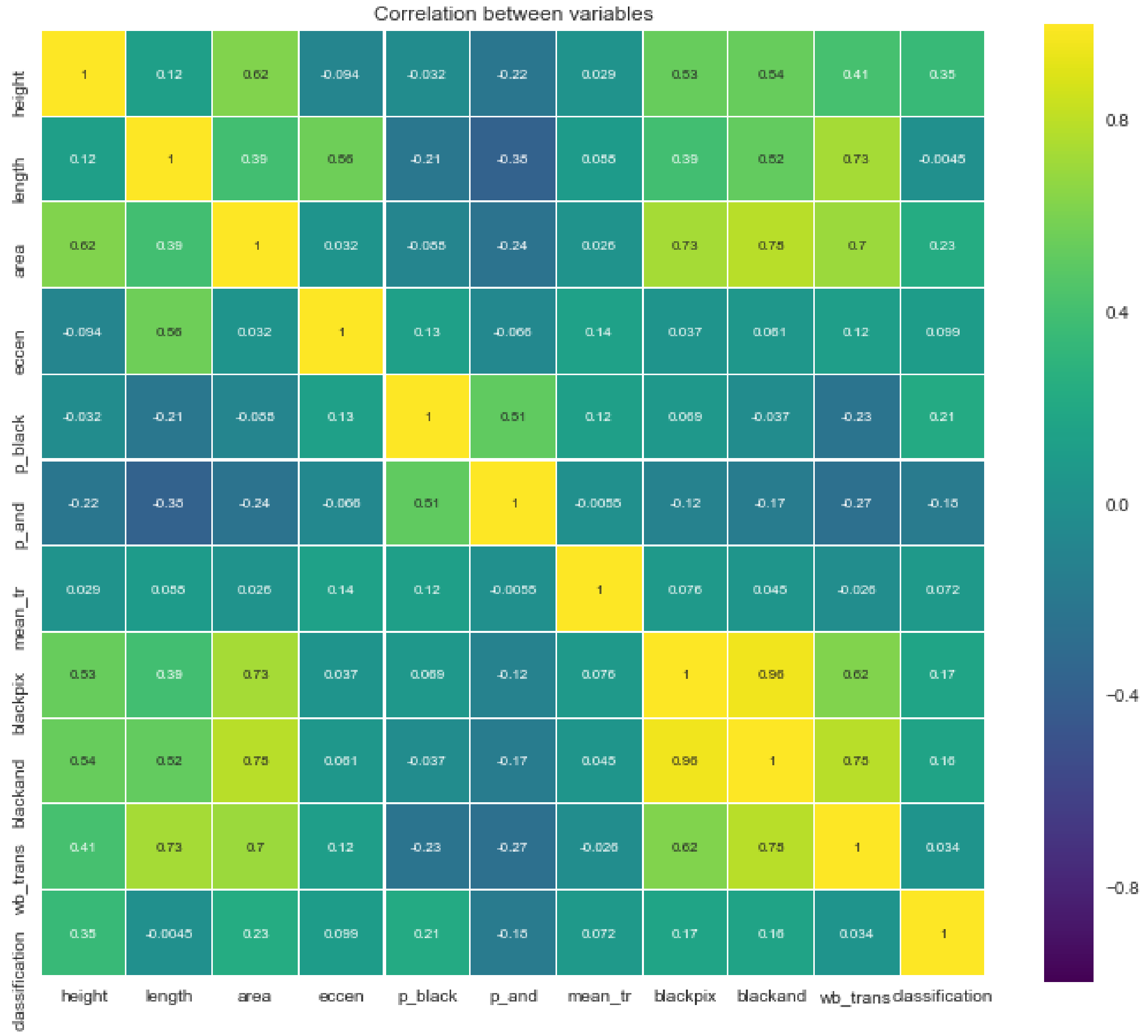
All the distribution are skewed

Bivariate visualisation

studying the correlation between all the variables

In [9]:

```
df_ca.correlation_plot()
```



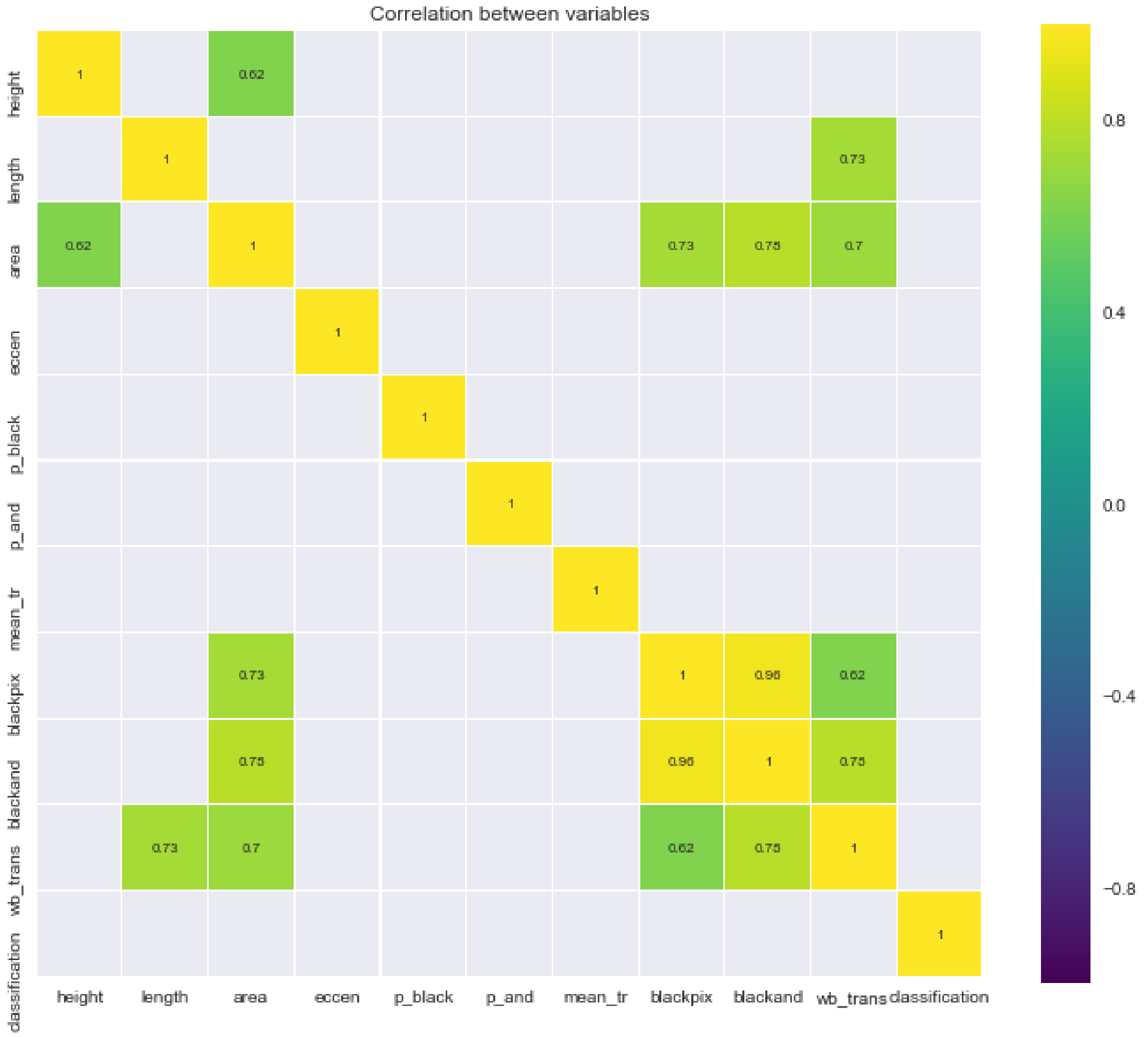
Correlation between variables

	height	length	area	eccen	p_black	p_and	mean_tr	blackpix	blackand	wb_trans	classification
height	1	0.12	0.62	-0.094	-0.032	-0.22	0.029	0.53	0.54	0.41	0.35
length	0.12	1	0.39	0.56	-0.21	-0.35	0.055	0.39	0.52	0.73	-0.0045
area	0.62	0.39	1	0.032	-0.055	-0.24	0.026	0.73	0.75	0.7	0.23
eccen	-0.094	0.56	0.032	1	0.13	-0.066	0.14	0.037	0.061	0.12	0.099
p_black	-0.032	-0.21	-0.055	0.13	1	0.51	0.12	0.069	-0.037	-0.23	0.21
p_and	-0.22	-0.35	-0.24	-0.066	0.51	1	-0.0055	-0.12	-0.17	-0.27	-0.15
mean_tr	0.029	0.055	0.026	0.14	0.12	-0.0055	1	0.076	0.045	-0.026	0.072
blackpix	0.53	0.39	0.73	0.037	0.069	-0.12	0.076	1	0.96	0.62	0.17
blackand	0.54	0.52	0.75	0.061	-0.037	-0.17	0.045	0.96	1	0.75	0.16
wb_trans	0.41	0.73	0.7	0.12	-0.23	-0.27	-0.026	0.62	0.75	1	0.034
classification	0.35	-0.0045	0.23	0.099	0.21	-0.15	0.072	0.17	0.16	0.034	1

visualization of having higher correlation than |0.5|

In [10]:

```
df_ca.correlation_plot(low=-0.6,high=0.6)
```



Observation:

We can observe some variables are highly correlated ., we have to check for multicollinearity in the further sections

especially

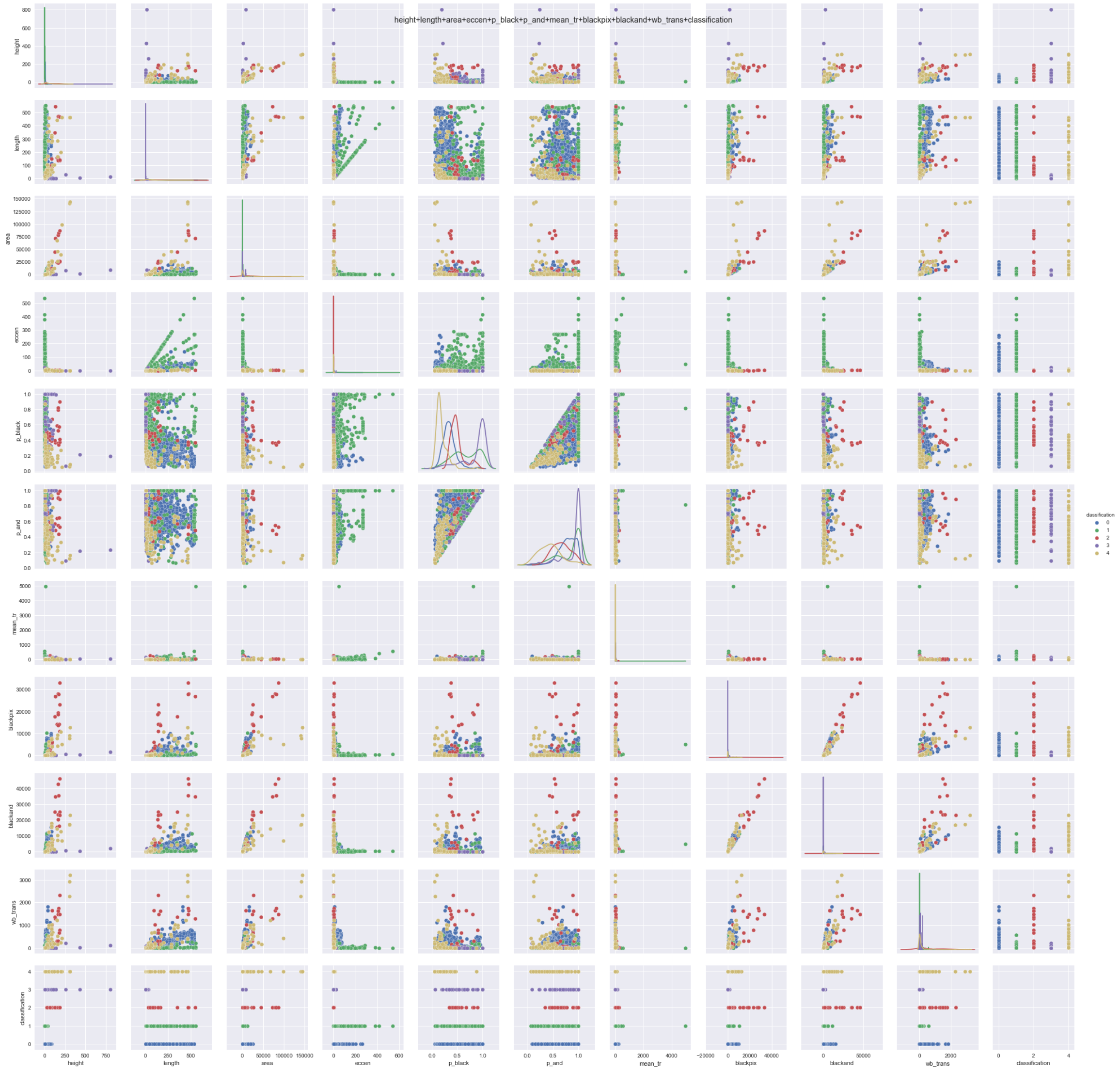
1. blackand and blackpix - 0.95
2. blackand,blackpix and area - 0.75
3. wbtrans and (length, area, blackpix, blackand) - (0.73,0.7,0.62,0.75)
4. height and area

seem to be highly correlated with each other, area and height have high correlation

Pairplot of all the variables

In [11]:

```
df_ca.pairplot()
```



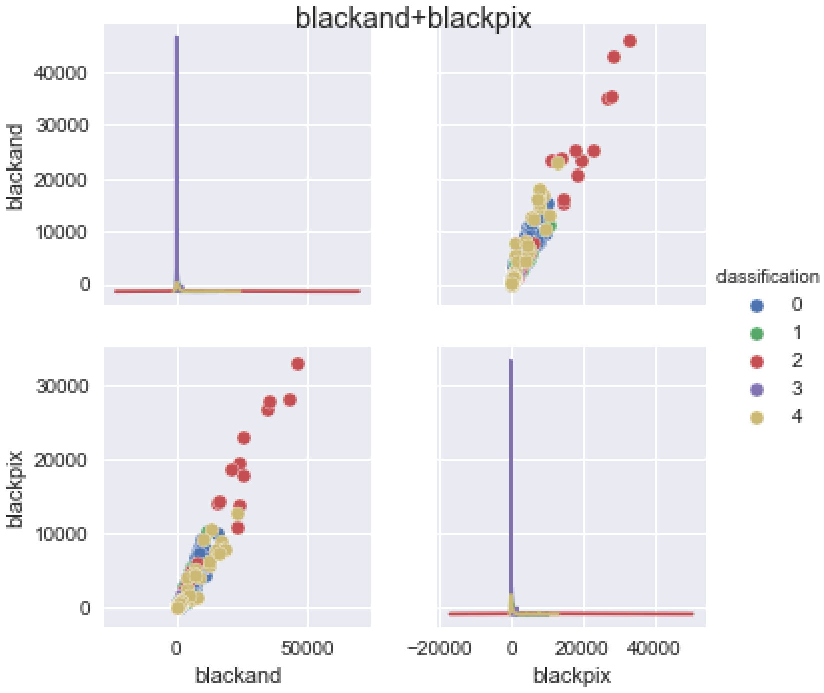
comparing the variables which we suspect to have high correlation

In [12]:

```
import seaborn as sns
```

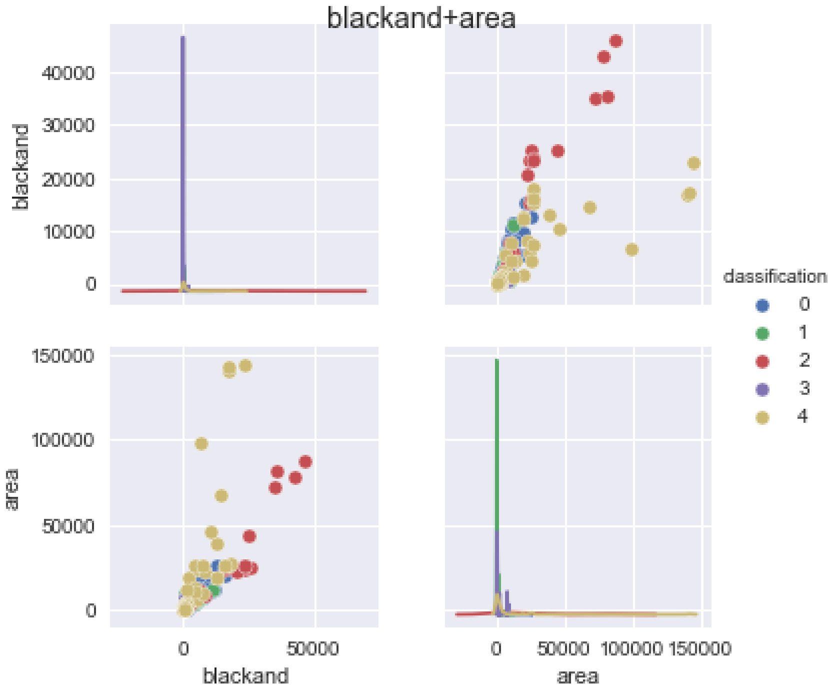
```
In [13]:
```

```
df_ca.pairplot(['blackand','blackpix'])
```



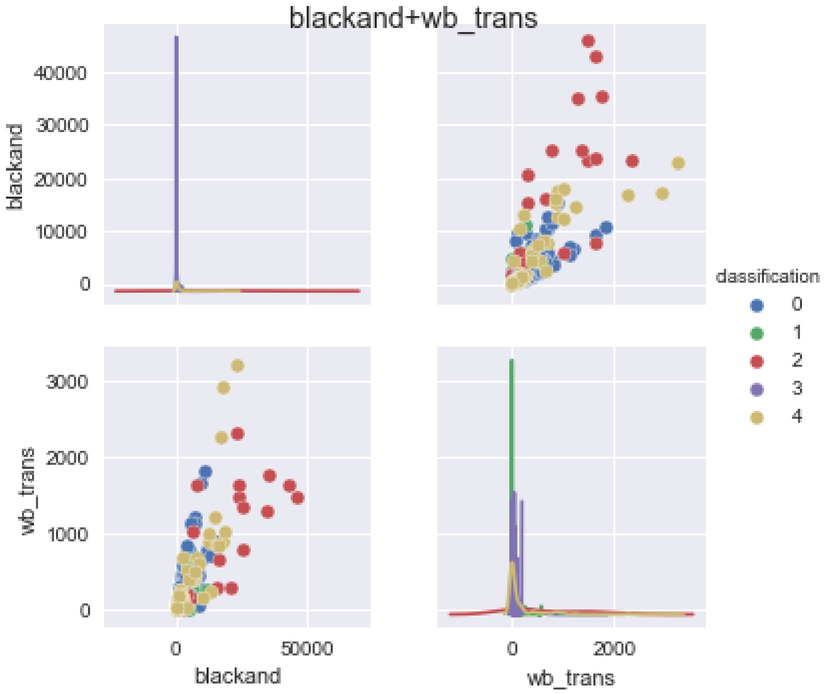
In [14]:

```
df_ca.pairplot(['blackand','area'])
```



In [15]:

```
df_ca.pairplot(['blackand','wb_trans'])
```

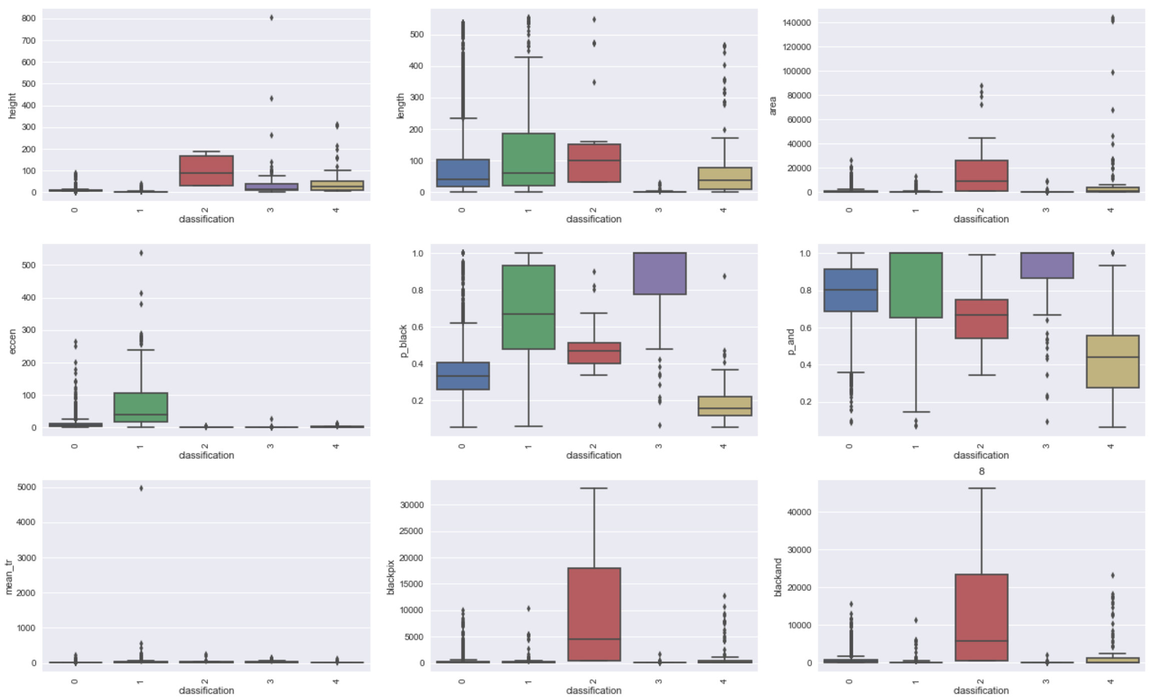


Plot for the numerical depedent variables and categorical Independent variables

In [16]:

```
df_ca.boxplots()
```

Variation of Numerical values WRT class response



Observations

All the variables are influenced by outliers

blackpix and blackand have similar distribution

Preliminary Feature elimination strategy

1. checking for variance inflation factor
2. checking for pvalue in multinominal logistic regression (in R)

In [17]:

```
VIF = df_ca.variance_explained()
VIF
```

Out[17]:

	Features	F Score	P Value	Support	VIF
0	height	436.524030	0.000000e+00	True	1.8
1	length	33.198844	2.122360e-27	True	5.7
2	area	251.063421	2.666377e-198	True	3.5
3	eccen	574.695126	0.000000e+00	True	2.3
4	p_black	696.405859	0.000000e+00	True	1.7
5	p_and	148.223104	1.573973e-120	True	1.7
6	mean_tr	34.117505	3.626194e-28	True	1.1
7	blackpix	587.079995	0.000000e+00	True	25.8
8	blackand	448.770720	0.000000e+00	True	39.5
9	wb_trans	135.335426	2.011844e-110	True	8.7

Observations:

we can accept variance inflation factor up to a value of 10

we can eliminate the either of the blackpix or blackand., we will go with blackand

In [18]:

```
from data_helper_2.data_helper import completeanalysis
df_ca = completeanalysis(df.drop('blackand',axis=1))
```

```
creating separate list of numerical and categorical variables

categorical variable in the data... []

Numerical varialbles in the data... ['height', 'length', 'area', 'eccen',
'p_black', 'p_and', 'mean_tr', 'blackpix', 'wb_trans']

Splitting the data for train and test purpose 80% and 20 percent respectiv
ely..

Creating stratified samples of 5 fold...

A simple stratified sample and K-fold startified sample is created.
 The same              sample is used for comparing performance of multipl
e models
```

In [19]:

```
df_ca.variance_explained()
```

Out[19]:

	Features	F Score	P Value	Support	VIF
0	height	436.524030	0.000000e+00	True	1.8
1	length	33.198844	2.122360e-27	True	5.7
2	area	251.063421	2.666377e-198	True	3.4
3	eccen	574.695126	0.000000e+00	True	2.3
4	p_black	696.405859	0.000000e+00	True	1.6
5	p_and	148.223104	1.573973e-120	True	1.7
6	mean_tr	34.117505	3.626194e-28	True	1.1
7	blackpix	587.079995	0.000000e+00	True	2.6
8	wb_trans	135.335426	2.011844e-110	True	5.5

Observations

Based on the F-score and VIF , also based on the pvalue from multi class logistic regression (R results , python didn't converge'), we can choose the above 8 factors

In [20]:

```
df_ca.compare_algorithm()
```

```
Fitting the model in simple stratified sample
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=Tru
e,
          intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
          penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
          verbose=0, warm_start=False)

Confusion matrix in the test dataset
[[977   5   0   0   1]
 [ 19  47   0   0   0]
 [  1   0   4   0   0]
 [  1   1   0  16   0]
 [ 16   0   0   0   7]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.96      0.99      0.98       983
          1       0.89      0.71      0.79        66
          2       1.00      0.80      0.89         5
          3       1.00      0.89      0.94        18
          4       0.88      0.30      0.45        23

avg / total       0.96      0.96      0.96      1095


The accuracy on the simple startified sample... 0.9598173515981735

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.96806569 0.94708029 0.96350365 0.936871
0.9532967 ]

The Accuracy mean : 0.9537634671176921

Accuracy - standard deviation 0.011221970143786501

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
LR: 0.953763 (0.011222)

Fitting the model in simple stratified sample
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto', kernel='rbf',
  max_iter=-1, probability=False, random_state=0, shrinking=True,
  tol=0.001, verbose=False)

Confusion matrix in the test dataset
[[981   2   0   0   0]
 [ 55  11   0   0   0]
 [  5   0   0   0   0]
 [  7   1   0  10   0]
 [ 23   0   0   0   0]]

The classification report for the fit..
```

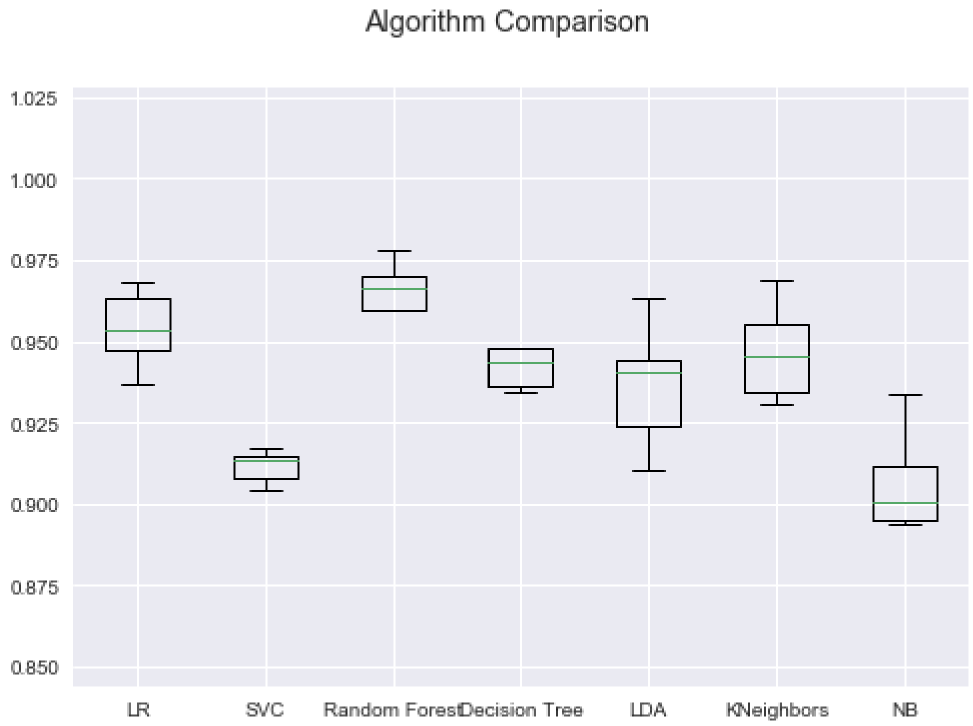
```
             precision    recall  f1-score   support

          0       0.92      1.00      0.96       983
          1       0.79      0.17      0.27        66
          2       0.00      0.00      0.00         5
          3       1.00      0.56      0.71        18
          4       0.00      0.00      0.00        23

avg / total       0.89      0.92      0.89      1095


The accuracy on the simple startified sample... 0.915068493150685

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.90784672 0.91332117 0.90419708 0.91674291
0.91483516]

The Accuracy mean : 0.9113886075524839

Accuracy - standard deviation 0.004659295224109462

The following will be returned ..
              1. accuracy of cv - sample


===========================================================================
======
SVC: 0.911389 (0.004659)

Fitting the model in simple stratified sample
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='entro
py',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=1,
            oob_score=False, random_state=0, verbose=0, warm_start=False)

Confusion matrix in the test dataset
[[974   3   1   0   5]
 [  7  58   0   1   0]
 [  1   0   4   0   0]
 [  1   2   0  15   0]
 [ 10   0   0   0  13]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.98      0.99      0.99       983
          1       0.92      0.88      0.90        66
          2       0.80      0.80      0.80         5
          3       0.94      0.83      0.88        18
          4       0.72      0.57      0.63        23

avg / total       0.97      0.97      0.97      1095


The accuracy on the simple startified sample... 0.971689497716895
```

```
Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.96989051 0.96624088 0.97810219 0.95974382
0.94230769]

The Accuracy mean : 0.9632570186573434

Accuracy - standard deviation 0.012037462678081168

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
Random Forest: 0.963257 (0.012037)

Fitting the model in simple stratified sample
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=Non
e,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=Non
e,
            splitter='best')

Confusion matrix in the test dataset
[[965   5   0   1  12]
 [ 11  53   0   0   2]
 [  1   0   3   0   1]
 [  1   2   0  15   0]
 [  5   0   0   0  18]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.98      0.98      0.98       983
          1       0.88      0.80      0.84        66
          2       1.00      0.60      0.75         5
          3       0.94      0.83      0.88        18
          4       0.55      0.78      0.64        23

avg / total       0.97      0.96      0.96      1095


The accuracy on the simple startified sample... 0.9625570776255707

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.9479927  0.94343066 0.96806569 0.93412626
0.93589744]

The Accuracy mean : 0.9459025489995632

Accuracy - standard deviation 0.012169365418748884

The following will be returned ..
              1. accuracy of cv - sample
```

```
==========================================================================
======
Decision Tree: 0.945903 (0.012169)

Fitting the model in simple stratified sample
LinearDiscriminantAnalysis(n_components=None, priors=None, shrinkage=None,
              solver='svd', store_covariance=False, tol=0.0001)

Confusion matrix in the test dataset
[[971   1   1   8   2]
 [ 29  32   1   3   1]
 [  1   0   2   0   2]
 [  3   1   0  14   0]
 [ 15   0   3   0   5]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.95      0.99      0.97       983
          1       0.94      0.48      0.64        66
          2       0.29      0.40      0.33         5
          3       0.56      0.78      0.65        18
          4       0.50      0.22      0.30        23

avg / total       0.93      0.94      0.93      1095


The accuracy on the simple startified sample... 0.9351598173515981

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.94069343 0.94434307 0.96350365 0.92406221
0.91025641]

The Accuracy mean : 0.9365717540662948

Accuracy - standard deviation 0.0181752271595648

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
LDA: 0.936572 (0.018175)

Fitting the model in simple stratified sample
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=1, n_neighbors=5, p=2,
           weights='uniform')

Confusion matrix in the test dataset
[[973   9   0   0   1]
 [ 11  55   0   0   0]
 [  2   0   3   0   0]
 [  2   1   0  15   0]
 [ 15   0   0   0   8]]

The classification report for the fit..
```

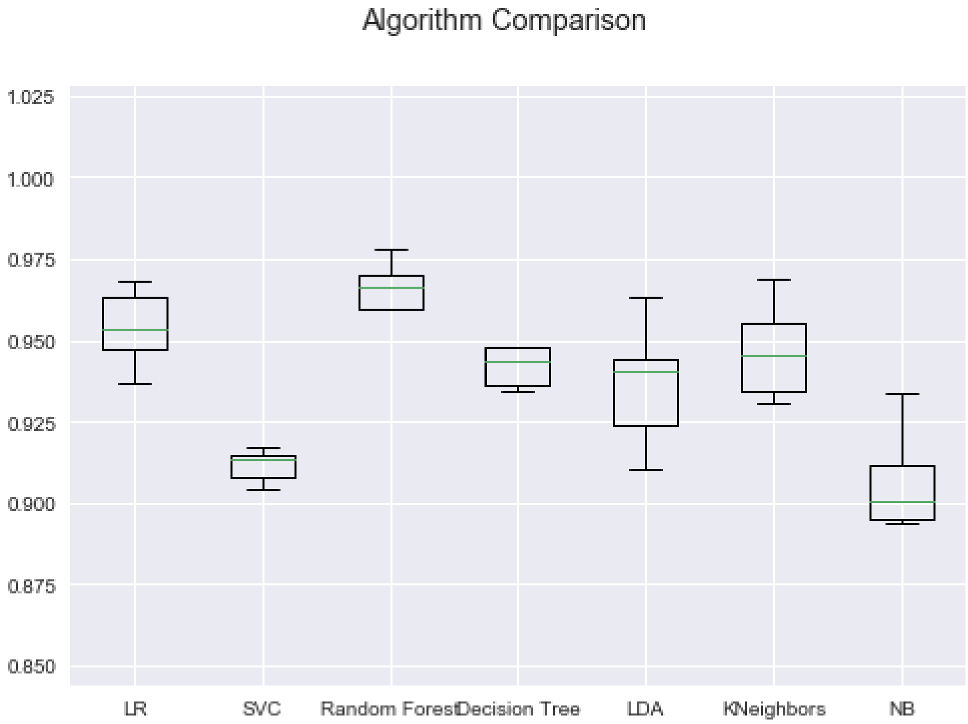
```
             precision    recall  f1-score   support

          0       0.97      0.99      0.98       983
          1       0.85      0.83      0.84        66
          2       1.00      0.60      0.75         5
          3       1.00      0.83      0.91        18
          4       0.89      0.35      0.50        23

avg / total       0.96      0.96      0.96      1095


The accuracy on the simple startified sample... 0.9625570776255707

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.94525547 0.95529197 0.9689781  0.93046661
0.93406593]

The Accuracy mean : 0.9468116174367301

Accuracy - standard deviation 0.014108882837932686

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
KNeighbors: 0.946812 (0.014109)

Fitting the model in simple stratified sample
GaussianNB(priors=None)

Confusion matrix in the test dataset
[[939   7   0  27  10]
 [ 24  39   1   1   1]
 [  1   0   3   0   1]
 [  0   1   0  17   0]
 [ 11   0   0   0  12]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.96      0.96      0.96       983
          1       0.83      0.59      0.69        66
          2       0.75      0.60      0.67         5
          3       0.38      0.94      0.54        18
          4       0.50      0.52      0.51        23

avg / total       0.93      0.92      0.93      1095


The accuracy on the simple startified sample... 0.9223744292237442

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.91149635 0.90054745 0.93339416 0.894785
0.89377289]
```

```
The Accuracy mean : 0.9067991690805416

Accuracy - standard deviation 0.014713703496232742

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
NB: 0.906799 (0.014714)
```



In [21]:

```
df_ca.plot_comparison_algorithm()
```



From the above graph we can say that in general the random forest algorithm performs better than most of the algorithm in average. Also randomforest performs well in the cross validation also with highest accuracy of 0.98

Best models in each model

1. Decision Trees
2. Random Forest
3. Support vector classifiers

A range of values is being used to check the best scenario for choosing the model parameters:

Random Forest and Decision Trees

1. max_depth= [50,100,200]
2. max_features= [None, "sqrt","log2"]
3. min_samples_leaf= [3, 4,5]
4. min_samples_split= [8,10,12]
5. n_estimators= [100,200,300,400]
6. criterion = ["gini","entropy"]

Support vector classifiers

1. Cs = [0.001, 0.01, 0.1, 1, 10]
2. gammas = [0.001, 0.01, 0.1, 1]
3. kernels = ['linear','rbf']

In [8]:

```
df_ca.best_fit_decision_trees()
```

```
Fitting 5 folds for each of 162 candidates, totalling 810 fits

[Parallel(n_jobs=1)]: Done 810 out of 810 | elapsed:   15.5s finished

{'criterion': 'entropy', 'max_depth': 200, 'max_features': None, 'min_samp
les_leaf': 5, 'min_samples_split': 12}

Fitting the model in simple stratified sample
DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=2
00,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=5, min_samples_split=12,
            min_weight_fraction_leaf=0.0, presort=False, random_state=Non
e,
            splitter='best')

Confusion matrix in the test dataset
[[968   6   0   1   8]
 [  7  58   0   1   0]
 [  1   0   4   0   0]
 [  0   1   0  17   0]
 [ 10   0   0   0  13]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.98      0.98      0.98       983
          1       0.89      0.88      0.89        66
          2       1.00      0.80      0.89         5
          3       0.89      0.94      0.92        18
          4       0.62      0.57      0.59        23

avg / total       0.97      0.97      0.97      1095


The accuracy on the simple startified sample... 0.9680365296803652

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.95620438 0.96350365 0.97810219 0.96340348
0.93131868]

The Accuracy mean : 0.9585064753933

Accuracy - standard deviation 0.015345489149120189

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
 The best model is
 {'criterion': 'entropy', 'max_depth': 200, 'max_features': None, 'min_sam
ples_leaf': 5, 'min_samples_split': 12}
The accuracy of the model 0.9585064753933
```

In [22]:

```
df_ca.best_fit_randomforest()
```

```
Fitting 5 folds for each of 324 candidates, totalling 1620 fits

[Parallel(n_jobs=1)]: Done 1620 out of 1620 | elapsed: 77.0min finished

{'max_depth': 50, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samp
les_split': 8, 'n_estimators': 400}

Fitting the model in simple stratified sample
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gin
i',
            max_depth=50, max_features='sqrt', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=3, min_samples_split=8,
            min_weight_fraction_leaf=0.0, n_estimators=400, n_jobs=1,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)

Confusion matrix in the test dataset
[[975   5   0   1   2]
 [  6  59   0   1   0]
 [  2   0   3   0   0]
 [  0   1   0  17   0]
 [  9   0   0   0  14]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.98      0.99      0.99       983
          1       0.91      0.89      0.90        66
          2       1.00      0.60      0.75         5
          3       0.89      0.94      0.92        18
          4       0.88      0.61      0.72        23

avg / total       0.97      0.98      0.97      1095


The accuracy on the simple startified sample... 0.9753424657534246

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.96715328 0.96167883 0.97718978 0.96340348
0.9532967 ]

The Accuracy mean : 0.9645444155553278

Accuracy - standard deviation 0.007781205650609066

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
 The best model is
 {'max_depth': 50, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_sam
ples_split': 8, 'n_estimators': 400}
The accuracy of the model 0.9645444155553278
```

In [29]:

```
df_ca.best_fit_svc()
```

```
Fitting 5 folds for each of 40 candidates, totalling 200 fits

[Parallel(n_jobs=1)]: Done 200 out of 200 | elapsed: 183.3min finished

{'C': 1, 'gamma': 0.001, 'kernel': 'linear'}

Fitting the model in simple stratified sample
SVC(C=1, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma=0.001, kernel='linear',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.001, verbose=False)

Confusion matrix in the test dataset
[[977   3   0   0   3]
 [ 10  55   0   1   0]
 [  2   0   3   0   0]
 [  0   1   0  17   0]
 [  8   0   0   0  15]]

The classification report for the fit..

             precision    recall  f1-score   support

          0       0.98      0.99      0.99       983
          1       0.93      0.83      0.88        66
          2       1.00      0.60      0.75         5
          3       0.94      0.94      0.94        18
          4       0.83      0.65      0.73        23

avg / total       0.97      0.97      0.97      1095


The accuracy on the simple startified sample... 0.9744292237442922

Fitting the model on 5-Fold startified sample..

The Accuracy for 5 folds are  [0.96806569 0.95894161 0.97262774 0.94602013
0.9496337 ]

The Accuracy mean : 0.9590577728435763

Accuracy - standard deviation 0.010238099414108998

The following will be returned ..
              1. accuracy of cv - sample


==========================================================================
======
 The best model is
 {'C': 1, 'gamma': 0.001, 'kernel': 'linear'}
The accuracy of the model 0.9590577728435763
```