Pot-luck challenge: TIED

Advanced Analytics, Intel, LTD EUGENE.TUV@INTEL.COM
5000 W. Chandler Blvd
CH5-295
Chandler, AZ, 85226, USA

Task(s) solved:

- Using training data, find all minimal sets of features with optimal predictivity
- For each of the feature set identified, build a classifier model of the target variable using training data and apply it to the testing data.

Method:Rule induction on relevant features
Feature selection method (ACE - Artificial Contrasts with Ensembles) was used to remove irrelevant features. Two rule induction techniques were used to find sets of features with optimal predictability: CART with surrogate splits and a supervised APRIORI. Both point to the same optimal sets of features.

- Feature selection: ACE is a combination of three ideas: A) Estimating variable importance using RF ensemble of trees of a fixed depth (3-6 levels) with the split weight re-estimation on OOB samples (gives more accurate and unbiased estimate of variable importance in each tree), B) comparing variable importance against artificially constructed noise variables using a formal statistical test, and C) Iteratively removing the effect of identified important variables to allow detection of less important variables. ACE method is outlined in (Tuv et al., 2006). The more comprehensive paper is submitted to JMLR (currently under review).
  The results of ACE applied to the TIED dataset are shown on the Figure 1. The algorithm stopped after 3 iterations (no new relevant features found), and the resulting set of selected relevant (strongly and weakly) features is shown in the last column.
- Classification tree (Breiman et al., 1984) built on selected features shown on Figure1. Optimal tree has four terminal nodes, and gave CV BER ~ 0.02. The tree was used for the prediction on the test data. Figure 2 presents surrogate scores tables shown for each of the three splits. Note that for the first split on Column10 there are three surrogates with equivalent splits (Column1/2/3). Similarly for the second and the third splits equivalent splits are achieved by using Column11/12/13 and Column18/19/20 correspondingly.
- Supervised Apriori: we customized Apriori (Agrawal et al., 1993) algorithm to produce rules with known consequent - specific class of a categorical target. We use conditional support (fraction of the data from the specified class covered by the rule)

to dramatically simplify APRIORI rule tree construction. As a preprocessing step numeric predictors are discretized, and levels of categorical predictors are optionally clustered with respect to the target class using decision tree with MDL based pruning. The preprocessing is done on each variable independently, and could result in suboptimal rules (this is the case for the target class=2, TIED). The set of the best rules found by the algorithm is shown on Figures 3-4, and involve the same set of variables $\{1, 2, 3, 10\} \times \{11, 12, 13\} \times \{18, 19, 20\}$ found by a single tree (with surrogate splits).

Implementation: All the methods described above are implemented in C++ within Intel Statistical Learning framework - IDEAL. It is not publicly available.
Results:

- Minimal sets of features with optimal predictivity: 36 sets of vars $\longrightarrow \{1, 2, 3, 10\} \times \{11, 12, 13\} \times \{18, 19, 20\}$
- Model: Single 4-node classification tree built using any triple from the above cartesian product (see Figure 1) results in the equivalent model with CV BER ~ 0.02

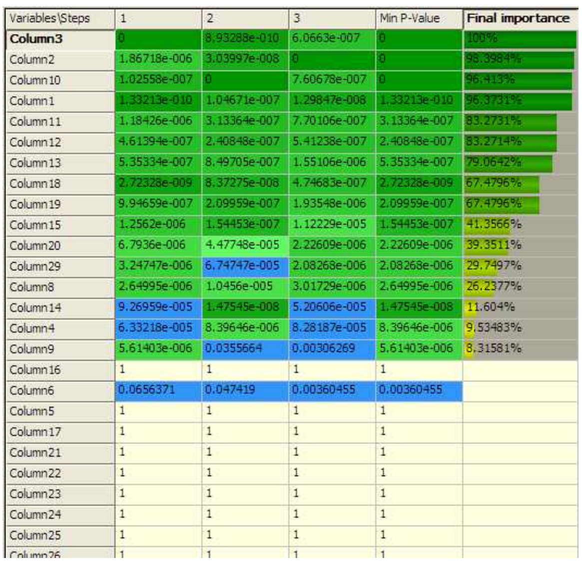
Keywords: feature selection, tree classifier, rule induction, supervised Apriori

References

R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington D.C., May 1993.

L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Wadsworth, Belmont, MA, 1984.

E. Tuv, A. Borisov, and K. Torkkola. Feature selection using ensemble based ranking against artificial contrasts. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2006.



Variables\Steps	1	2	3	Min P-Value	Final importance
Column3	0	8.93288e-010	6.0663e-007	0	100%
Column2	1.86718e-006	3.03997e-008	0	0	98.3984%
Column10	1.02558e-007	0	7.60678e-007	0	96.413%
Column1	1.33213e-010	1.04671e-007	1.29847e-008	1.33213e-010	96.3731%
Column11	1.18426e-006	3.13364e-007	7.70106e-007	3.13364e-007	83.2731%
Column12	4.61394e-007	2.40848e-007	5.41238e-007	2.40848e-007	83.2714%
Column13	5.35334e-007	8.49705e-007	1.55106e-006	5.35334e-007	79.0642%
Column18	2.72328e-009	8.37275e-008	4.74683e-007	2.72328e-009	67.4796%
Column19	9.94659e-007	2.09959e-007	1.93548e-006	2.09959e-007	67.4796%
Column15	1.2562e-006	1.54453e-007	1.12229e-005	1.54453e-007	41.3566%
Column20	6.7936e-006	4.47748e-005	2.22609e-006	2.22609e-006	39.3511%
Column29	3.24747e-006	6.74747e-005	2.08268e-006	2.08268e-006	29.7497%
Column8	2.64995e-006	1.0456e-005	3.01729e-006	2.64995e-006	26.2377%
Column14	9.26959e-005	1.47545e-008	5.20606e-005	1.47545e-008	11.604%
Column4	6.33218e-005	8.39646e-006	8.28187e-005	8.39646e-006	9.53483%
Column9	5.61403e-006	0.0355664	0.00306269	5.61403e-006	8.31581%
Column16	1	1	1	1	
Column6	0.0656371	0.047419	0.00360455	0.00360455	
Column5	1	1	1	1	
Column17	1	1	1	1	
Column21	1	1	1	1	
Column22	1	1	1	1	
Column23	1	1	1	1	
Column24	1	1	1	1	
Column25	1	1	1	1	
Column26	1	1	1	1	

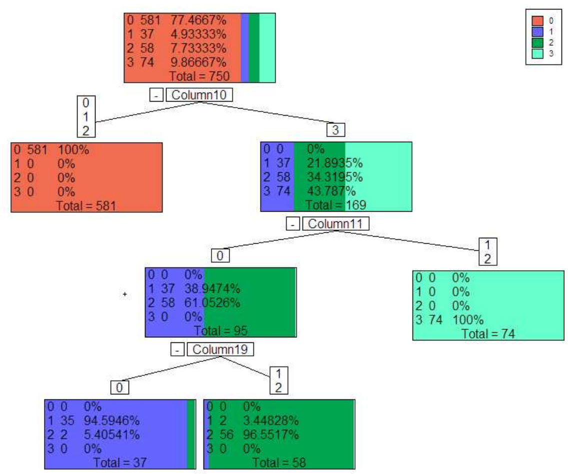


Figure 1: Left graph: The results of ACE applied to the TIED dataset.The algorithm stopped after 3 iterations (no new relevant features found), and the resulting set of selected relevant (strongly and weakly) features sorted by relative importance is shown in the last column. Right Graph: Classification tree built on the set of the relevant features identified by ACE. For each split surrogate scores are calculated for each variable (see the Figure 2)

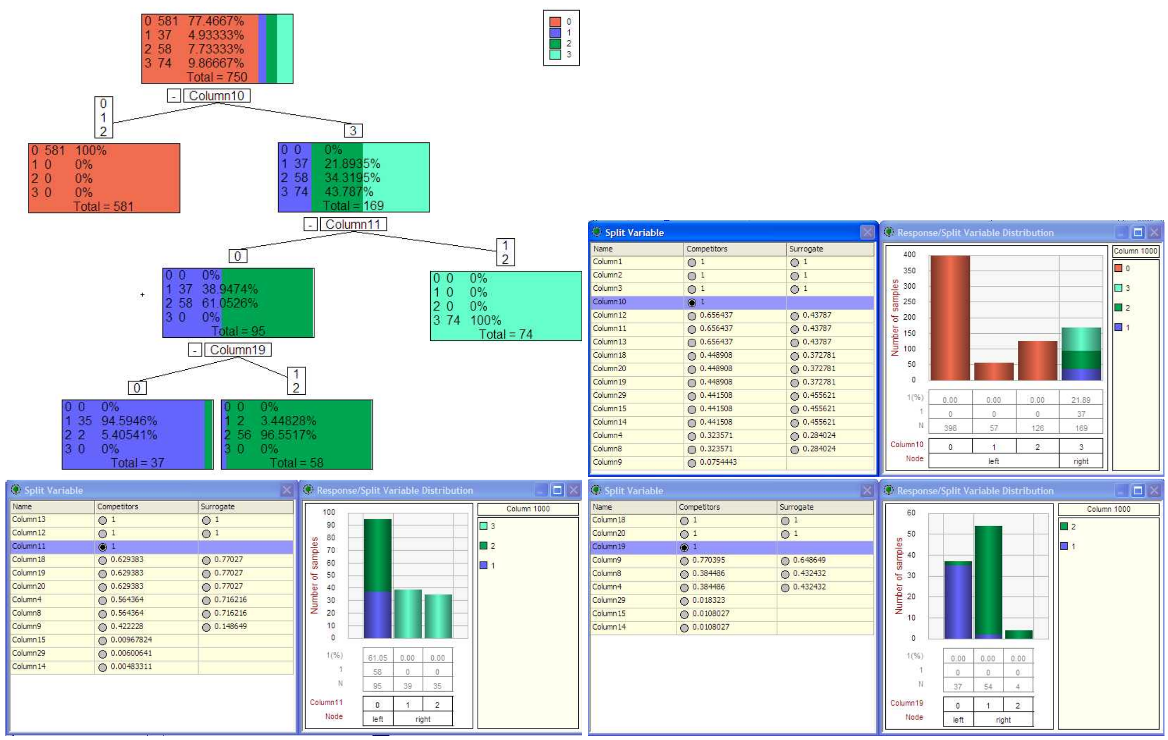


Figure 2: Surrogate scores tables shown for each of three splits for the tree model built to classify TIED target. Note that for the first split on Column10 there are three surrogates with equivalent splits (Column1/2/3). Similarly for the second and the third splits equivalent splits are achieved by using Column11/12/13 and Column18/19/20 correspondingly.

	Itemsets	Length	Confidence	Cond. Support	Support	Rank
☑	Column10 IN (0, 1, 2)	1	1	581 (100%)	581 (77.47%)	0
☑	Column3 IN (0, 1, 2)	1	1	581 (100%)	581 (77.47%)	1
☑	Column2 IN (0, 1, 2)	1	1	581 (100%)	581 (77.47%)	2
☑	Column1 IN (0, 1, 2)	1	1	581 (100%)	581 (77.47%)	3
☐	Column14 IN (0, 1)	1	0.934	526 (90.53%)	563 (75.07%)	4
☐	Column29 = 0	1	0.934	526 (90.53%)	563 (75.07%)	5
☐	Column15 = 0	1	0.934	526 (90.53%)	563 (75.07%)	6

	Itemsets	Length	Confidence	Cond. Support	Support	Rank
☑	Column11 IN (1, 2)	1	1	74 (100%)	74 (9.87%)	0
☑	Column12 IN (1, 2)	1	1	74 (100%)	74 (9.87%)	1
☑	Column13 IN (1, 2)	1	1	74 (100%)	74 (9.87%)	2
☐	Column3 = 3 AND Column18 = 1	2	0.938	61 (82.43%)	65 (8.67%)	3
☐	Column2 = 3 AND Column18 = 1	2	0.938	61 (82.43%)	65 (8.67%)	4
☐	Column1 = 3 AND Column19 = 2	2	0.938	61 (82.43%)	65 (8.67%)	5

Figure 3: Rules for the target class=0 (upper table). Perfect discrimination is achieved with one of the variables 1/2/3/10. Rules for the target class=3 (lower table). Perfect discrimination is achieved with one of the variables 11/12/13.

	Itemsets	Length	Confidence	Cond. Support	Support	Rank
☑	Column13 = 0 AND Column10 = 3 AND Column19 = 0	3	0.946	35 (94.59%)	37 (4.93%)	2
☑	Column13 = 0 AND Column10 = 3 AND Column20 = 0	3	0.946	35 (94.59%)	37 (4.93%)	3
☑	Column13 = 0 AND Column1 = 3 AND Column18 = 2	3	0.946	35 (94.59%)	37 (4.93%)	4
☑	Column11 = 0 AND Column3 = 3 AND Column19 = 0	3	0.946	35 (94.59%)	37 (4.93%)	5
☑	Column13 = 0 AND Column1 = 3 AND Column20 = 0	3	0.946	35 (94.59%)	37 (4.93%)	6
☑	Column11 = 0 AND Column3 = 3 AND Column20 = 0	3	0.946	35 (94.59%)	37 (4.93%)	7
☑	Column12 = 0 AND Column1 = 3 AND Column19 = 0	3	0.946	35 (94.59%)	37 (4.93%)	8

	Itemsets	Length	Confidence	Cond. Support	Support	Rank
☑	Column13 = 0 AND Column20 = 1	2	0.963	52 (89.66%)	54 (7.2%)	3
☑	Column12 = 0 AND Column20 = 1	2	0.963	52 (89.66%)	54 (7.2%)	4
☑	Column13 = 0 AND Column18 = 0	2	0.963	52 (89.66%)	54 (7.2%)	5
☑	Column13 = 0 AND Column19 = 1	2	0.963	52 (89.66%)	54 (7.2%)	6
☑	Column11 = 0 AND Column20 = 1	2	0.963	52 (89.66%)	54 (7.2%)	7
☑	Column11 = 0 AND Column18 = 0	2	0.963	52 (89.66%)	54 (7.2%)	8
☐	Column18 = 0 AND Column8 = 2	2	0.976	40 (68.97%)	41 (5.47%)	9

Figure 4: Rules for the target class=1 (upper table, a subset is shown). The best 36 equivalent rules found by the algorithm involve triples from the set $\{1, 2, 3, 10\} \times \{11, 12, 13\} \times \{18, 19, 20\}$. Rules for the target class=2 (lower table). The best 9 equivalent rules found by the algorithm involve tuples from the set $\{11, 12, 13\} \times \{18, 19, 20\}$.