Recursive Similarity-Based Algorithm for Deep Learning

Tomasz Maszczyk & Włodzisław Duch

Nicolaus Copernicus University
Toruń, Poland

ICONIP 2012

{tmaszczyk,wduch}@is.umk.pl

Outline

- Main idea
- Similarity-Based Learning (SBL)
- Deep Learning
- Recursive Similarity-Based Learning (RSBL)
- Results
- Conclusions

Main idea

- Classification is one of the most important area of machine learning.
- SB methods (incl. many variants of kNN) the most popular and simplest.
- Advantages:
  - easy handling of unlimited number of classes,
  - stability of solutions against small perturbations of data.
- Their applications are limited, because $O(n^2)$.
- Calculations @ time of actual classification instead of training.
- Real-time decisions - too slow.

Main idea

- TRN of all SBM (incl. kernel-based SVM) suffers from the same quadratic scaling problem.
- Fast methods for finding approx. neighbours can reduce time to $O(\log n)$.
- Simple ML methods seems to reach their limits.
- Future belongs to techniques that autom. compose many transf. as it is done in:
  - Meta-Learning based on search in the model space,
  - learning based on generation of novel features,
  - Deep Learning.

Main idea

- RSBL inspired by success of DL.
- Kernel methods make only one step, replacing orig. features with SB features and performing LD in this space.
- DL in NN is based on learning in new feature spaces created by adding many network layers in essence performing recursive transf.
- Instead of seq. performing I/O transf., RSBL systematically expands the feature space using info from all previous stages of data transf.
- Only transf. based on similarities to the nearest k-samples scaled by Gaussian kernel features.
- Other similarity measures may be used in the same way.
- In essence this connects SBM with DL techniques, creating higher-order kNN method with kernel features.

Similarity-Based Learning

- Categorization based on distance to points in TRN dataset is a simple and effective way of classification.
- Many parameters and procedures can be included in the data models M based on similarity.
- M optimized to calculate posterior probability $p(C_i|\boldsymbol{x}; M)$ that a vector $\boldsymbol{x}$ belongs to class C_i.
- Optimization includes:
  - type of distance functions, or type of kernel $D(\boldsymbol{x}, \boldsymbol{y})$,
  - selection of reference instances,
  - weighting of their influence.

Similarity-Based Learning

The most common distance functions:

- Minkowski's metric $D(\boldsymbol{x},\boldsymbol{y})^\alpha = \sum_{i=1}^d |x_i - y_i|^\alpha$, becoming Euclidean metric for $\alpha = 2$, the city block metric for $\alpha = 1$ and the Chebychev metric for $\alpha = \infty$.
- Mahalanobis distance $D(\boldsymbol{x},\boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})'\boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{y})}$ where $\boldsymbol{C}$ is the covariance matrix, taking into account scaling and rotation of data clusters.
- Cosine distance, equal to the normalized dot product $D(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}\cdot\boldsymbol{y}/||\boldsymbol{x}||||\boldsymbol{y}||$.
- Hamming distance is used for binary features $D(\boldsymbol{x},\boldsymbol{y}) = \#(x_i \neq y_i)/d$.
- Correlation distance is also often used:

$$D(\boldsymbol{x},\boldsymbol{y}) = \frac{\sum_{i=1}^d (x_i-\overline{x})(y_i-\overline{y})}{\sqrt{\sum_{i=1}^d (x_i-\overline{x})^2 \sum_{i=1}^d (y_i-\overline{y})^2}}$$

Deep Learning

- Start from rich information $->$ series of transformations $->$ output sufficient for high-level decision.
- This information compression process can be presented as a network, a flow graph where each node = elementary data transformation.
- Flow graphs have different depth.
- Popular classificators - low depth (SVM, RBF, kNN) $= 2$ (kernel/distance, LD or selection of nearest neighbours). MLP depth depends on no. hidden layers.
- Low depth, but universal approx (can represent arbitrary function to a given target acc).
- Bengio shows examples of functions that can be represented in a simple way with deep archit., but a shallow one may require exp large no. of nodes in the flow graph and difficult to optimize.

Deep Learning

- Motivation for DL - comes from signal processing by the brain, for ex. image processing - done im many areas, each extracts some features from input and communicates results to the next level. Each level represents the input at a different level of abstraction with more abstract features further up in the hierarchy.
- People organize ideas and concepts hierarchically, learning first simpler concepts ant then composing them to represent more abstract ones.
- Engineers break-up solutions into multiple levels of abstraction and processing using divide-and-conquer at many levels.
- RSBL is inspired by such observations.

Recursive Similarity-Based Learning

- DL combined with distance-based and Gaussian kernel features - recursive supervised algorithm to create new features; used to provide optimal feature space for any classificator.
- RSBL - Euclidean distance + Gaussian kernel features (σ=0.1) $->$ new feature spaces @ each depth level.
- Classification done by SVML (C=2^5) or 1NN.
- In each case $k_{\max} = 20$ and $\alpha = 5$.
- In essence RSBL at each level of depth transforms the actual feature space into the extended feature space, discovering useful info by creating new redundant features.
- Original features are available at each depth.

Recursive Similarity-Based Learning

- The final analysis in the extended space (and optimization of parameters at each level of RSBL, including feature selection) may be done by various ML methods.
- Emphasis on generation of new features using DL rather than optimization of learning.
- RSBL may be presented as a constructive algorithm, with new layers representing transformations and procedures to extract and add to the overall pool more features, and a final layer analyzing the image of data in the enhanced feature space.

Recursive Similarity-Based Learning

Algorithm 1. Recursive similarity-based learning

Require: Fix the values of internal parameters: $k_{\max}$, maximum depth α, and σ (dispersion).

Standardize the dataset, n vectors, d features.
Set the initial space $\mathcal{H}^{(0)}$ using input features x_{ij}, $i = 1..n$ vectors and $j = 1..d$ features.
Set the current number of features $d(0) = d$.
for $m = 1$ to α **do**
  for $k = 1$ to $k_{\max}$ **do**
    For every training vector $\boldsymbol{x}_i$ find k nearest neighbors $\boldsymbol{x}_{j,i}$ in the $\mathcal{H}^{(m-1)}$ space.
    Create nk new kernel features $z_{j,i}(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{x}_{j,i})$, $j = 1..k$; $i = 1..n$ for all vectors using kernel functions as new features.
    Add new nk features to the $\mathcal{H}^{(m-1)}$ space, creating temporary $\mathcal{H}^{(m,k)}$ space.
    Estimate error $E(m, k)$ in the $\mathcal{H}^{(m,k)}$ space on the training or validation set.
  end for
  Choose k' that minimizes $E(m, k')$ error and retain $\mathcal{H}^{(m,k')}$ space as the new $\mathcal{H}^{(m)}$ space.
end for
Build the final model in the enhanced feature space $\mathcal{H}^{(\alpha)}$.
Classify test data mapped into the enhanced space.

Recursive Similarity-Based Learning

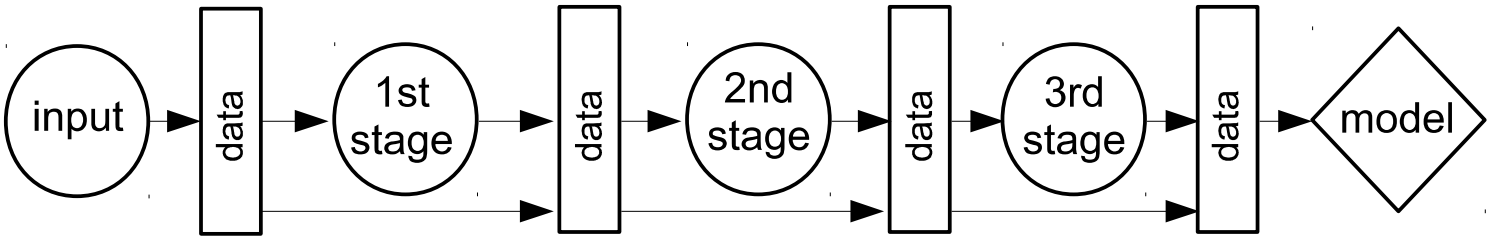

Figure: RSBL method presented in graphical form for depth equal three.

Results

- Many sophisticated ML methods introduced every year, tested on trivial problems from UCI.
- Most of them easy: simple and fast algorithms with $O(nd)$ - results s.s. not worse worse than those obtained by the best known alg.
- Some benchmark problems are not trivial - require complicated decision borders and may be handled only using sophisticated techniques.
- To distinguish dataset trivial or not - simple methods with $O(nd)$ complexity have been compared with the optimized SVMG.
- New methods should improve results of simple ML methods in non-trivial cases.

Results

Table: Summary of datasets used in experiments.

Dataset	#Vectors	#Features	#Classes
ionosphere	351	34	2
monks-1	556	6	2
monks-2	601	6	2
parkinsons	195	22	2
sonar	208	60	2

Duch W., Jankowski N., Maszczyk T.: **Make it cheap: learning with O(nd) complexity.** In: Proceedings of the IEEE World Congress on Computational Intelligence, Brisbane, Australia (2012), pp. 132-135

Which dataset from UCI is Trivial

(our previous work)

Data	Trivial	MajorityClass	1NP	MLC	LVQ	NaiveBayes	K2MLP	SVML	SVMG
arrhythmia	+	29.43±7.82	44.00±27.42	29.79±14.76	***63.69±17.46***	10.19±9.80	58.00±16.63	50.92±17.31	43.36±21.47
autos	+	28.18±3.14	56.83±10.39	62.23±11.97	30.23±10.18	63.32±10.28	**70.44±10.31**	54.48±13.75	74.29±12.58
balance-scale	+	45.37±0.55	70.07±6.19	53.80±5.56	89.88±9.10	90.80±1.40	***91.15±4.03***	84.47±3.17	89.83±2.09
blood-transfusion-service-center	+	**76.20±0.57**	69.13±3.74	60.28±3.08	76.20±0.57	75.01±3.02	70.59±8.35	76.20±0.48	79.14±4.57
breast-cancer-wisconsin-diagnostic	+	62.74±0.73	93.08±3.47	91.78±3.44	93.04±3.47	93.18±3.47	**97.00±2.33**	97.36±2.51	97.54±2.50
breast-cancer-wisconsin-original	+	65.01±0.83	96.44±2.22	94.46±2.62	95.83±2.22	96.24±2.30	**96.72±2.17**	96.48±2.49	96.77±2.48
breast-cancer-wisconsin-prognostic	+	**76.34±3.02**	62.64±10.02	74.17±8.28	76.34±3.02	66.39±8.83	73.00±9.92	77.84±8.03	76.86±8.20
breast-tissue	+	17.68±3.64	61.73±11.70	***68.34±11.45***	15.85±6.55	64.03±12.78	63.04±11.17	53.00±10.87	63.27±7.59
car-evaluation	−	70.02±0.16	73.22±2.90	84.08±2.50	73.59±3.65	3.76±0.33	**91.13±2.54**	69.57±2.03	98.84±0.77
cardiotocography-1	+	27.23±0.16	54.80±3.02	70.98±2.46	27.23±0.16	72.57±2.28	**77.88±2.83**	57.90±4.12	80.43±2.79
cardiotocography-2	−	77.85±0.31	76.57±1.78	73.72±2.21	77.85±0.31	82.54±1.86	**87.32±2.69**	87.53±1.48	92.09±2.01
chess-king-rook-vs-king-pawn	−	52.22±0.12	86.25±1.33	83.86±1.68	61.40±15.17	67.27±1.81	**90.90±3.29**	96.21±1.38	99.28±0.36
CMC (Contraceptive Methods)	+	42.70±0.23	46.03±3.52	47.89±3.62	22.61±0.34	***49.64±3.96***	48.82±3.40	19.14±2.14	34.09±3.67
congressional-voting-records	+	53.46±2.26	89.86±5.40	94.73±4.37	89.69±5.27	94.25±5.11	***94.95±3.94***	94.48±3.62	92.65±4.11
connectionist-bench-sonar	−	53.35±2.26	69.65±7.49	70.62±5.99	71.67±7.44	69.03±8.68	**76.73±8.05**	75.47±8.26	85.52±5.28
connectionist-bench-vowel	−	7.58±0.06	51.98±6.61	52.00±5.99	9.09±1.36	67.53±6.28	**80.95±5.02**	25.76±5.01	96.77±2.20
cylinder-bands	+	64.25±1.17	68.97±8.43	38.68±7.02	64.47±4.26	**74.07±7.51**	71.35±8.32	74.58±5.23	76.89±7.57
dermatology	+	31.01±0.97	**96.87±3.15**	88.40±4.55	91.30±3.79	90.13±4.52	94.88±3.84	94.01±3.54	94.49±3.88
ecoli	+	42.57±1.58	81.38±5.76	77.13±12.09	78.50±9.39	70.76±20.46	**83.55±6.04**	78.48±5.90	84.17±5.82
glass	+	35.54±2.56	48.82±9.88	48.67±6.25	34.79±4.52	43.34±8.44	**59.78±8.84**	42.61±10.05	62.43±8.70
habermans-survival	+	73.54±1.86	74.45±7.19	71.21±7.65	73.19±4.07	***74.83±5.58***	64.34±14.53	73.52±1.86	71.55±8.42
hepatitis	+	83.75±5.76	82.50±13.06	90.38±10.18	83.75±5.76	**91.25±9.15**	84.00±12.32	83.25±11.54	84.87±11.98
ionosphere	−	64.10±1.43	81.14±6.41	59.23±6.24	83.72±5.34	84.24±6.15	**86.46±5.48**	87.72±4.63	94.61±3.68
iris	+	33.33±0.00	85.80±8.67	94.60±5.42	85.67±8.50	95.40±5.42	***95.60±4.76***	72.20±7.59	94.86±5.75
libras-movement	−	4.64±1.31	56.86±6.22	51.92±7.07	6.67±1.48	**65.50±6.57**	52.67±8.17	49.16±5.24	84.44±6.02
liver-disorders	+	57.96±1.62	57.60±8.18	**65.12±7.97**	57.85±3.43	56.28±7.93	62.61±8.29	68.46±7.36	70.30±7.90
lymph	+	54.71±4.52	***86.43±8.61***	78.79±9.36	82.49±9.35	81.18±8.95	82.67±9.24	81.26±9.79	83.61±9.82
monks-problems-1	−	49.46±0.60	74.64±4.18	74.64±4.18	70.58±10.05	52.28±2.05	**83.07±3.03**	65.81±6.50	99.82±0.56
monks-problems-2	−	65.72±0.84	54.90±5.85	53.85±6.19	65.54±1.28	54.50±4.21	**74.76±5.14**	65.72±0.82	84.86±4.91
monks-problems-3	+	51.99±0.86	96.39±2.17	96.39±2.17	96.39±2.17	94.78±5.79	***98.57±1.69***	80.13±4.91	96.75±2.22
parkinsons	−	75.44±3.19	73.55±8.71	78.22±8.46	77.76±6.89	69.83±9.09	**85.64±7.57**	86.26±10.17	93.26±5.61
pima-indians-diabetes	+	65.10±0.54	72.72±4.84	68.63±4.66	75.02±4.50	**75.30±4.39**	73.82±5.03	77.08±4.27	76.04±3.69
sonar	−	53.35±2.26	69.65±7.49	70.62±5.99	71.67±7.44	69.03±8.68	**76.73±8.05**	73.71±9.62	86.42±7.65
spambase	+	60.60±0.10	89.50±1.31	87.41±1.44	82.79±11.23	81.78±1.52	**91.58±1.57**	92.96±1.30	93.69±1.04
SPECT-heart	+	79.42±2.05	72.15±8.20	***83.64±6.80***	79.42±2.05	72.19±6.99	77.37±7.75	82.72±7.43	83.50±6.62
SPECTF-heart	+	**79.42±2.05**	66.48±8.43	79.31±2.06	79.42±2.05	67.68±8.35	72.89±11.01	78.61±9.73	80.18±7.34
statlog-australian-credit	+	55.51±0.67	**84.45±4.35**	79.55±4.81	83.13±7.53	79.54±4.40	82.43±4.69	85.50±3.86	84.49±3.48
statlog-german-credit-numeric	+	70.00±0.00	72.81±4.26	67.55±4.73	72.09±3.38	72.84±4.33	**72.93±4.61**	77.50±2.32	76.01±3.16
statlog-heart	+	55.56±0.00	84.63±6.08	82.41±7.41	***85.07±6.24***	84.22±7.13	82.44±7.00	82.96±7.65	81.48±4.61
statlog-vehicle-silhouettes	−	25.12±0.54	45.33±4.57	52.96±4.20	25.77±0.86	45.69±3.98	**72.85±4.65**	69.86±2.74	79.78±2.66
teaching-assistant-evaluation	+	34.45±2.74	50.85±12.71	48.58±12.82	33.05±3.60	24.32±8.88	**52.09±12.00**	13.25±9.94	42.37±9.44
thyroid-disease	−	92.58±0.09	71.07±1.84	86.56±1.44	92.58±0.09	36.55±3.36	**94.74±2.17**	93.76±0.47	97.47±0.66
vote	+	53.46±2.26	89.86±5.40	94.73±4.37	89.69±5.27	91.93±4.98	**95.29±4.04**	96.12±3.85	96.89±3.11
wine	+	39.91±1.85	97.25±3.94	96.84±3.72	97.25±3.94	**97.30±3.80**	96.18±3.91	97.71±2.95	97.15±4.08
ZOO	+	40.63±3.19	**91.55±6.91**	86.34±8.54	83.45±7.11	86.82±8.58	83.25±8.42	91.61±6.67	93.27±7.53
wins | ties | losses		**31 | 64 | 130**	**75 | 73 | 77**	**73 | 64 | 88**	**80 | 59 | 86**	**81 | 63 | 81**	**137 | 61 | 27**		

Results

Table: 10 x 10 crossvalidation accuracy and standard deviation for RSBL combined with SVM.

Dataset	Method						
	ORG	RSBL(1)	RSBL(2)	RSBL(3)	RSBL(4)	RSBL(5)	SVMG
ionosphere	88.2±6.4	92.3±3.8	94.0±3.9	94.0±3.9	94.0±3.9	94.0±3.9	94.6±3.7
monks-1	74.6±4.6	100±0.0	100±0.0	100±0.0	100±0.0	100±0.0	99.8±0.6
monks-2	65.7±0.6	79.6±3.1	84.9±4.1	85.7±4.2	85.7±4.2	85.7±4.2	84.9±4.9
parkinsons	88.7±7.8	89.3±5.4	93.3±4.9	91.3±6.0	89.2±5.1	87.7±5.4	93.2±5.6
sonar	74.9±9.5	82.2±7.9	85.1±4.6	86.6±7.0	87.4±7.7	87.9±7.3	86.4±7.6

Results

Table: 10 x 10 crossvalidation accuracy and standard deviation for RSBL combined with 1NN.

Dataset	Method					
	ORG	RSBL(1)	RSBL(2)	RSBL(3)	RSBL(4)	RSBL(5)
ionosphere	87.1±5.2	87.4±4.8	87.8±4.9	87.8±4.9	87.8±4.9	87.8±4.9
monks-1	100±0.0	99.9±0.1	99.4±1.2	99.3±1.2	99.4±1.1	99.4±1.0
monks-2	68.8±6.2	69.2±8.7	71.6±6.2	71.6±6.2	71.6±6.2	71.8±6.2
parkinsons	93.8±5.4	92.8±6.6	91.7±6.1	91.7±6.1	91.7±6.1	91.7±6.1
sonar	85.0±5.8	85.5±6.8	86.0±6.6	87.9±6.5	87.9±6.5	87.9±6.5

Conclusions

- The most important goal of CI - create methods that can automatically discover the best models for a given data.
- Simple methods - no; therefore such techniques like DL, meta-L or feature construction should be used.
- RSBL focused on hierarchical heneration of new distance-based and kernel-based features rather than improvement in optimization and classification alg.
- Finding interesting views on the data by systematic addition of novel features is very important, because combination of such transformation-based systems should bring us closer to the practical applications that automatically create the best data models for any data.

Conclusions

- Results on several non-trivial problems shows that RSBL creates explicitly feature spaces in which linear methods reach results that are at least as good as optimized SVMG.
- Further improvements:
  - different distance measures,
  - fast approximate neighbors,
  - feature selection and global optimization of the whole procedure.

Thank you for your attention.