Machine Learning Techniques for Predictive Analytics

May 13, 2023

0.0.1 Import Heart Dataset from drive

```
[1]: data<-read.csv('heart.csv')
```

0.0.2 Data Display

```
[2]: data
```

A data.frame: 303 × 14

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpea
<int>	<int>	<int>	<int>	<int>	<int>	<int>	<int>	<int>	<dbl
63	1	3	145	233	1	0	150	0	2.3
37	1	2	130	250	0	1	187	0	3.5
41	0	1	130	204	0	0	172	0	1.4
56	1	1	120	236	0	1	178	0	0.8
57	0	0	120	354	0	1	163	1	0.6
57	1	0	140	192	0	1	148	0	0.4
56	0	1	140	294	0	0	153	0	1.3
44	1	1	120	263	0	1	173	0	0.0
52	1	2	172	199	1	1	162	0	0.5
57	1	2	150	168	0	1	174	0	1.6
54	1	0	140	239	0	1	160	0	1.2
48	0	2	130	275	0	1	139	0	0.2
49	1	1	130	266	0	1	171	0	0.6
64	1	3	110	211	0	0	144	1	1.8
58	0	3	150	283	1	0	162	0	1.0
50	0	2	120	219	0	1	158	0	1.6
58	0	2	120	340	0	1	172	0	0.0
66	0	3	150	226	0	1	114	0	2.6
43	1	0	150	247	0	1	171	0	1.5
69	0	3	140	239	0	1	151	0	1.8
59	1	0	135	234	0	1	161	0	0.5
44	1	2	130	233	0	1	179	1	0.4
42	1	0	140	226	0	1	178	0	0.0
61	1	2	150	243	1	1	137	1	1.0
40	1	3	140	199	0	1	178	1	1.4
71	0	1	160	302	0	1	162	0	0.4
59	1	2	150	212	1	1	157	0	1.6
51	1	2	110	175	0	1	123	0	0.6
65	0	2	140	417	1	0	157	0	0.8
53	1	2	130	197	1	0	152	0	1.2
58	1	0	100	234	0	1	156	0	0.1
47	1	0	110	275	0	0	118	1	1.0
52	1	0	125	212	0	1	168	0	1.0
58	1	0	146	218	0	1	105	0	2.0
57	1	1	124	261	0	1	141	0	0.3
58	0	1	136	319	1	0	152	0	0.0
61	1	0	138	166	0	0	125	1	3.6
42	1	0	136	315	0	1	125	1	1.8
52	1	0	128	204	1	1	156	1	1.0
59	1	2	126	218	1	1	134	0	2.2
40	1	0	152	223	0	1	181	0	0.0
61	1	0	140	207	0	0	138	1	1.9
46	1	0	140	311	0	1	120	1	1.8
59	1	3	134	204	0	1	162	0	0.8
57	1	1	154	232	0	0	164	0	0.0
57	1	0	110	335	0	1	143	1	3.0
55	0	0	128	205	0	2	130	1	2.0
61	1	0	148	203	0	1	161	0	0.0
58	1	0	114	318	0	2	140	0	4.4
58	0	0	170	225	1	0	146	1	2.8

0.0.3 Structure of Data

[3]:
```
str(data)
```

```
'data.frame':	303 obs. of  14 variables:
 $ age     : int  63 37 41 56 57 57 56 44 52 57 …
 $ sex     : int  1 1 0 1 0 1 0 1 1 1 …
 $ cp      : int  3 2 1 1 0 0 1 1 2 2 …
 $ trestbps: int  145 130 130 120 120 140 140 120 172 150 …
 $ chol    : int  233 250 204 236 354 192 294 263 199 168 …
 $ fbs     : int  1 0 0 0 0 0 0 0 1 0 …
 $ restecg : int  0 1 0 1 1 1 0 1 1 1 …
 $ thalach : int  150 187 172 178 163 148 153 173 162 174 …
 $ exang   : int  0 0 0 0 1 0 0 0 0 0 …
 $ oldpeak : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 …
 $ slope   : int  0 0 2 2 2 1 1 2 2 2 …
 $ ca      : int  0 0 0 0 0 0 0 0 0 0 …
 $ thal    : int  1 2 2 2 2 1 2 3 3 2 …
 $ target  : int  1 1 1 1 1 1 1 1 1 1 …
```

0.0.4 Remove Reduntant data From the Data Set (Data Preparation)

[4]:
```
data$sno<-NULL
```

0.0.5 Restructured Data

[5]:
```
str(data)
```

```
'data.frame':	303 obs. of  14 variables:
 $ age     : int  63 37 41 56 57 57 56 44 52 57 …
 $ sex     : int  1 1 0 1 0 1 0 1 1 1 …
 $ cp      : int  3 2 1 1 0 0 1 1 2 2 …
 $ trestbps: int  145 130 130 120 120 140 140 120 172 150 …
 $ chol    : int  233 250 204 236 354 192 294 263 199 168 …
 $ fbs     : int  1 0 0 0 0 0 0 0 1 0 …
 $ restecg : int  0 1 0 1 1 1 0 1 1 1 …
 $ thalach : int  150 187 172 178 163 148 153 173 162 174 …
 $ exang   : int  0 0 0 0 1 0 0 0 0 0 …
 $ oldpeak : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 …
 $ slope   : int  0 0 2 2 2 1 1 2 2 2 …
 $ ca      : int  0 0 0 0 0 0 0 0 0 0 …
 $ thal    : int  1 2 2 2 2 1 2 3 3 2 …
 $ target  : int  1 1 1 1 1 1 1 1 1 1 …
```

0.0.6 Transform classes for output i.e Classifier declaration for prediction

[6]:
```
data$target<-factor(ifelse(data$target==1,"positive","negative"))
```

0.0.7 Structure of Dataset with Changed target column

[7]:

```
str(data)
```

```
'data.frame':	303 obs. of  14 variables:
 $ age     : int  63 37 41 56 57 57 56 44 52 57 …
 $ sex     : int  1 1 0 1 0 1 0 1 1 1 …
 $ cp      : int  3 2 1 1 0 0 1 1 2 2 …
 $ trestbps: int  145 130 130 120 120 140 140 120 172 150 …
 $ chol    : int  233 250 204 236 354 192 294 263 199 168 …
 $ fbs     : int  1 0 0 0 0 0 0 0 1 0 …
 $ restecg : int  0 1 0 1 1 1 0 1 1 1 …
 $ thalach : int  150 187 172 178 163 148 153 173 162 174 …
 $ exang   : int  0 0 0 0 1 0 0 0 0 0 …
 $ oldpeak : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 …
 $ slope   : int  0 0 2 2 2 1 1 2 2 2 …
 $ ca      : int  0 0 0 0 0 0 0 0 0 0 …
 $ thal    : int  1 2 2 2 2 1 2 3 3 2 …
 $ target  : Factor w/ 2 levels "negative","positive": 2 2 2 2 2 2 2 2 2 2 …
```

0.0.8 BUILD MODEL (Data Sliced into Testing and Training Data)

[8]:

```
trainingset<-data[1:200,1:13]
testset<-data[201:303,1:13]
trainoutcome<-data[1:200,14]
testoutcome<-data[201:303,14]
```

0.0.9 import class library for rendering Classification algorithm

[9]:

```
library(class)
```

0.0.10 Predictor using 14-Nearest Neighbour Algorithm (k=14) for classification

[10]:

```
predictions<-knn(train=trainingset,cl=trainoutcome,k=14,test=testset)
```

0.0.11 Prediction outcome for all Data

[11]:

```
predictions
```

1. positive 2. positive 3. negative 4. positive 5. positive 6. positive 7. positive 8. positive 9. positive 10. positive 11. positive 12. positive 13. positive 14. positive 15. positive 16. positive 17. positive 18. positive 19. positive 20. positive 21. positive 22. positive 23. positive 24. positive 25. positive 26. positive 27. positive 28. positive 29. positive 30. positive 31. positive 32. positive 33. positive 34. positive 35. negative 36. positive 37. positive 38. positive 39. positive 40. positive 41. positive 42. positive 43. positive 44. negative 45. positive 46. positive 47. positive 48. positive 49. positive 50. positive 51. positive 52. positive 53. positive 54. positive 55. positive 56. positive 57. positive 58. positive 59. positive 60. positive 61. positive 62. positive 63. positive 64. positive 65. positive 66. positive 67. positive 68. positive 69. positive 70. positive 71. positive 72. positive 73. positive

74. positive 75. positive 76. positive 77. positive 78. positive 79. positive 80. positive 81. positive 82. positive 83. positive 84. positive 85. positive 86. positive 87. positive 88. positive 89. positive 90. positive 91. positive 92. positive 93. positive 94. positive 95. positive 96. positive 97. positive 98. positive 99. positive 100. positive 101. positive 102. positive 103. positive

Levels: 1. 'negative' 2. 'positive'

0.0.12 Prediction `Matrix` output

[12]:

```
table(testoutcome,predictions)
```

```
           predictions
testoutcome negative positive
   negative        3      100
   positive        0        0
```