TRL Codebook

The working directory should contain the TRL runnable JAR file **TRL.jar**.

We can call the java interpreter with the command **java**. We call the TRL tool using the *-jar* flag.

TRL takes in three major flags-

- learner parameters (*-lp*),
- pre-processing parameters (*-ppp*), and
- data parameters (*-dp*).

Each flag can accept several sub-flags, the complete list can be found in the tool manual. We show one such example here.

Data We use two example datasets, in the task of brain cancer classification, to demonstrate TRL-

Source dataset The source dataset is a gene-expression data obtained from the HG-U133 Plus 2 platform from Gene Expression Omnibus ID GSE16011 (Gravendeel et al., 2009).

Target dataset The target dataset is also a gene-expression data obtained from the HG-U133A platform from Gene Expression Omnibus ID GSE1993 (Petalidis et al., 2008).

Learner parameters (*-lp*): The learner we choose is RL as baseline, compared to TRL with structure transfer (*-tr 2*). We perform stratified 10-fold cross-validation using option (*-cv 10*). Each training fold is discretized using the EBD method for discretization, setting the lambda parameter to 0.5 (*-d 9 0.5*).

Pre-processing parameters (*-ppp*): We do not pre-process the data.

Data parameters (*-dp*): The target dataset is the last argument (*Target.txt*). The source data for transfer learning is set using (*-src Source.txt*). The output directory is set using (*-od*).

Baseline with RL

```
java -jar TRL.jar -lp -cv 10 -d 9 0.5 -ppp -dp -odir demo-baseline/ Target.txt
```

```
## Rule Learner version 2010-05-29
## Args: -lp -cv 10 -d 9 0.5 -ppp -dp -odir demo-baseline/ Target.txt
## Starting: 2015-07-14 20:41:15
## On machine JEYAs-MacBook-Pro.local
## With max memory 2047m(2147280773)
## Lines: 59. Separator is '	'. Parsing........... 6021 attributes, 58 instances.
## Writing training file...
## Doing 10-fold cross-validation...
## Fold 1. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 720 attributes with
## Learned 10 rules
```

```
## Fold 2. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 863 attributes with 
## Learned 10 rules
## Fold 3. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 906 attributes with 
## Learned 10 rules
## Fold 4. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 1271 attributes with
## Learned 8 rules
## Fold 5. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 962 attributes with 
## Learned 12 rules
## Fold 6. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 884 attributes with 
## Learned 10 rules
## Fold 7. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 958 attributes with 
## Learned 9 rules
## Fold 8. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 874 attributes with 
## Learned 8 rules
## Fold 9. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 947 attributes with 
## Learned 10 rules
## Fold 10. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 956 attributes with
## Learned 10 rules
##
## === Learner parameters ===
##
## Training data: Target.txt
## Discretization method: EBD using lambda of 0.5 (2009)
## Learning method: Rule Learner
## CF function: PPV (1)
## Min CF: 0.85
## Min conjuncts: 1
## Max conjuncts: 5
## Min coverage: 4.0
## Max FP coverage: 0.1
## Min TP coverage: 0.05
## Inductive strengthening: 1
## Inference type: Weighted voting (0)
## Beam width: 1000
## Specialize satisfactory rules: false
## Validation method: 10-fold cross-validation with seed 1
##
## === Stratified cross-validation performance ===
##
## Accuracy ignoring abst.:                85.96 %
## Accuracy including abst.:               84.48 %
## Abstentions:                            1 (1.72 %)
## Relative classifier information:        37.32 +/- 10.85
##
## Class  Sensitivity(%)           Specificity(%)           Balanced Accuracy(%)     AUROC
## ------ ------------------------ ------------------------ ------------------------ ------------------
## 1      92.5(3.82)               75(11.18)                83.75(6.19)              0.83(0.06)
## 0      75(11.18)                92.5(3.82)               83.75(6.19)              0.83(0.06)
##
## Confusion matrix:
##  c1  c2  Abstentions
##  35   3   1  | c1  =  1
##   5  14   0  | c2  =  0
##
```

```
##
## Learning on whole training data...
## Discretizing using EBD using lambda of 0.5 (2009) .......... There are 1000 attributes with cut poi
## Learned 11 rules
##
## === Learner parameters ===
##
## Training data: Target.txt
## Discretization method: EBD using lambda of 0.5 (2009)
## Learning method: Rule Learner
## CF function: PPV (1)
## Min CF: 0.85
## Min conjuncts: 1
## Max conjuncts: 5
## Min coverage: 4.0
## Max FP coverage: 0.1
## Min TP coverage: 0.05
## Inductive strengthening: 1
## Inference type: Weighted voting (0)
## Beam width: 1000
## Specialize satisfactory rules: false
## Validation method: 10-fold cross-validation with seed 1
##
## === Rules (11) ===
##
## 0. ((KLRC3 = -inf..3.986)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=27, FP=0, Pos=39, Neg=19, TP_in_model=27, FP_in_model=0
##
## 1. ((ZFPL1 = 6.737..7.026)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=24, FP=0, Pos=39, Neg=19, TP_in_model=24, FP_in_model=0
##
## 2. ((ATP9A = -inf..8.151)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=23, FP=0, Pos=39, Neg=19, TP_in_model=23, FP_in_model=0
##
## 3. ((GABRB3 = -inf..4.037)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=23, FP=0, Pos=39, Neg=19, TP_in_model=23, FP_in_model=0
##
## 4. ((NR1D2 = -inf..4.991)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=21, FP=0, Pos=39, Neg=19, TP_in_model=21, FP_in_model=0
##
## 5. ((COL5A2 = -inf..5.552)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=14, FP=0, Pos=19, Neg=39, TP_in_model=14, FP_in_model=0
##
## 6. ((COL4A2 = -inf..8.226)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=13, FP=0, Pos=19, Neg=39, TP_in_model=13, FP_in_model=0
##
## 7. ((P4HB = -inf..9.790)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=12, FP=0, Pos=19, Neg=39, TP_in_model=12, FP_in_model=0
##
## 8. ((SMC4 = -inf..6.228)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=12, FP=0, Pos=19, Neg=39, TP_in_model=12, FP_in_model=0
##
## 9. ((USH1C = 6.769..inf)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=12, FP=0, Pos=19, Neg=39, TP_in_model=12, FP_in_model=0
```

```
##
## 10. ((TTC35 = 6.738..inf)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=11, FP=0, Pos=19, Neg=39, TP_in_model=11, FP_in_model=0
##
## Attributes used (11):
## KLRC3, GABRB3, NR1D2, ZFPL1, P4HB, COL4A2, TTC35, COL5A2, SMC4, ATP9A, USH1C
##
## === Classification performance on training data ===
##
## Accuracy ignoring abst.:                100 %
## Accuracy inclding abst.:                100 %
## Abstentions:                            0 (0 %)
## Relative classifier information:        100
##
## Class  Sensitivity(%) Specificity(%) Balanced Accuracy(%) AUROC
## ------ -------------- -------------- -------------------- --------------
## 1      100            100            100                  1
## 0      100            100            100                  1
##
## Confusion matrix:
##  c1  c2  Abstentions
##  39   0   0  | c1  =  1
##   0  19   0  | c2  =  0
##
##
## Total running time: 40 s.
```

The output shows the input arguments followed by some description of the discretization performed in each fold of cross-validation. The next section (Learner parameters) shows the parameters used for learning. The next section (Stratified cross-validation performance) shows the performance on cross-validation. We see a mean AUC of **0.83** with **0.06** standard error of mean (SEM). The accuracy with abstentions is **84.48%**.

The rule model obtained by learning on the entire target data is shown under (Rules). We see 11 rules here (counts start with 0). This model includes 11 attributes: *KLRC3, GABRB3, NR1D2, ZFPL1, P4HB, COL4A2, TTC35, COL5A2, SMC4, ATP9A, USH1C*. The last section shows the performance of this model on the overall target data (Classification performance on training data).

Transfer rule learning

We perform structure transfer as follows-

```
java -jar TRL.jar -lp -tr 2 -cv 10 -d 9 0.5 -ppp -dp -od demo-transfer/ -src  Source.txt Target.txt
```

```
## Rule Learner version 2010-05-29
## Args: -lp -tr 2 -cv 10 -d 9 0.5 -ppp -dp -od demo-transfer/ -src Source.txt Target.txt
## Starting: 2015-07-14 20:41:56
## On machine JEYAs-MacBook-Pro.local
## With max memory 2047m(2147280773)
## Lines: 59. Separator is '    '. Parsing........... 6021 attributes, 58 instances.
## Loading source data from file Source.txt...
## Lines: 176. Separator is '   '. Parsing........... 6021 attributes, 175 instances.
## Writing training file...
## Writing source file...
```

```
## Discretizing using EBD using lambda of 0.5 (2009) .......... There are 361 attributes with cut poin
## Learning prior model...
## Done learning prior rules
## Doing 10-fold cross-validation...
## Fold 1. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 720 attributes with
## Prior structures 20. Imported 35. Retained 6 rules, 6 atts in the new model of 11 rules, 11 atts.
## Learned 11 rules
## Fold 2. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 863 attributes with
## Prior structures 20. Imported 35. Retained 5 rules, 5 atts in the new model of 11 rules, 11 atts.
## Learned 11 rules
## Fold 3. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 906 attributes with
## Prior structures 20. Imported 38. Retained 6 rules, 6 atts in the new model of 11 rules, 11 atts.
## Learned 11 rules
## Fold 4. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 1271 attributes with
## Prior structures 20. Imported 48. Retained 5 rules, 7 atts in the new model of 6 rules, 8 atts.
## Learned 6 rules
## Fold 5. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 962 attributes with
## Prior structures 20. Imported 35. Retained 5 rules, 5 atts in the new model of 10 rules, 10 atts.
## Learned 10 rules
## Fold 6. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 884 attributes with
## Prior structures 20. Imported 39. Retained 5 rules, 5 atts in the new model of 10 rules, 10 atts.
## Learned 10 rules
## Fold 7. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 958 attributes with
## Prior structures 20. Imported 36. Retained 7 rules, 7 atts in the new model of 10 rules, 10 atts.
## Learned 10 rules
## Fold 8. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 874 attributes with
## Prior structures 20. Imported 36. Retained 5 rules, 5 atts in the new model of 10 rules, 10 atts.
## Learned 10 rules
## Fold 9. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 947 attributes with
## Prior structures 20. Imported 35. Retained 6 rules, 6 atts in the new model of 11 rules, 11 atts.
## Learned 11 rules
## Fold 10. Discretizing using EBD using lambda of 0.5 (2009) .......... There are 956 attributes with
## Prior structures 20. Imported 37. Retained 7 rules, 7 atts in the new model of 9 rules, 9 atts.
## Learned 9 rules
##
## === Learner parameters ===
##
## Training data: Target.txt
## Discretization method: EBD using lambda of 0.5 (2009)
## Learning method: Rule Learner
## CF function: PPV (1)
## Min CF: 0.85
## Min conjuncts: 1
## Max conjuncts: 5
## Min coverage: 4.0
## Max FP coverage: 0.1
## Min TP coverage: 0.05
## Inductive strengthening: 1
## Inference type: Weighted voting (0)
## Beam width: 1000
## Specialize satisfactory rules: false
## Validation method: 10-fold cross-validation with seed 1
## Transfer of prior rules:
##     Source data: Source.txt
```

```
##     Transfer type: 2
##     Specialize prior rules: true
##     Ind. strengthening for prior rules: true
##
## === Stratified cross-validation performance ===
##
## Accuracy ignoring abst.:                89.29 %
## Accuracy including abst.:               86.21 %
## Abstentions:                            2 (3.45 %)
## Relative classifier information:        49.02 +/- 10.17
##
## Class  Sensitivity(%)           Specificity(%)           Balanced Accuracy(%)     AUROC
## ------ ------------------------ ------------------------ ------------------------ ------------------
## 1      91.67(4.3)               85(10.67)                88.33(6.68)              0.88(0.07)
## 0      85(10.67)                91.67(4.3)               88.33(6.68)              0.88(0.07)
##
## Confusion matrix:
##  c1  c2  Abstentions
##  34   3   2  | c1  =  1
##   3  16   0  | c2  =  0
##
##
## Learning on whole training data...
## Discretizing using EBD using lambda of 0.5 (2009) .......... There are 1000 attributes with cut poi
## Prior structures 20. Imported 39. Retained 6 rules, 6 atts in the new model of 12 rules, 12 atts.
##
## === Prior rules (20) ===
##
## 0. ((VEGFA = 9.367..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=116, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 1. ((LDHA = 12.670..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=109, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 2. ((ADCY7 = -inf..6.850)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=102, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 3. ((IGFBP2 = 11.040..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=102, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 4. ((BTAF1 = -inf..8.655)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=93, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 5. ((CSTF2T = -inf..7.885)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=91, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 6. ((ABCC10 = -inf..7.433)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=90, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 7. ((EML2 = 6.473..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=90, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 8. ((STRN = 5.110..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=89, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
```

```
##
## 9. ((PDIA4 = 8.723..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=89, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 10. ((ZIC3 = -inf..4.464)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=81, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 11. ((CTSL2 = 6.850..inf)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=72, FP=0, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 12. ((3-Mar = 9.338..inf)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=4, FP=0, Pos=16, Neg=159, TP_in_model=0, FP_in_model=0
##
## 13. ((CYP2E1 = -inf..4.844)) ==> (@Class = 1)
##  CF=0.992, P=0.0, TP=122, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 14. ((LRMP = -inf..6.911)) ==> (@Class = 1)
##  CF=0.991, P=0.0, TP=111, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 15. ((COMMD3 = -inf..10.126)) ==> (@Class = 1)
##  CF=0.991, P=0.0, TP=110, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 16. ((SERPINH1 = 8.732..inf)) ==> (@Class = 1)
##  CF=0.99, P=0.0, TP=103, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 17. ((SGPL1 = -inf..6.181)) ==> (@Class = 1)
##  CF=0.99, P=0.0, TP=103, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 18. ((EFNB2 = 7.846..inf)) ==> (@Class = 1)
##  CF=0.99, P=0.0, TP=99, FP=1, Pos=159, Neg=16, TP_in_model=0, FP_in_model=0
##
## 19. ((MCM2 = 7.411..inf) (ADORA1 = -inf..6.518) (ZNF187 = 8.643..inf)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=5, FP=0, Pos=16, Neg=159, TP_in_model=0, FP_in_model=0
##
## Attributes used (22):
## CTSL2, BTAF1, LDHA, ADCY7, STRN, PDIA4, ADORA1, SERPINH1, ZIC3, ZNF187, 3-Mar, SGPL1, EFNB2, ABCC10,
##
##
## Learned 12 rules
##
## === Learner parameters ===
##
## Training data: Target.txt
## Discretization method: EBD using lambda of 0.5 (2009)
## Learning method: Rule Learner
## CF function: PPV (1)
## Min CF: 0.85
## Min conjuncts: 1
## Max conjuncts: 5
## Min coverage: 4.0
## Max FP coverage: 0.1
## Min TP coverage: 0.05
## Inductive strengthening: 1
## Inference type: Weighted voting (0)
```

```
## Beam width: 1000
## Specialize satisfactory rules: false
## Validation method: 10-fold cross-validation with seed 1
## Transfer of prior rules:
##     Source data: Source.txt
##     Transfer type: 2
##     Specialize prior rules: true
##     Ind. strengthening for prior rules: true
##
## === Rules (12) ===
##
## p0. ((3-Mar = -inf..5.862)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=16, FP=0, Pos=39, Neg=19, TP_in_model=16, FP_in_model=0
##
## p1. ((IGFBP2 = -inf..6.238)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=11, FP=0, Pos=19, Neg=39, TP_in_model=11, FP_in_model=0
##
## p2. ((SERPINH1 = -inf..7.591)) ==> (@Class = 0)
##  CF=0.933, P=0.0, TP=14, FP=1, Pos=19, Neg=39, TP_in_model=14, FP_in_model=1
##
## p3. ((VEGFA = -inf..8.676)) ==> (@Class = 0)
##  CF=0.929, P=0.0, TP=13, FP=1, Pos=19, Neg=39, TP_in_model=13, FP_in_model=0
##
## p4. ((LDHA = -inf..10.110)) ==> (@Class = 0)
##  CF=0.875, P=0.0, TP=14, FP=2, Pos=19, Neg=39, TP_in_model=14, FP_in_model=1
##
## p5. ((PDIA4 = -inf..7.687)) ==> (@Class = 0)
##  CF=0.867, P=0.0, TP=13, FP=2, Pos=19, Neg=39, TP_in_model=13, FP_in_model=0
##
## 6. ((KLRC3 = -inf..3.986)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=27, FP=0, Pos=39, Neg=19, TP_in_model=27, FP_in_model=0
##
## 7. ((ZFPL1 = 6.737..7.026)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=24, FP=0, Pos=39, Neg=19, TP_in_model=23, FP_in_model=0
##
## 8. ((ATP9A = -inf..8.151)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=23, FP=0, Pos=39, Neg=19, TP_in_model=23, FP_in_model=0
##
## 9. ((NR1D2 = -inf..4.991)) ==> (@Class = 1)
##  CF=1.0, P=0.0, TP=21, FP=0, Pos=39, Neg=19, TP_in_model=21, FP_in_model=0
##
## 10. ((GGH = -inf..4.192)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=12, FP=0, Pos=19, Neg=39, TP_in_model=12, FP_in_model=0
##
## 11. ((TTC35 = 6.738..inf)) ==> (@Class = 0)
##  CF=1.0, P=0.0, TP=11, FP=0, Pos=19, Neg=39, TP_in_model=11, FP_in_model=0
##
## Attributes used (12):
## LDHA, KLRC3, PDIA4, SERPINH1, NR1D2, ZFPL1, 3-Mar, TTC35, GGH, ATP9A, VEGFA, IGFBP2
##
## === Classification performance on training data ===
##
## Accuracy ignoring abst.:                98.28 %
## Accuracy inclding abst.:                98.28 %
```

```
## Abstentions:                            0 (0 %)
## Relative classifier information:        89.18
##
## Class  Sensitivity(%) Specificity(%) Balanced Accuracy(%) AUROC
## ------ -------------- -------------- -------------------- --------------
## 1      97.44          100            98.72                0.99
## 0      100            97.44          98.72                0.99
##
## Confusion matrix:
##  c1  c2  Abstentions
##  38   1   0  | c1  =  1
##   0  19   0  | c2  =  0
##
##
## Total running time: 55 s.
```

The TRL output is also structured similar to the RL output shown with the baseline model. The cross-folds give additional information from transfer. The section (Stratified cross-validation performance) shows the performance on cross-validation. We see a mean AUC of **0.88** and a SEM of **0.07**. This is a gain of about **0.05** from the baseline performance of **0.83**. The accuracy with abstentions is **86.21%**, which is also an improvement over the baseline accuracy of **84.48%**.

The rule model learned on the source dataset that got transfered in this experiment is listed under (Prior rules) with 20 rules and 22 attributes. The final rule model learned on the target dataset as a result of seeding with the prior rule model is shown under (Rules). We see 12 rules here, out of which, the first 6 rules (prefixed by p) are prior rules that were imported from the source model. This model has 12 attributes: *LDHA, KLRC3, PDIA4, SERPINH1, NR1D2, ZFPL1, 3-Mar, TTC35, GGH, ATP9A, VEGFA, IGFBP2*. The last section shows the performance of this model on the overall target data (Classification performance on training data).

References

1. Gravendeel L, Kouwenhoven M, Gevaert O, de Rooi J, Stubbs A, et al. (2009) Intrinsic gene expression profiles of gliomas are a better predictor of survival than histology. Cancer research 69: 9065.
2. Petalidis L, Oulas A, Backlund M, Wayland M, Liu L, et al. (2008) Improved grading and survival prediction of human astrocytic brain tumors by artificial neural network analysis of gene expression microarray data. Molecular cancer therapeutics 7: 1013.