Data Mining on Medical Images.

Jaimeen Fumakiya.
MSc. Big Data Analytics, Jai Hind College, Mumbai, Maharashtra, India.

Abstract: The aim of this article is to design an expert system for medical image diagnosis. We propose a method based on association rule mining combined with classification technique to enhance the diagnosis of medical images. This paper proposes data mining classifiers for medical image classification. There are many studies which uses different approaches like J48 decision tree and Random Forest (RF) classifiers for classifying CT scan brain images or association rule mining combined with classification technique to enhance the diagnosis of medical images. The performance of approaches is compared with two different classifiers Fuzzy-SVM and multilayer back propagation neural network.

Key words: brain tumor, image processing, association rule mining, CAD, Texture features, Data Mining classifiers.

Introduction

Medical imaging is the method and interaction of imaging the inside of a body for clinical analysis and medical intercession, just as visual portrayal of the capacity of certain organs or tissues. Medical images are one of the important data source which helps to diagnose many disease. Imaging plays a central role in the diagnosis of brain tumors. Computed Tomography (CT) has become a commonly performed procedure, which is a noninvasive, safe, and well-tolerated one.

The brain CT examines various structures of the brain to look for a mass, stroke, area of bleeding, or blood vessel abnormality. Computer Aided Diagnosis (CAD) system can help radiologists in interpreting brain CT images for tumor detection and classification. Association rule mining is one of the important techniques to do research in data mining. We try to explore the idea of finding association rules for the medical data. Here, we present a classification technique based on association rule mining for automatic diagnosing of medical images.

Dataset

For the Case Study Dataset is retrieved and used from Kaggle.com – “Brain Tumor”.

This is a brain tumor feature dataset including five first-order features and eight texture features with the target level (in the column Class). The brain tumor dataset includes 3762 with attributes such as Class, Mean, variance, skewness, etc. and The bt_dataset_t3 dataset includes 1644 rows of data with attributes such as Mean, Variance, ASM, Correlation etc.

Image column defines image name and Class column defines either the image has tumor or not (1 = Tumor, 0 = Non-Tumor) source: https://www.smir.ch/BRATS/Start2015

First-order and second other features are extracted for the above-mentioned Brat2015 MRI image using various mathematical terminologies.

Background

Texture perception plays an important role in the human visual system of recognition and interpretation. After feature extraction, the next important step is building, training and assessing the classifier. Researchers have tried to solve image classification problems using typical pattern recognition methods such as Support Vector Machine (SVM), Artificial Neural Networks (ANN) and Bayesian Networks (BN). Principal component analysis (PCA) technique used for selecting most relevant features and these features can be used for generating association rule based classifier.

All the textural features are derived from these angular nearest-neighbor gray-tone spatial-dependence matrices. In general, images with noise reduce the efficiency of the system. Therefore, preprocessing of medical images is essential for increasing the efficiency and reducing the complexity of the CAD system.

Exploring Dataset

data1 = read.csv("D:/Jaimeen e-books/MScBDA/Sem-2/data mining/Assignment/Brain Tumor.csv")

```
> data = data1[,2:15]

> head(data)

  Class     Mean  Variance  Standard.Deviation     Entropy    Skewness
1     0 6.535339  619.5878          24.89152  0.109059009 4.276477
2     0 8.749969  805.9576          28.38939  0.266538307 3.718116
3     1 7.341095 1143.8082          33.82023  0.001466811 5.061750
4     1 5.958145  959.7120          30.97922  0.001477124 5.677977
5     0 7.315231  729.5406          27.01001  0.146760596 4.283221
6     0 7.524109  607.3953          24.64539  0.214085626 3.729886

Kurtosis  Contrast     Energy        ASM Homogeneity
1 18.90057  98.61397 0.29331450 0.086033394   0.5309411
2 14.46462  63.85882 0.47505130 0.225673734   0.6513520
3 26.47956  81.86721 0.03191671 0.001018677   0.2682749
4 33.42885 151.22974 0.03202375 0.001025520   0.2438509
5 19.07911 174.98876 0.34384941 0.118232419   0.5011395
6 14.47174 105.07788 0.42158745 0.177735981   0.5981692
  Dissimilarity Correlation    Coarseness
1      4.473346   0.9819387 7.458341e-155
2      3.220072   0.9888344 7.458341e-155
3      5.981800   0.9780137 7.458341e-155
4      7.700919   0.9641892 7.458341e-155
5      6.834689   0.9727887 7.458341e-155
6      4.193146   0.9764850 7.458341e-155
```

```
dim(data1)

[1] 3762   15

> table(data1$Class)

0    1
2079 1683


> summary(data1$Energy)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.02473 0.06962 0.22550 0.20471 0.29890 0.58968


> names(data1)

 [1] "Image"              "Class"              "Mean"
 [4] "Variance"           "Standard.Deviation" "Entropy"
 [7] "Skewness"           "Kurtosis"           "Contrast"
[10] "Energy"             "ASM"                "Homogeneity"
[13] "Dissimilarity"      "Correlation"        "Coarseness"
```

Neural Networks

```
> nrow(data)
[1] 3762

> samplesize = 0.60 * nrow(data)

> set.seed(80)

> nr = sample(seq_len( nrow(data)), size = samplesize)

> dataTrain = data[nr, ]

> dataTest = data[-nr, ]

> mx = apply(data, 2, max)

> mx

             Class               Mean           Variance
      1.000000e+00       3.323997e+01       2.910582e+03
Standard.Deviation            Entropy           Skewness
      5.394981e+01       3.945386e-01       3.693129e+01
          Kurtosis           Contrast             Energy
```

```
   1.371640e+03     3.382574e+03     5.896818e-01
          ASM       Homogeneity     Dissimilarity
   3.477246e-01     8.109208e-01     2.782775e+01
    Correlation       Coarseness
   9.899724e-01     7.458341e-155
```

```
> mn = apply(data, 2, min)

> mn

        Class            Mean         Variance
   0.000000e+00     7.865906e-02     3.145628e+00
Standard.Deviation          Entropy         Skewness
   1.773592e+00     8.815796e-04     1.886014e+00
      Kurtosis         Contrast           Energy
   3.942402e+00     3.194733e+00     2.473117e-02
          ASM       Homogeneity     Dissimilarity
   6.116308e-04     1.054898e-01     6.811207e-01
    Correlation       Coarseness
   5.494262e-01     7.458341e-155
```

```
> scl = as.data.frame(scale(data, center = mn, scale = mx-mn))

> install.packages("neuralnet")

> library(neuralnet)

> NNtrain = scl[nr, ]

> NNtest = scl[-nr, ]

> NN = neuralnet(Class ~ Mean + Variance + Standard.Deviation + Entropy + Skewness+ Kurtosis +
Contrast + Energy +       ASM + Homogeneity + Dissimilarity + Correlation + Coarseness, NNtrain, hidden
= 6, linear.output = T)

> plot(NN)
```

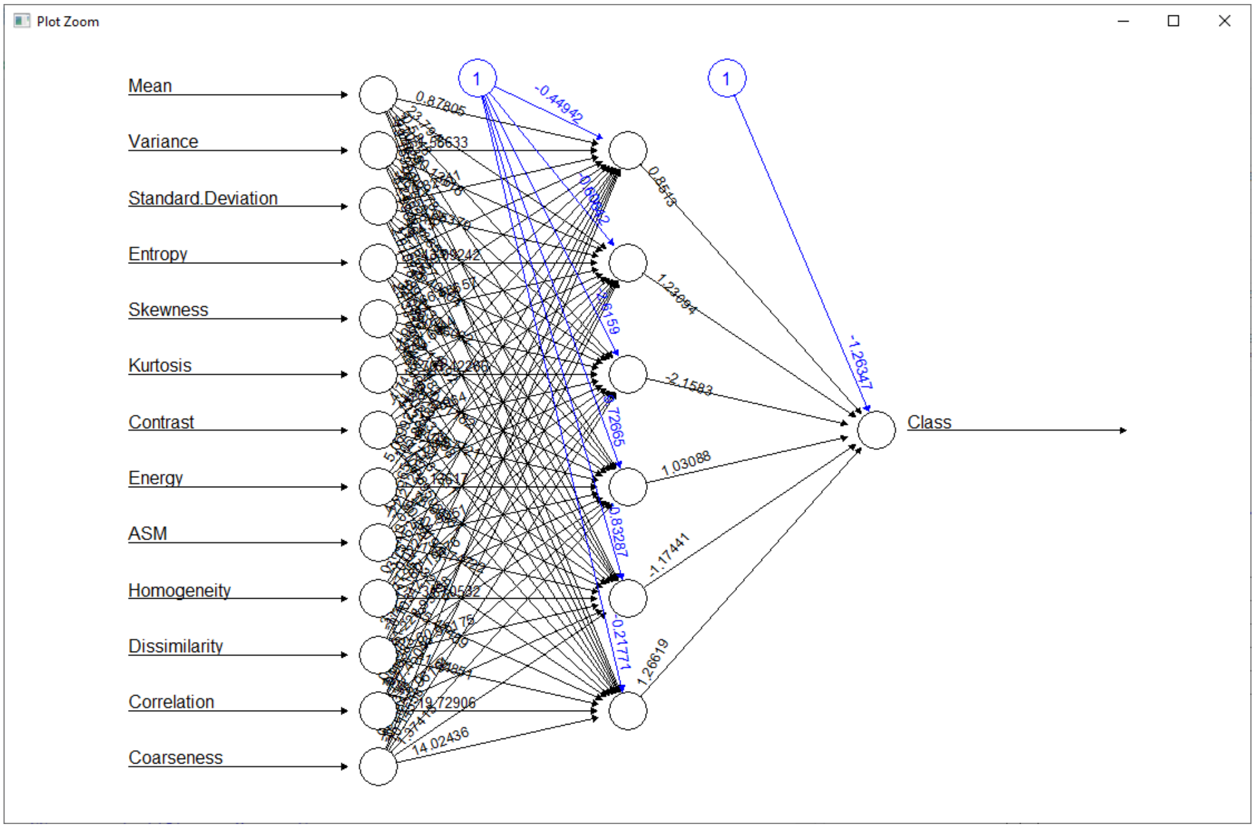
Plot Zoom
Mean
Variance
Standard.Deviation
Entropy
Skewness
Kurtosis
Contrast
Energy
ASM
Homogeneity
Dissimilarity
Correlation
Coarseness
1
1
-0.44942
0.87805
-2.1583
1.03088
-1.17441
1.26619
-1.26347
-0.21771
19.72906
14.02436
Class

Figure No. 1

Linear Regression

```
result <- lm(Class ~ Mean + Entropy + Skewness+ Kurtosis + Contrast + Energy +ASM + Homogeneity +
Dissimilarity + Correlation + Coarseness, data)

> summary(result)

Call:
lm(formula = Class ~ Mean + Entropy + Skewness + Kurtosis + Contrast +
    Energy + ASM + Homogeneity + Dissimilarity + Correlation +
    Coarseness, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1.35437 -0.09156 -0.00865  0.07716  0.97045

Coefficients: (1 not defined because of singularities)
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   -5.281e-01  1.754e-01  -3.011  0.00262 **
Mean          -3.565e-04  9.963e-04  -0.358  0.72051
Entropy       -9.620e+01  3.947e+00 -24.375  < 2e-16 ***
Skewness       3.970e-02  5.351e-03   7.419 1.45e-13 ***
Kurtosis      -7.107e-04  1.820e-04  -3.906 9.55e-05 ***
Contrast       9.046e-04  6.574e-05  13.761  < 2e-16 ***
Energy         9.002e-01  3.302e-01   2.726  0.00644 **
ASM            1.076e+02  4.113e+00  26.170  < 2e-16 ***
Homogeneity   -2.915e-01  1.251e-01  -2.329  0.01989 *
Dissimilarity -4.961e-02  6.152e-03  -8.064 9.86e-16 ***
Correlation    1.753e+00  1.547e-01  11.330  < 2e-16 ***
Coarseness            NA         NA      NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1836 on 3751 degrees of freedom
Multiple R-squared:  0.864,	Adjusted R-squared:  0.8637
F-statistic:  2384 on 10 and 3751 DF,  p-value: < 2.2e-16
```

```
> result <- lm(Class ~ Entropy + Skewness+ Kurtosis + Contrast +  ASM + Dissimilarity + Correlation, data)
> summary(result)

Call:
lm(formula = Class ~ Entropy + Skewness + Kurtosis + Contrast + 
    ASM + Dissimilarity + Correlation, data = data)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.36267 -0.09316 -0.01014  0.07687  0.96245 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -7.068e-01  1.493e-01  -4.735 2.27e-06 ***
Entropy       -8.860e+01  1.305e+00 -67.896  < 2e-16 ***
Skewness       4.151e-02  3.229e-03  12.855  < 2e-16 ***
Kurtosis      -7.166e-04  1.341e-04  -5.345 9.59e-08 ***
Contrast       8.574e-04  5.138e-05  16.687  < 2e-16 ***
ASM            1.000e+02  1.553e+00  64.419  < 2e-16 ***
Dissimilarity -4.065e-02  3.728e-03 -10.904  < 2e-16 ***
Correlation    1.820e+00  1.515e-01  12.014  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1838 on 3754 degrees of freedom
Multiple R-squared:  0.8636,	Adjusted R-squared:  0.8634 
F-statistic:  3397 on 7 and 3754 DF,  p-value: < 2.2e-16


> test_data = read.csv("D:/Jaimeen e-books/MScBDA/Sem-2/data
mining/Assignment/bt_dataset_t3.csv")

> PRED_data <- predict(result,test_data, level=0.95,interval = "confidence")
> PRED_data

         fit        lwr       upr
1    4.577264   4.3569467  4.797580
2   11.483434  11.0190175 11.947850
3   11.833029  11.3574145 12.308643
4    9.211218   8.8134868  9.608949
5    5.239712   4.9919588  5.487465
6          NA         NA        NA
7    6.648213   6.3412498  6.955176
8   12.736004  12.2335356 13.238473
9   10.749449  10.3089113 11.189987
10  10.480940  10.0485367 10.913343
11  12.785895  12.2674319 13.304358
12  12.419860  11.8180191 13.021700
13  11.107206  10.6553185 11.559093
14  12.015571  11.5358882 12.495254
15  11.409148  10.9456861 11.872609
16   3.788604   3.5961712  3.981036
17  12.448917  11.8595417 13.038293
18  12.430553  11.9322564 12.928850
19  11.281489  10.8228210 11.740157
20  10.814255  10.3704852 11.258024
21  12.344097  11.8534692 12.834724
22  12.310952  11.8236026 12.798300
23  12.159788  11.6737921 12.645783
24         NA         NA        NA
25   4.603650   4.3792364  4.828063
```

26	NA	NA	NA
27	8.430895	5.9551639	10.906627
28	5.077054	4.8333065	5.320802
29	10.992729	10.5432714	11.442186
30	12.782800	12.2638396	13.301760
31	12.440819	11.9425663	12.939072
32	NA	NA	NA
33	NA	NA	NA
34	12.287630	11.5938171	12.981443
35	5.120168	4.8739138	5.366423
36	11.177941	10.7215120	11.634370
37	10.975544	10.5268445	11.424244
38	12.012836	11.5326471	12.493024
39	12.224717	11.7394524	12.709981
40	NA	NA	NA
41	11.657482	10.8722211	12.442743
42	NA	NA	NA
43	NA	NA	NA
44	NA	NA	NA
45	NA	NA	NA
46	12.461810	11.9635458	12.960074
47	11.915005	11.4366110	12.393398
48	12.516486	12.0244840	13.008489
49	NA	NA	NA
50	NA	NA	NA
51	NA	NA	NA
52	12.644234	12.1264716	13.161997
53	12.584822	12.0818285	13.087815
54	11.381126	10.9178938	11.844357
55	12.472274	11.9776419	12.966906
56	NA	NA	NA
57	NA	NA	NA
58	10.930446	10.4808010	11.380090
59	10.019782	9.6028670	10.436697
60	10.593095	10.1553632	11.030827
61	11.264502	10.8062532	11.722751
62	12.478568	11.9421614	13.014974
63	11.567541	11.0969272	12.038155
64	12.333188	11.8380668	12.828308
65	NA	NA	NA
66	11.768053	11.2883738	12.247732
67	11.014028	9.8483594	12.179698
68	9.825103	9.4112246	10.238981
69	NA	NA	NA
70	NA	NA	NA
71	3.856702	3.6632434	4.050160
72	3.332721	3.1592278	3.506213
73	NA	NA	NA
74	3.289059	3.1175090	3.460608
75	12.279518	11.7803249	12.778710
76	10.275837	8.7944953	11.757180
77	11.500675	11.0349051	11.966444
78	11.280525	10.8173125	11.743738
79	NA	NA	NA
80	NA	NA	NA
81	11.051746	10.5999692	11.503524
82	11.106781	10.6523361	11.561226
83	11.799974	11.3211004	12.278848
84	12.807164	12.3014846	13.312844
85	12.668093	12.1478247	13.188361
86	12.121134	11.6359922	12.606275
87	10.875978	10.4317347	11.320221
88	12.595497	12.0228932	13.168102
89	12.193734	11.6996201	12.687847
90	11.040710	10.5898159	11.491603
91	12.679027	12.1126632	13.245390
92	12.781086	12.2371512	13.325021
93	12.734003	12.1844305	13.283575
94	NA	NA	NA
95	11.162771	9.9956061	12.329936
96	11.562207	11.0939137	12.030500
97	12.700845	12.1213530	13.280336
98	12.458271	11.9626041	12.953939
99	12.334567	11.6993723	12.969761
100	11.306334	10.8475883	11.765080
101	NA	NA	NA
102	12.549049	12.0536974	13.044400
103	12.622858	12.1200146	13.125702
104	12.842118	12.3221669	13.362069
105	12.766117	12.2585337	13.273701
106	11.148942	9.9709963	12.326887
107	NA	NA	NA
108	12.827432	12.3026507	13.352213
109	12.554891	11.9975788	13.112204
110	12.478398	11.9815957	12.975201
111	NA	NA	NA
112	11.674418	10.6935625	12.655274
113	11.993895	11.5156009	12.472190
114	12.607178	12.0667459	13.147609
115	12.220807	11.7362604	12.705354
116	12.754102	12.2328108	13.275393
117	NA	NA	NA
118	12.161359	11.6774696	12.645249
119	11.511091	11.0461634	11.976019
120	12.303656	11.8119842	12.795327
121	4.757851	4.5295838	4.986118

122	12.299407	11.8104222	12.788392
123	12.598633	11.9801763	13.217089
124	9.770559	9.3619112	10.179206
125	12.507103	11.9468485	13.067357
126	11.634386	11.1639020	12.104870
127	12.302646	11.7453604	12.859931
128	11.839668	11.3611065	12.318230
129	12.065165	11.5790477	12.551283
130	4.780173	4.5499791	5.010367
131	12.530319	12.0315022	13.029136
132	12.727145	12.1774713	13.276819
133	12.531833	12.0342369	13.029429
134	11.581347	11.1132688	12.049426
135	12.644141	12.1445949	13.143687
136	NA	NA	NA
137	NA	NA	NA
138	4.767945	4.5372136	4.998676
139	12.248091	11.5980930	12.898090
140	11.256024	10.7818648	11.730183
141	8.891105	8.5127581	9.269452
142	9.347186	8.9519347	9.742438
143	10.376200	9.9463148	10.806085
144	11.269262	10.8118020	11.726723
145	12.630503	12.1183973	13.142608
146	NA	NA	NA
147	11.821567	11.0186136	12.624520
148	NA	NA	NA
149	12.020617	11.5348283	12.506406
150	12.050475	11.5644665	12.536483
151	11.791599	11.3154337	12.267764
152	11.609463	11.1388384	12.080089
153	10.659824	10.2198963	11.099751
154	12.413833	11.9242541	12.903411
155	12.402764	11.7193297	13.086198
156	NA	NA	NA
157	12.191476	11.7078506	12.675101
158	11.781044	11.3086745	12.253414
159	11.840028	11.3646453	12.315412
160	11.575620	11.1089555	12.042284
161	12.797668	12.2655414	13.329795
162	12.755211	12.2487972	13.261626
163	7.886346	7.5411765	8.231516
164	3.743674	3.5545079	3.932840
165	NA	NA	NA
166	4.020498	3.8195053	4.221491
167	NA	NA	NA
168	12.702934	12.1911143	13.214753
169	12.683509	12.1847447	13.182274
170	4.401416	4.1867778	4.616055
171	11.907726	11.4289927	12.386460
172	11.560821	11.0934473	12.028195
173	-4.347618	-12.1103950	3.415160
174	12.537956	11.9964076	13.079503
175	NA	NA	NA
176	12.725054	12.2116060	13.238502
177	NA	NA	NA
178	NA	NA	NA
179	11.699887	11.2292307	12.170544
180	11.659854	11.1903827	12.129325
181	12.300898	11.8003320	12.801464
182	11.506705	10.6115191	12.401892
183	12.185883	11.6991738	12.672593
184	11.880691	11.4020125	12.359369
185	12.586154	12.0873575	13.084951
186	11.378854	10.9130504	11.844657
187	9.866518	9.4482940	10.284741
188	11.479286	11.0108519	11.947721
189	12.850422	12.3327474	13.368096
190	11.993878	11.4941390	12.493617
191	11.953768	11.4745766	12.432960
192	10.993464	10.5453316	11.441596
193	11.722586	11.2509144	12.194257
194	12.176728	11.6880423	12.665413
195	NA	NA	NA
196	11.658508	11.1870600	12.129955
197	10.980625	10.5314522	11.429798
198	10.669598	10.2311141	11.108082
199	10.934965	10.4886852	11.381245
200	12.029896	11.5492966	12.510496
201	NA	NA	NA
202	NA	NA	NA
203	NA	NA	NA
204	12.860078	12.3557384	13.364417
205	NA	NA	NA
206	6.189955	5.9005953	6.479315
207	NA	NA	NA
208	NA	NA	NA
209	3.409396	3.2315896	3.587203
210	NA	NA	NA
211	12.507245	11.9829830	13.031506
212	12.386094	11.8875525	12.884635
213	12.011465	11.5231242	12.499806
214	12.644900	12.1276502	13.162151
215	11.278833	10.8138583	11.743809
216	4.421702	4.2058250	4.637579
217	3.413741	3.2368305	3.590652

218	3.634266	3.4486234	3.819908
219	11.752062	11.2770060	12.227118
220	12.462891	11.9673656	12.958417
221	11.978820	11.4973480	12.460291
222	NA	NA	NA
223	12.462694	11.7910286	13.134360
224	12.516976	12.0158420	13.018111
225	NA	NA	NA
226	6.127120	5.8453677	6.408873
227	10.053429	9.6214937	10.485365
228	11.925761	11.4285357	12.422987
229	12.082396	11.5789952	12.585796
230	9.343202	8.9427880	9.743616
231	10.479870	10.0465578	10.913183
232	NA	NA	NA
233	10.552200	10.1176677	10.986732
234	12.755422	12.2313078	13.279536
235	12.377533	11.8835355	12.871531
236	NA	NA	NA
237	NA	NA	NA
238	12.385063	11.8937384	12.876388
239	12.681590	12.1862082	13.176972
240	12.746221	12.2492486	13.243194
241	12.852818	12.3426542	13.362982
242	NA	NA	NA
243	NA	NA	NA
244	NA	NA	NA
245	NA	NA	NA
246	11.440584	10.9698787	11.911289
247	10.768280	10.3209740	11.215587
248	3.647868	3.4616002	3.834136
249	3.783392	3.5905385	3.976245
250	9.776900	9.3625269	10.191273
251	12.132658	11.6495599	12.615756
252	12.491935	11.9948074	12.989063
253	NA	NA	NA
254	NA	NA	NA
255	4.893918	4.6581447	5.129691
256	11.996974	11.5166098	12.477338
257	12.354668	11.8632844	12.846052
258	12.479320	11.9874753	12.971166
259	NA	NA	NA
260	NA	NA	NA
261	11.171234	10.0013090	12.341160
262	12.417162	11.9094629	12.924860
263	12.637412	12.1252550	13.149569
264	5.999994	5.7175978	6.282391
265	NA	NA	NA
266	11.242745	10.7851339	11.700357
267	10.182701	9.7596831	10.605720
268	11.222380	10.7645845	11.680176
269	11.508291	11.0417935	11.974788
270	12.020752	11.5414355	12.500069
271	9.074977	8.6896587	9.460296
272	11.027219	10.5744853	11.479954
273	11.805352	11.3272755	12.283429
274	11.440850	10.9767492	11.904951
275	12.248911	11.7598227	12.737999
276	NA	NA	NA
277	NA	NA	NA
278	12.729324	12.1974767	13.261171
279	12.633636	12.0898170	13.177455
280	12.710002	12.2022454	13.217758
281	7.437257	7.1011980	7.773317
282	NA	NA	NA
283	7.452089	7.1187289	7.785450
284	10.847945	10.4024587	11.293431
285	11.954125	11.4760729	12.432178
286	12.262257	11.7549317	12.769583
287	12.728470	12.2264558	13.230484
288	12.712766	12.1783353	13.247197
289	12.461284	11.8016201	13.120948
290	12.757324	12.2420588	13.272589
291	7.731767	7.3903081	8.073226
292	11.912667	11.4313859	12.393948
293	12.282862	11.7792861	12.786439
294	10.919817	10.4717206	11.367913
295	12.091951	11.6003700	12.583531
296	9.729087	7.8807026	11.577472
297	12.491548	12.0003860	12.982711
298	12.717572	12.2215458	13.213598
299	NA	NA	NA
300	NA	NA	NA
301	12.666596	12.1709594	13.162232
302	11.545139	11.0713999	12.018877
303	11.807726	10.8690276	12.746424
304	9.957342	9.5362936	10.378389
305	11.806749	11.3303832	12.283115
306	NA	NA	NA
307	9.418570	9.0197471	9.817394
308	NA	NA	NA
309	NA	NA	NA
310	9.068352	8.6842573	9.452446
311	NA	NA	NA
312	12.774725	12.2640515	13.285398
313	5.523100	5.2626795	5.783521

```
314   12.074342  11.5846925 12.563992
315   12.546863  12.0439751 13.049750
316   12.539990  12.0410287 13.038950
317   12.687352  12.1778636 13.196841
318    6.040304   5.7587565  6.321851
319    7.683943   7.3398402  8.028045
320   10.171824   9.7372115 10.606436
321   12.789355  12.2838667 13.294844
322         NA         NA        NA
323         NA         NA        NA
324         NA         NA        NA
325         NA         NA        NA
326   11.164158   9.9970368 12.331279
327    9.119855   8.7212094  9.518500
328         NA         NA        NA
329    7.133368   6.8127247  7.454011
330   12.720695  12.2144050 13.226984
331   10.599384  10.1625636 11.036205
332   11.366656  10.4348928 12.298418
333   12.592457  12.0415350 13.143380

 [ reached getOption("max.print") -- omitted 1311 rows ]

> plot(result)
```

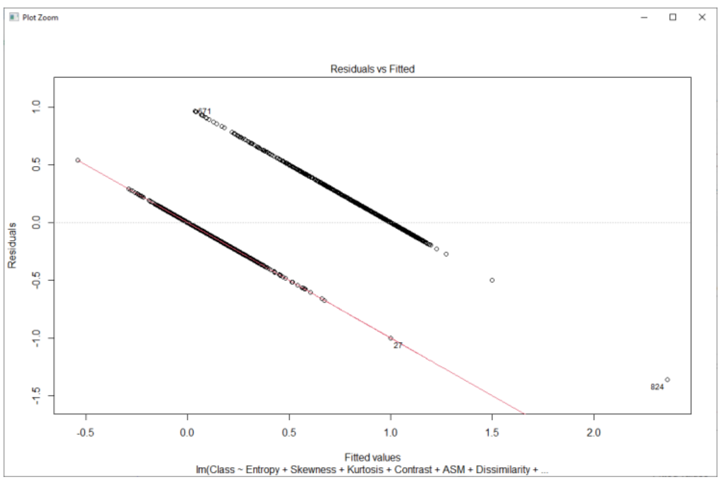

Figure No. 2

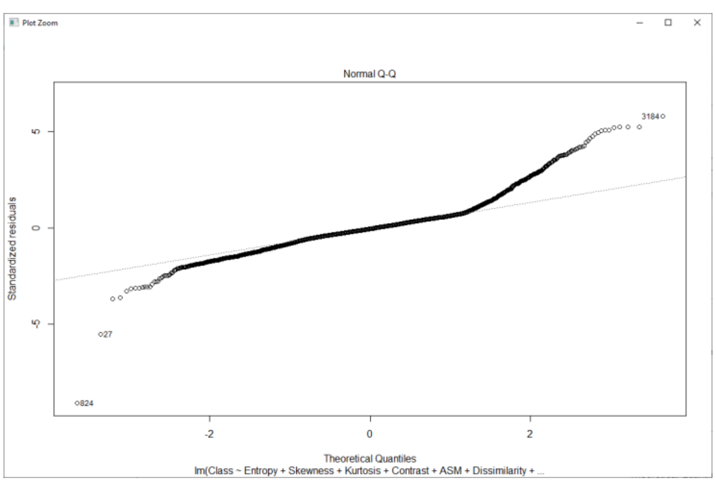

Figure No. 3

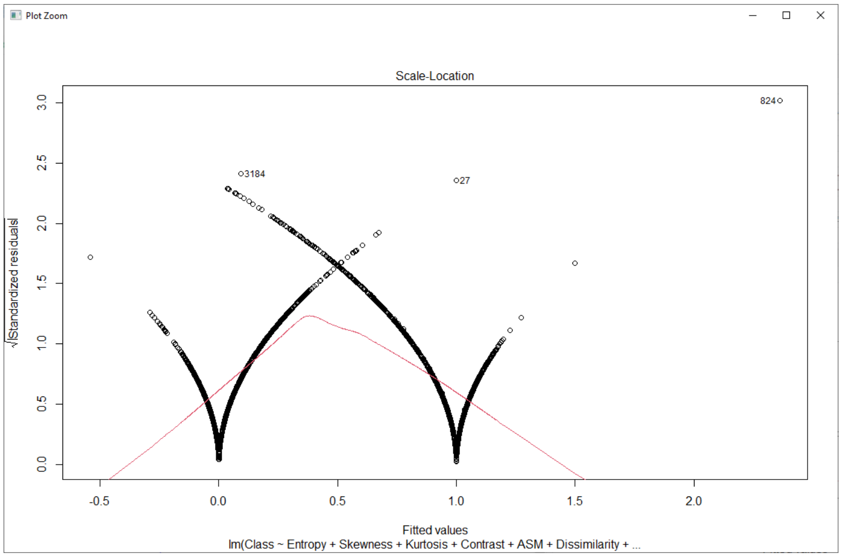

Figure No. 4

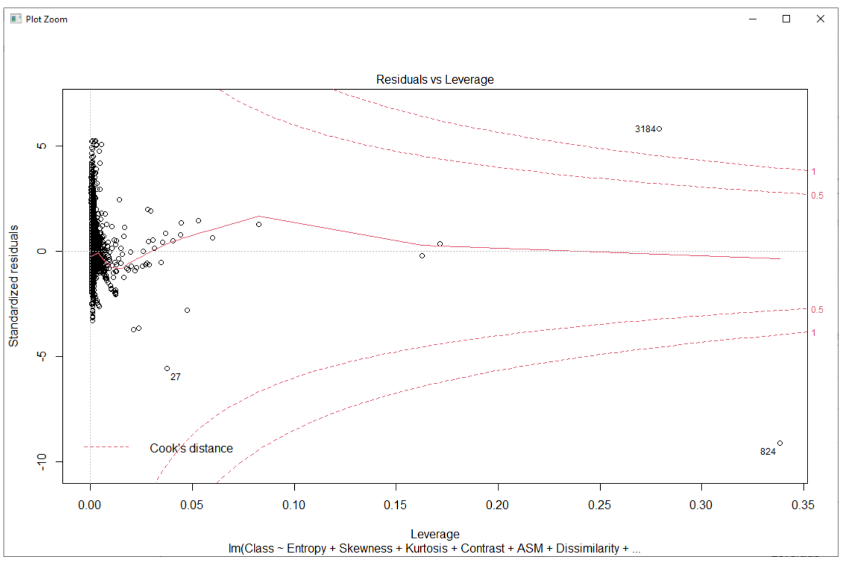

Figure No. 5

Fitting Decision Tree Model

> test_data = read.csv("D:/Jaimeen e-books/MScBDA/Sem-2/data mining/Assignment/bt_dataset_t3.csv")

> library(rpart)

> library(rpart.plot)

```
> tree.model <- rpart(Class ~ Mean + Entropy + Skewness+ Kurtosis + Contrast + Energy + ASM +
Homogeneity + Dissimilarity + Correlation + Coarseness, data, method = 'class')

> rpart.plot(tree.model)
```

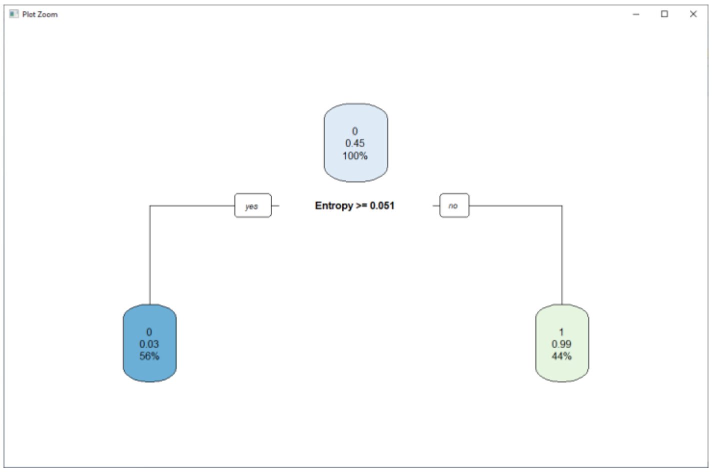

Figure No. 6

```
> tree.predict<- predict(tree.model, test_data, type = 'class')
>sum(tree.predict == 0)
[1] 1644
> sum(tree.predict == 1)
[1] 0
```

Fitting Support Vector Machine Model

```
> set.seed(123)
> library(e1071)
> svmfit = svm(Class ~., data = data,type = 'C-classification', kernel = 'linear', cost = 10, scale = FALSE)

WARNING: reaching max number of iterations
> y_pred = predict(svmfit, newdata = test_data)
> sum(y_pred == 0)
[1] 1275
> sum(y_pred == 1)
[1] 0
```

Experimental Results

On the dataset, I evaluated a few models. The data is tallied, demonstrating variations in accuracy. Table shows the accuracy of each model

Methods	Accuracy
Linear Regression	0.99235
Decision Tree	0.9985
Support Vector Machine	0.99979

Table No. 1

Association Rule Mining

As in any learning process for building a classifier, the classification performed with association rule mining comprised two steps. The first one is represented by the training of the system, while the second one deals with the classification of the new images.

First step is to generate the frequent item sets and second step is to generate association rules from the frequent item sets.

Conclusion

In this paper, we have described the classification techniques for CT scan brain images. We have used data mining classifiers Decision Tree, SVM and Linear Regression Data Mining tool for our experiment. The classifiers are compared as shown in Table 1.

Reference

John, S., Ioannis, V., et al.: Computer Aided Diagnosis based on Medical Image Processing and Artificial Intelligence Methods. Nuclear Instruments and Methods in Physics Research A 569, 591–595 (2006)

Classification of Medical Images Using Data Mining Techniques B.G. Prasad1 and Krishna A.N

An Intelligent Mining System for Diagnosing Medical Images Using Combined Texture-Histogram Features. K. Dhanalakshmi,1 V. Rajaman