```
% ENTROPY classificaton tree with carseats

T = readtable('Carseats_train.csv')

% need to create high variable and attach it to the table

T{T.Sales < 8, "High" } = 0
T{T.Sales >= 8, "High" } = 1


%Delete Sales and Index column

T= removevars(T,{'Sales'});
T= removevars(T,{'Var1'});

%fitting classification tree with entropy

tc = fitctree(T, "High", "MinLeafSize", 5, "MinParentSize",
 10, "MergeLeaves", "off", "SplitCriterion","deviance")

%View tree
%view(tc,'Mode','graph')
view(tc,'Mode','text')


%looks good, can save the graph
%need to see if i can get tabular output


%preduction
test = readtable('Carseats_test.csv')

test{test.Sales < 8, "High" } = 0
test{test.Sales >= 8, "High" } = 1

y_test = test.High
test= removevars(test,{'Var1'});
test= removevars(test,{'High'});
test= removevars(test,{'Sales'});


test_pred = predict(tc, test)
test_results = table(y_test, test_pred)

y_train = T.High
train= removevars(T,{'High'});

missclass = 0
for i = 1:length(y_test)
    if test_results.y_test(i) ~= test_results.test_pred(i)
        missclass = missclass +1
    end
```

```
end


%misclassifications

miss_test = []
miss_train = []
leaves = []
for i = 1:max(tc.PruneList)
    ptc = prune(tc, "level", i )
    test_pred = predict(ptc, test)
    test_results = table(y_test, test_pred)
    leaves = [leaves sum(~ptc.IsBranchNode)]
    missclass = 0
    for i = 1:length(y_test)
        if test_results.y_test(i) ~= test_results.test_pred(i)
            missclass = missclass +1
        end
    end


    miss_test = [miss_test missclass]

    train_pred = predict(ptc, train)
    train_results = table(y_train, train_pred)

    missclass = 0
    for i = 1:length(y_test)
        if train_results.y_train(i) ~= train_results.train_pred(i)
            missclass = missclass +1
        end
    end
    miss_train = [miss_train missclass]

end


figure
plot(leaves, miss_test, leaves, miss_train)
title('Carseats Classification Entropy Matlab, Leaves vs Missclassifications
 for Test and Train')
xlabel('Leaves')
ylabel('Missclassifications')
ax = gca
ax.XDir = "reverse"




ptc = prune(tc, "level", 6 ) %max level (15) - min mse level (3) +1
view(ptc,'Mode','graph')

test_pred = predict(ptc, test)
test_results = table(y_test, test_pred)
```

```
missclass = 0
for i = 1:length(y_test)
    if test_results.y_test(i) ~= test_results.test_pred(i)
        missclass = missclass +1
    end
end
missclass
```

T =

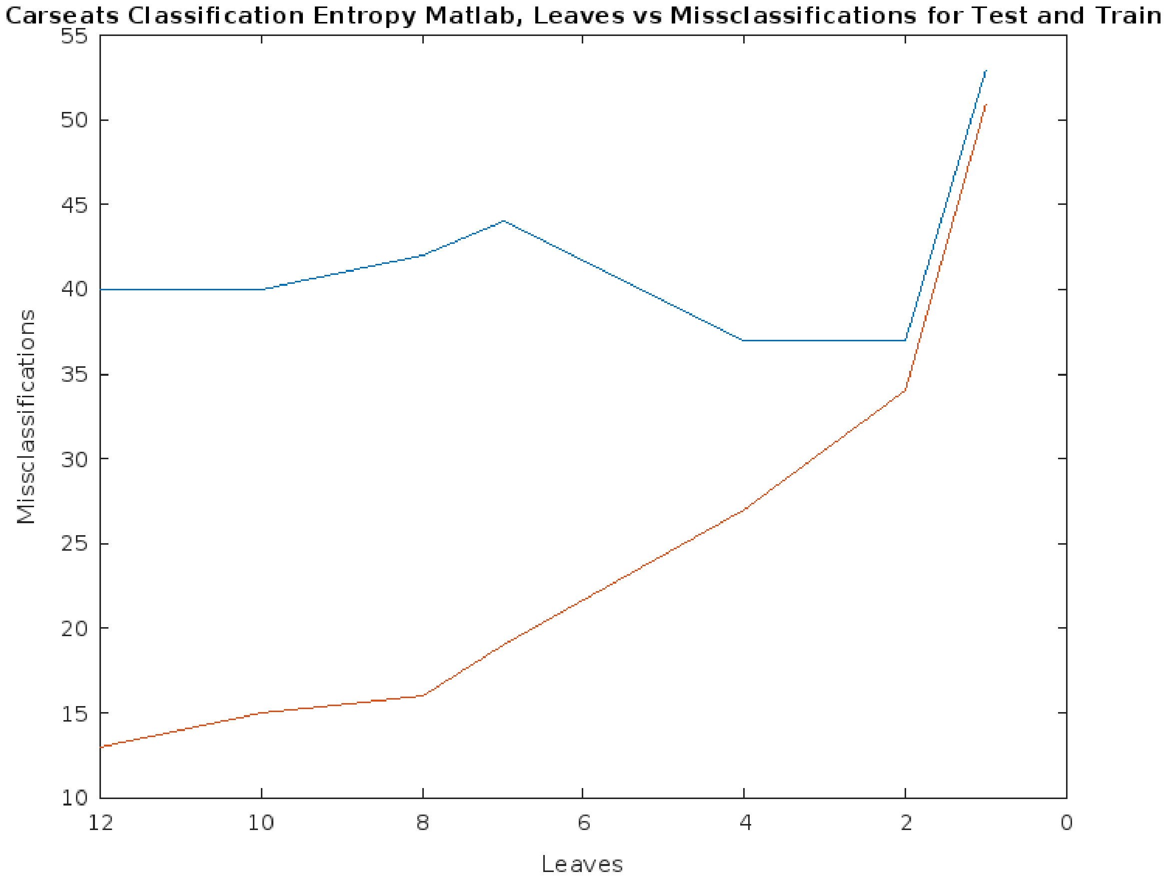
280x12 table

Var1	Sales	CompPrice	Income	Advertising	Population	Price	ShelveLoc	Age	Education	Urban	US
1	9.5	138	73	11	276	120	{'Bad' }	42	17	{'Yes'}	{'Yes'}
2	11.22	111	48	16	260	83	{'Good' }	65	10	{'Yes'}	{'Yes'}
3	10.06	113	35	10	269	80	{'Medium'}	59	12	{'Yes'}	{'Yes'}
4	7.4	117	100	4	466	97	{'Medium'}	55	14	{'Yes'}	{'Yes'}
5	4.15	141	64	3	340	128	{'Bad' }	38	13	{'Yes'}	{'No' }
6	10.81	124	113	13	501	72	{'Bad' }	78	16	{'No' }	{'Yes'}
7	6.63	115	105	0	45	108	{'Medium'}	71	15	{'Yes'}	{'No' }
8	11.85	136	81	15	425	120	{'Good' }	67	10	{'Yes'}	{'Yes'}
9	6.54	132	110	0	108	124	{'Medium'}	76	10	{'No' }	{'No' }
10	4.69	132	113	0	131	124	{'Medium'}	76	17	{'No' }	{'Yes'}
11	9.01	121	78	9	150	100	{'Bad' }	26	10	{'No' }	{'Yes'}
12	11.96	117	94	4	503	94	{'Good' }	50	13	{'Yes'}	{'Yes'}
13	3.98	122	35	2	393	136	{'Medium'}	62	18	{'Yes'}	{'No' }
14	10.96	115	28	11	29	86	{'Good' }	53	18	{'Yes'}	{'Yes'}
15	11.17	107	117	11	148	118	{'Good' }	52	18	{'Yes'}	{'Yes'}
16	8.71	149	95	5	400	144	{'Medium'}	76	18	{'No' }	{'No' }
17	7.58	118	32	0	284	110	{'Good' }	63	13	{'Yes'}	{'No' }

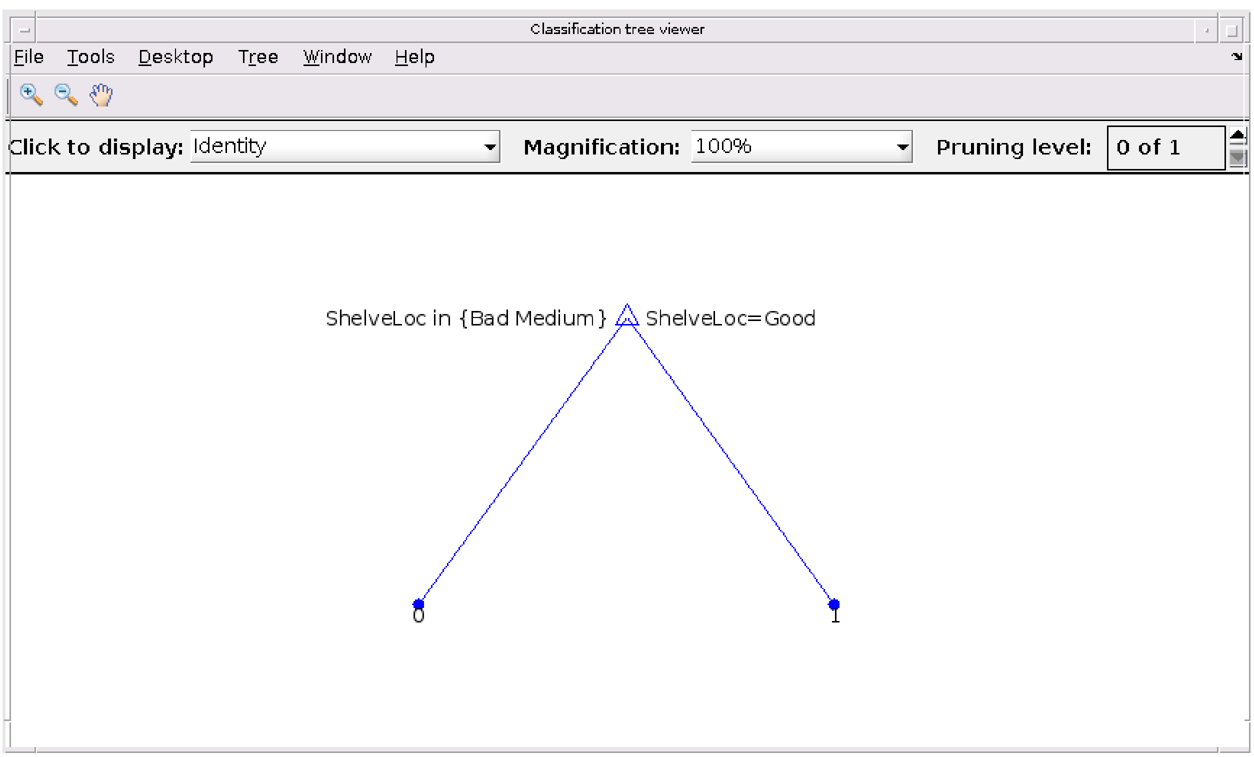
18	12.29	147	74	13	251	131	{'Good' }	52	10	{'Yes'}	{'Yes'}
19	13.91	110	110	0	408	68	{'Good' }	46	17	{'No' }	{'Yes'}
20	8.73	129	76	16	58	121	{'Medium'}	69	12	{'Yes'}	{'Yes'}
21	6.41	125	90	2	367	131	{'Medium'}	35	18	{'Yes'}	{'Yes'}
22	12.13	134	29	12	239	109	{'Good' }	62	18	{'No' }	{'Yes'}
23	5.08	128	46	6	497	138	{'Medium'}	42	13	{'Yes'}	{'No' }
24	5.87	121	31	0	292	109	{'Medium'}	79	10	{'Yes'}	{'No' }
25	10.14	145	119	16	294	113	{'Bad' }	42	12	{'Yes'}	{'Yes'}
26	14.9	139	32	0	176	82	{'Good' }	54	11	{'No' }	{'No' }
27	8.33	107	115	11	496	131	{'Good' }	50	11	{'No' }	{'Yes'}
28	5.27	98	118	0	19	107	{'Medium'}	64	17	{'Yes'}	{'No' }
29	2.99	103	74	0	359	97	{'Bad' }	55	11	{'Yes'}	{'Yes'}
30	7.81	104	99	15	226	102	{'Bad' }	58	17	{'Yes'}	{'Yes'}
31	13.55	125	94	0	447	89	{'Good' }	30	12	{'Yes'}	{'No' }
32	8.25	136	58	16	241	131	{'Medium'}	44	18	{'Yes'}	{'Yes'}
33	6.2	107	32	12	236	137	{'Good' }	64	10	{'No' }	{'Yes'}
34	8.77	114	38	13	317	128	{'Good' }	50	16	{'Yes'}	{'Yes'}
35	2.67	115	54	0	406	128	{'Medium'}	42	17	{'Yes'}	{'Yes'}
36	11.07	131	84	11	29	96	{'Medium'}	44	17	{'No' }	{'Yes'}
37	8.89	122	76	0	270	100	{'Good' }	60	18	{'No' }	{'No' }
38	4.95	121	41	5	412	110	{'Medium'}	54	10	{'Yes'}	{'Yes'}
39	6.59	109	73	0	454	102	{'Medium'}	65	15	{'Yes'}	{'No' }
40	3.24	130	60	0	144	138	{'Bad' }	38	10	{'No' }	{'No' }
41	2.07	119	98	0	18	126	{'Bad' }	73	17	{'No' }	{'No' }
42	7.96	157	53	0	403	124	{'Bad' }	58	16	{'Yes'}	{'No' }
43	10.43	77	69	0	25	24	{'Medium'}	50	18	{'Yes'}	{'No' }
44	4.12	123	42	11	16	134	{'Medium'}	59	13	{'Yes'}	{'Yes'}

45	4.16	85	79	6	325	95	{'Medium'}	69	13	{'Yes'}	{'Yes'}
46	4.56	141	63	0	168	135	{'Bad' }	44	12	{'Yes'}	{'Yes'}
47	12.44	127	90	14	16	70	{'Medium'}	48	15	{'No' }	{'Yes'}
48	4.38	126	98	0	173	108	{'Bad' }	55	16	{'Yes'}	{'No' }
49	3.91	116	52	0	349	98	{'Bad' }	69	18	{'Yes'}	{'No' }
50	10.61	157	93	0	51	149	{'Good' }	32	17	{'Yes'}	{'No' }
51	1.42	99	32	18	341	108	{'Bad' }	80	16	{'Yes'}	{'Yes'}
52	4.42	121	90	0	150	108	{'Bad' }	75	16	{'Yes'}	{'No' }
53	7.91	153	40	3	112	129	{'Bad' }	39	18	{'Yes'}	{'Yes'}
54	6.92	109	64	13	39	119	{'Medium'}	61	17	{'Yes'}	{'Yes'}
55	4.9	134	103	13	25	144	{'Medium'}	76	17	{'No' }	{'Yes'}
56	6.85	143	81	5	60	154	{'Medium'}	61	18	{'Yes'}	{'Yes'}
57	11.91	133	82	0	54	84	{'Medium'}	50	17	{'Yes'}	{'No' }
58	0.91	93	91	0	22	117	{'Bad' }	75	11	{'Yes'}	{'No' }
59	5.42	103	93	15	188	103	{'Bad' }	74	16	{'Yes'}	{'Yes'}
60	5.21	118	71	4	148	114	{'Medium'}	80	13	{'Yes'}	{'No' }
61	8.32	122	102	19	469	123	{'Bad' }	29	13	{'Yes'}	{'Yes'}
62	7.32	105	32	0	358	107	{'Medium'}	26	13	{'No' }	{'No' }
63	1.82	139	45	0	146	133	{'Bad' }	77	17	{'Yes'}	{'Yes'}
64	8.47	119	88	10	170	101	{'Medium'}	61	13	{'Yes'}	{'Yes'}
65	7.8	100	67	12	184	104	{'Medium'}	32	16	{'No' }	{'Yes'}
66	4.9	122	26	0	197	128	{'Medium'}	55	13	{'No' }	{'No' }
67	8.85	127	92	0	508	91	{'Medium'}	56	18	{'Yes'}	{'No' }
68	9.01	126	61	14	152	115	{'Medium'}	47	16	{'Yes'}	{'Yes'}
69	13.39	149	69	20	366	134	{'Good' }	60	13	{'Yes'}	{'Yes'}
70	7.99	127	59	0	339	99	{'Medium'}	65	12	{'Yes'}	{'No' }
71	9.46	89	81	15	237	99	{'Good' }	74	12	{'Yes'}	{'Yes'}

```
    72       6.5       148        51        16          148         150
   {'Medium'}    58         17        {'No' }    {'Yes'}
    73      5.52       115        45         0          432         116
   {'Medium'}    25         15        {'Yes'}    {'No' }
    74     12.61       118        90        10           54         104
   {'Good'  }    31         11        {'No' }    {'Yes'}
    75       6.2       150        68         5          125         136
   {'Medium'}    64         13        {'No' }    {'Yes'}
    76      8.55        88       111        23          480          92
   {'Bad'   }    36         16        {'No' }    {'Yes'}
    77     10.64       102        87        10          346          70
   {'Medium'}    64         15        {'Yes'}    {'Yes'}
    78       7.7       118        71        12           44          89
   {'Medium'}    67         18        {'No' }    {'Yes'}
    79      4.43       134        48         1          139         145
   {'Medium'}    65         12        {'Yes'}    {'Yes'}
    80      9.14       134        67         0          286          90
   {'Bad'   }    41         13        {'Yes'}    {'No' }
    81      8.01       113       100        16          353          79
   {'Bad'   }    68         11        {'Yes'}    {'Yes'}
    82      7.52       116        72         0          237         128
   {'Good'  }    70         13        {'Yes'}    {'No' }
    83     11.62       151        83         4          325         139
   {'Good'  }    28         17        {'Yes'}    {'Yes'}
    84      4.42       109        36         7          468          94
   {'Bad'   }    56         11        {'Yes'}    {'Yes'}
    85      2.23       111        25         0           52         121
   {'Bad'   }    43         18        {'No' }    {'No' }
    86      8.47       125       103         0          304         112
   {'Medium'}    49         13        {'No' }    {'No' }
    87       8.7       150        84         9          432         134
   {'Medium'}    64         15        {'Yes'}    {'No' }
    88      11.7       131        67         7          272         126
   {'Good'  }    54         16        {'No' }    {'Yes'}
    89      6.56       117        42         7          144         111
   {'Medium'}    62         10        {'Yes'}    {'Yes'}
    90      7.95       128        66         3          493         119
   {'Medium'}    45         16        {'No' }    {'No' }
    91      5.33       115        22         0          491         103
   {'Medium'}    64         11        {'No' }    {'No' }
    92      4.81        97        46        11          267         107
   {'Medium'}    80         15        {'Yes'}    {'Yes'}
...
```



Carseats Classification Entropy Matlab, Leaves vs Missclassifications for Test and Train
Missclassifications
Leaves
55
50
45
40
35
30
25
20
15
10
12
10
8
6
4
2
0



Classification tree viewer
File Tools Desktop Tree Window Help
Click to display: Identity
Magnification: 100%
Pruning level: 0 of 1
ShelveLoc in {Bad Medium}
ShelveLoc=Good
0
1

Published with MATLAB® R2023a