#checking dimensions,missed values and maxima/minima

dataset.shape

Out[105]: (310, 15)

dataset.isnull().sum(axis=0)

Out[106]:

pelvic\_incidence 0

pelvic\_tilt 0

lumbar\_lordosis\_angle 0

sacral\_slope 0

pelvic\_radius 0

degree\_spondylolisthesis 0

pelvic\_slope 0

Direct\_tilt 0

thoracic\_slope 0

cervical\_tilt 0

sacrum\_angle 0

scoliosis\_slope 0

class 0

x 0

y 0

dtype: int64

dataset.describe()

Out[108]:

pelvic\_incidence pelvic\_tilt lumbar\_lordosis\_angle \

count 310.000000 310.000000 310.000000

mean 60.496653 17.542822 51.930930

std 17.236520 10.008330 18.554064

min 26.147921 -6.554948 14.000000

25% 46.430294 10.667069 37.000000

50% 58.691038 16.357689 49.562398

75% 72.877696 22.120395 63.000000

max 129.834041 49.431864 125.742385

sacral\_slope pelvic\_radius degree\_spondylolisthesis \

count 310.000000 310.000000 310.000000

mean 42.953831 117.920655 26.296694

std 13.423102 13.317377 37.559027

min 13.366931 70.082575 -11.058179

25% 33.347122 110.709196 1.603727

50% 42.404912 118.268178 11.767934

75% 52.695888 125.467674 41.287352

max 121.429566 163.071041 418.543082

pelvic\_slope Direct\_tilt thoracic\_slope cervical\_tilt \

count 310.000000 310.000000 310.000000 310.000000

mean 0.472979 21.321526 13.064511 11.933317

std 0.285787 8.639423 3.399713 2.893265

min 0.003220 7.027000 7.037800 7.030600

25% 0.224367 13.054400 10.417800 9.541140

50% 0.475989 21.907150 12.938450 11.953835

75% 0.704846 28.954075 15.889525 14.371810

max 0.998827 36.743900 19.324000 16.821080

sacrum\_angle scoliosis\_slope class x y

count 310.000000 310.000000 310.000000 3.100000e+02 3.100000e+02

mean -14.053139 25.645981 0.645161 -4.767513e-15 8.251464e-16

std 12.225582 10.450558 0.479238 4.221232e+01 1.859783e+01

min -35.287375 7.007900 0.000000 -6.081816e+01 -6.140102e+01

25% -24.289522 17.189075 0.000000 -3.077479e+01 -1.126134e+01

50% -14.622856 24.931950 1.000000 -1.154024e+01 6.940316e-01

75% -3.497094 33.979600 1.000000 2.250444e+01 1.104167e+01

max 6.972071 44.341200 1.000000 3.754472e+02 1.263135e+02

#visualising the data flow

Check the plot attached Back.png

#checking the optimal number of cluster we can get

By the graph, we can infer that that this data can be clustered to 4 clusters.

Check the graph “ Inertia.png”

#Plotting Count after clustering

Check the plot “countclass\_plot.png”

#cluster Visulaisation

Check the plot “BackPain.png”

#2.Visualising Confusion matrix

Check the plot “confusion\_matrix2.png”

#3.Applying 3 cross validation

accuracy.mean()

Out[122]: 0.92727005583308841

accuracy.std()

Out[123]: 0.026639466506580944

#4.CVgridSearch for pruning activities

Best accuracy=95.96%

Beat Param: max\_depth=3,min\_samples\_leaf=5,min\_samples\_split=5

#5.Prining Accuracy Score

Accuracy on training set: 0.972

Accuracy on test set: 0.952

#6.Printing Precision,Recall,Accuracy

precision recall f1-score support

0 0.96 0.92 0.94 26

1 0.95 0.97 0.96 36

avg / total 0.95 0.95 0.95 62