

Ankit Rathi

Github username -- ankitrathi33

1.

Params:

min_rule_covered=5.0
induction_measure=Correlation
pruning_measure=Correlation
voting_measure=Correlation

Rules:

r1: IF tear-prod-rate = {reduced} THEN class = {none} (p=8.0, n=0.0, P=11.0, N=7.0, weight=0.7135060680126758, pval=0.0037707390648567124)

Coverage of training examples by rules (1-based):

-,1*,-,1*,1*,-,1*,-,-,1*,-,-,1*,1*,-,1*,-

Params:

min_rule_covered=5.0
induction_measure=RSS
pruning_measure=RSS
voting_measure=RSS

Rules:

r1: IF tear-prod-rate = {reduced} THEN class = {none} (p=8.0, n=0.0, P=11.0, N=7.0, weight=0.7272727272727273, pval=0.0037707390648567124)

Coverage of training examples by rules (1-based):

-,1*,-,1*,1*,-,1*,-,-,1*,-,-,1*,1*,-,1*,-

Params:

min_rule_covered=5.0
induction_measure=C2
pruning_measure=C2
voting_measure=C2

Rules:

r1: IF tear-prod-rate = {reduced} THEN class = {none} (p=8.0, n=0.0, P=11.0, N=7.0, weight=0.8636363636363636, pval=0.0037707390648567124)

Coverage of training examples by rules (1-based):

-,1*,-,1*,1*,-,1*,-,1*,-,1*,1*,-,1*,-

2.

Measures	Numbers of rules	Training accuracy	Test accuracy
Correlation	1	61.11	67.67
c2	1	61.11	67.67
rss	1	61.11	67.67

3. In my given dataset there is only one rule for all three model and accuracy of every model remain same.

4.

JRIP rules:

=====

(age = young) and (astigmatism = yes) and (tear-prod-rate = normal) => class=hard (2.0/0.0)

(astigmatism = no) and (tear-prod-rate = normal) => class=soft (5.0/1.0)

=> class=none (11.0/1.0)

Number of Rules : 3

accuracy: 83.33%

In Rulekit ther was only 1 rule present for my data so accuracy was low but for jrip the number of rules present is 3 thus the accuracy changes gradually.

5.

Params:

min_rule_covered=5.0

induction_measure=C2

pruning_measure=C2

voting_measure=C2

Rules:

r1: IF THORAX = <31.5, 35.5) THEN class = {31.0} [27.50142886309282,34.49857113690718]
(p=4.0, n=1.0, P=13.0, N=75.0, weight=0.5004102564102565, pval=0.003337850915959032)
r2: IF THORAX = (-inf, 35.5) THEN class = {32.0} [21.7019619938136,42.2980380061864]
(p=8.0, n=3.0, P=35.0, N=53.0, weight=0.3361185983827493, pval=0.037859423634075774)
r3: IF SLEEP = <0.81, inf) AND THORAX = <69.0, 71.0) THEN class = {23.0}
[16.98843706770952,29.01156293229048] (p=5.0, n=1.0, P=33.0, N=55.0,
weight=0.4222222222222222, pval=0.01293825983793209)
r4: IF THORAX = <51.0, 67.5) AND TYPE = {0} THEN class = {26.0}
[6.96439739155425,45.03560260844575] (p=11.0, n=1.0, P=70.0, N=18.0,
weight=0.34285714285714297, pval=0.7487855353734036)
r5: IF SLEEP = <0.81, inf) AND THORAX = <60.5, 69.0) THEN class = {12.0}
[5.936549489047889,18.063450510952112] (p=7.0, n=2.0, P=35.0, N=53.0,
weight=0.37861635220125783, pval=0.0018120647530237666)
r6: IF SLEEP = <0.83, inf) AND THORAX = <57.0, inf) THEN class = {14.0}
[1.9760103738146793,26.02398962618532] (p=28.0, n=7.0, P=57.0, N=31.0,
weight=0.32229767968307876, pval=0.006159030300510378)
r7: IF SLEEP = (-inf, 0.89) AND THORAX = <46.5, inf) AND TYPE = {1} THEN class = {21.0}
[4.539627053100517,37.46037294689948] (p=13.0, n=1.0, P=72.0, N=16.0,
weight=0.3583829365079363, pval=0.4796161333251511)
r8: IF SLEEP = <0.81, inf) AND THORAX = <37.0, 71.0) THEN class = {13.0}
[-1.172167138295256,27.172167138295258] (p=28.0, n=5.0, P=58.0, N=30.0,
weight=0.41187739463601547, pval=0.09417767516930091)
r9: IF SLEEP = <0.79, 0.81) AND THORAX = <31.5, inf) THEN class = {28.0}
[11.65711163839146,44.342888361608544] (p=8.0, n=2.0, P=59.0, N=29.0,
weight=0.22320280537697265, pval=0.49084416469655895)
r10: IF SLEEP = <0.75, inf) AND THORAX = (-inf, 46.5) THEN class = {42.0}
[22.835190039335053,61.16480996066495] (p=7.0, n=6.0, P=34.0, N=54.0,
weight=0.1494469582704877, pval=0.7654252446722116)
r11: IF SLEEP = (-inf, 0.74) AND THORAX = <38.0, 51.0) THEN class = {62.0}
[40.00545522180556,83.99454477819444] (p=2.0, n=3.0, P=10.0, N=78.0,
weight=0.19384615384615386, pval=0.8438921773884462)

Coverage of training examples by rules (1-based):

1*,2*,-;11*,11*,10*,10*,4*,4*,4*,9*,4*,9*,9*,4,5,6,8*,4,5,6,8*,3*,6,8;6*,6*,3*,6,8;3*,6,8;6*,6*,11*,-,9*
;5,6,8*,6*,6*,6,8*,6,8*,5,6,8*,6*,5,6,8*,6*,6*,6*,4*,10*,4*,9*,10;9*,4*,9,-;4,5,6,8*,4,5,6,8*,3*,6,8;6*
;8*,10;4,8*,3*,6,8;3*,6,8;6*,6*,2*,11*,11*,7*,10*,9*,10;7*,9;7*,9;7,8*,7,8*,6,7,8*,6,7,8*,5,6,7,8*,6,
7,8*,6,7*,8*,8*,8*,6*,8*,5,6,8*,2*,2*,2*,1*,2,1*,2,2*,10;10*,1*,2,10;2*,10;8*,10;7,8*,1*,2,10;7,8*,7,
8*,7,8*

Params:

min_rule_covered=5.0
induction_measure=Correlation
pruning_measure=Correlation
voting_measure=Correlation

Rules:

r1: IF THORAX = <31.5, 35.5) THEN class = {31.0} [27.50142886309282,34.49857113690718]
(p=4.0, n=1.0, P=13.0, N=75.0, weight=0.4511859223029893, pval=0.003337850915959032)
r2: IF THORAX = (-inf, 33.5) AND PARTNERS = {8} THEN class = {34.0}
[22.096744142882585,45.903255857117415] (p=5.0, n=3.0, P=31.0, N=57.0,
weight=0.18054811848918012, pval=0.1547109868120587)
r3: IF THORAX = <47.5, 51.0) THEN class = {66.0} [46.93286713856654,85.06713286143346]
(p=3.0, n=3.0, P=9.0, N=79.0, weight=0.35506015880885067, pval=0.7163245009656751)
r4: IF SLEEP = (-inf, 0.78) AND THORAX = <36.0, 47.5) THEN class = {83.0}
[57.46649260285615,108.53350739714385] (p=2.0, n=8.0, P=6.0, N=82.0,
weight=0.18725272068796442, pval=0.9829215488311097)
r5: IF SLEEP = <0.78, 0.86) AND THORAX = (-inf, 47.5) THEN class = {32.0}
[20.02460856589648,43.97539143410352] (p=6.0, n=4.0, P=36.0, N=52.0,
weight=0.13903017140231752, pval=0.1190761793775627)
r6: IF SLEEP = <0.81, inf) AND THORAX = <64.0, 71.0) THEN class = {18.0}
[12.467369522550777,23.532630477449224] (p=7.0, n=3.0, P=29.0, N=59.0,
weight=0.2821923875156103, pval=4.4683632594360115E-4)
r7: IF SLEEP = <0.81, 0.9) AND THORAX = <59.5, 62.5) THEN class = {5.0}
[-0.3065996645686395,10.30659966456864] (p=4.0, n=1.0, P=20.0, N=68.0,
weight=0.33543396688095134, pval=0.016439910748649218)
r8: IF SLEEP = <0.81, inf) AND THORAX = <59.5, 78.0) THEN class = {14.0}
[1.8544331997469836,26.145566800253015] (p=22.0, n=4.0, P=57.0, N=31.0,
weight=0.2690017215324781, pval=0.019566217793350254)
r9: IF SLEEP = <0.83, 0.9299999999999999) AND THORAX = <62.5, 85.0) THEN class =
{23.0} [11.541136375907271,34.45886362409273] (p=19.0, n=5.0, P=51.0, N=37.0,
weight=0.26314580806948046, pval=0.011216149264208842)
r10: IF SLEEP = <0.81, 0.9299999999999999) AND THORAX = <53.5, inf) THEN class = {14.0}
[1.9953438794229914,26.00465612057701] (p=32.0, n=7.0, P=57.0, N=31.0,
weight=0.3227052828494027, pval=0.0038295057213001216)
r11: IF SLEEP = <0.79, 0.86) AND THORAX = <55.0, inf) THEN class = {10.0}
[0.6209306352516908,19.37906936474831] (p=14.0, n=8.0, P=40.0, N=48.0,
weight=0.21081851067789195, pval=8.801010436381291E-4)
r12: IF SLEEP = (-inf, 0.89) AND THORAX = <45.0, 67.5) THEN class = {66.0}
[46.917272867875425,85.08272713212457] (p=5.0, n=32.0, P=9.0, N=79.0,
weight=0.09237682410924102, pval=0.8553535171141503)

Coverage of training examples by rules (1-based):

1*;4*;3*,12;4*,12;4*;4*,12;12*;12*;11*,12;11*,12;11*;8,9,10*,11,12;6,8,9,10*,11,12;6,8,9,10*,11;
8,9,10*,11;9,10*,11;6,8,9,10*;6,8,9,10*;8,9,10*;9,10*;4*,4*,12;-;5*,12;7*,8,10,11,12;8,9,10*,11;1
0*,11;10*,12;10*,12;7*,8,10,12;9,10*;8,10*;8,9,10*;10*;8*;12*;4*,12;12*;5*;3*,12;11*,12;10*,11;8
,9,10*,11,12;6,8,9,10*,11,12;6,8,9,10*,11;8,9,10*,11;12*;10*,12;6,8,9,10*;6,8,9,10*;8,9,10*;8,9,1
0*;-;4*;4*;12*;5*;5*;11*,12;11*,12;3*,12;12*;7*,8,10,11,12;7*,8,10,11,12;6,8,9,10*,11,12;7*,8,10,
12;9,10*;3*;3*;10*;9,10*;3*;6,8,9,10*;2*;2*;2*;1*,2;1*,2;2*;4*;1*,2,5;2*,5;5*;10*,12;1*,5;5*,12;5*,1
2;10*,12

Params:

min_rule_covered=5.0
induction_measure=RSS
pruning_measure=RSS
voting_measure=RSS

Rules:

r1: IF SLEEP = (-inf, 0.91) AND THORAX = <38.0, 53.5) AND PARTNERS = {1} THEN class =
{62.0} [41.30624719973012,82.69375280026988] (p=5.0, n=6.0, P=9.0, N=79.0,
weight=0.47960618846694797, pval=0.8565037367387689)
r2: IF SLEEP = <0.78, inf) AND THORAX = <47.0, 78.0) AND TYPE = {0} AND PARTNERS =
{1} THEN class = {66.0} [47.90931932533418,84.09068067466582] (p=2.0, n=9.0, P=9.0,
N=79.0, weight=0.10829817158931082, pval=0.6602879521508763)
r3: IF SLEEP = <0.78, 0.89) AND THORAX = <50.5, 64.0) AND TYPE = {1} AND PARTNERS =
{1} THEN class = {73.0} [50.76826592458429,95.23173407541572] (p=1.0, n=5.0, P=6.0,
N=82.0, weight=0.1056910569105691, pval=0.8683809604572656)
r4: IF THORAX = <53.5, 71.0) THEN class = {15.0}
[0.9231880099765064,29.076811990023494] (p=31.0, n=5.0, P=65.0, N=23.0,
weight=0.25953177257525084, pval=0.07587871220501666)
r5: IF SLEEP = <0.77, inf) THEN class = {21.0} [6.568462067077709,35.43153793292229]
(p=49.0, n=15.0, P=64.0, N=24.0, weight=0.140625, pval=0.04238611689419267)
r6: IF SLEEP = (-inf, 0.78) AND THORAX = <36.0, inf) AND TYPE = {0} AND PARTNERS = {8}
THEN class = {81.0} [48.80924204124446,113.19075795875554] (p=3.0, n=4.0, P=9.0, N=79.0,
weight=0.28270042194092826, pval=0.9984056755174)
r7: IF SLEEP = <0.66, 0.77) AND THORAX = <31.5, 47.5) THEN class = {83.0}
[59.67455490480645,106.32544509519354] (p=2.0, n=10.0, P=6.0, N=82.0,
weight=0.2113821138211382, pval=0.9615428792762628)
r8: IF THORAX = (-inf, 35.5) THEN class = {32.0} [21.7019619938136,42.2980380061864]
(p=8.0, n=3.0, P=35.0, N=53.0, weight=0.1719676549865229, pval=0.037859423634075774)

Coverage of training examples by rules (1-based):

8*;6*,7;6*;6*,7;6*,7;4,6*;4,6*;4*,5;4*,5;4*,5;4*,5;5*;5*;4*,5;4*,5;5*;5*;7*;7*;4*,5;5*;4*,5;5*;5*;4*,5;4*,5;4*,5;5*;4*,5;5*;5*;4*;1*,7;4*;1*,5;1*,2,5;2,4*,5;5*;2,4*,5;2,4*,5;2,4*,5;2,5*;1*,5;2,4*,5;2,4*,5;2,4*,5;2,5*;2,5*;8*;1*,7;1*,7;4*;5*;1*,5;3,4*,5;3,4*,5;1*,5;1*,3,5;3,4*,5;3,4*,5;4*,5;3,4*,5;5*;1*,5;1*,5;4*,5;5*;5*;4*,5;8*;8*;8*;7*,8;7*,8;8*;7*;5,8*;5,8*;5*;4*,5;5,8*;5*;5*;4*,5

6.

measures	Number of rules	Training rule_relative_squared _error	Test rule_relative_squared _error
correlation	12	1.663	2.797
c2	11	1.146	1.857
rss	8	1.461	2.495

7.

The **root relative squared error** is relative to what it would have been if a simple predictor had been used. More specifically, this simple predictor is just the average of the actual values. Thus, the relative squared error takes the total squared error and normalizes it by dividing by the total squared error of the simple predictor. By taking the square root of the relative squared error one reduces the error to the same dimensions as the quantity being predicted [1].

$$E_i = \sqrt{\frac{\sum_{j=1}^n (P_{(i)} - T_j)^2}{\sum_{j=1}^n (T_j - \bar{T})^2}}$$

The main difference between rule based model generated by c2, correlation and rss is that they took different number of rules to generate rule_relative_squared_error, and training and test data rule_relative_squared_error changes for every model subsequently.

8.

M5 pruned model tree:

(using smoothed linear models)

LM1 (88/100%)

LM num: 1

class =

+ 24.1477

Number of Rules : 1

In Rulekit there was only more than 5 rules present for my data but for m5 the number of rules present is 1.