ORA 2024

Twin Support Vector Machines (TSVM):

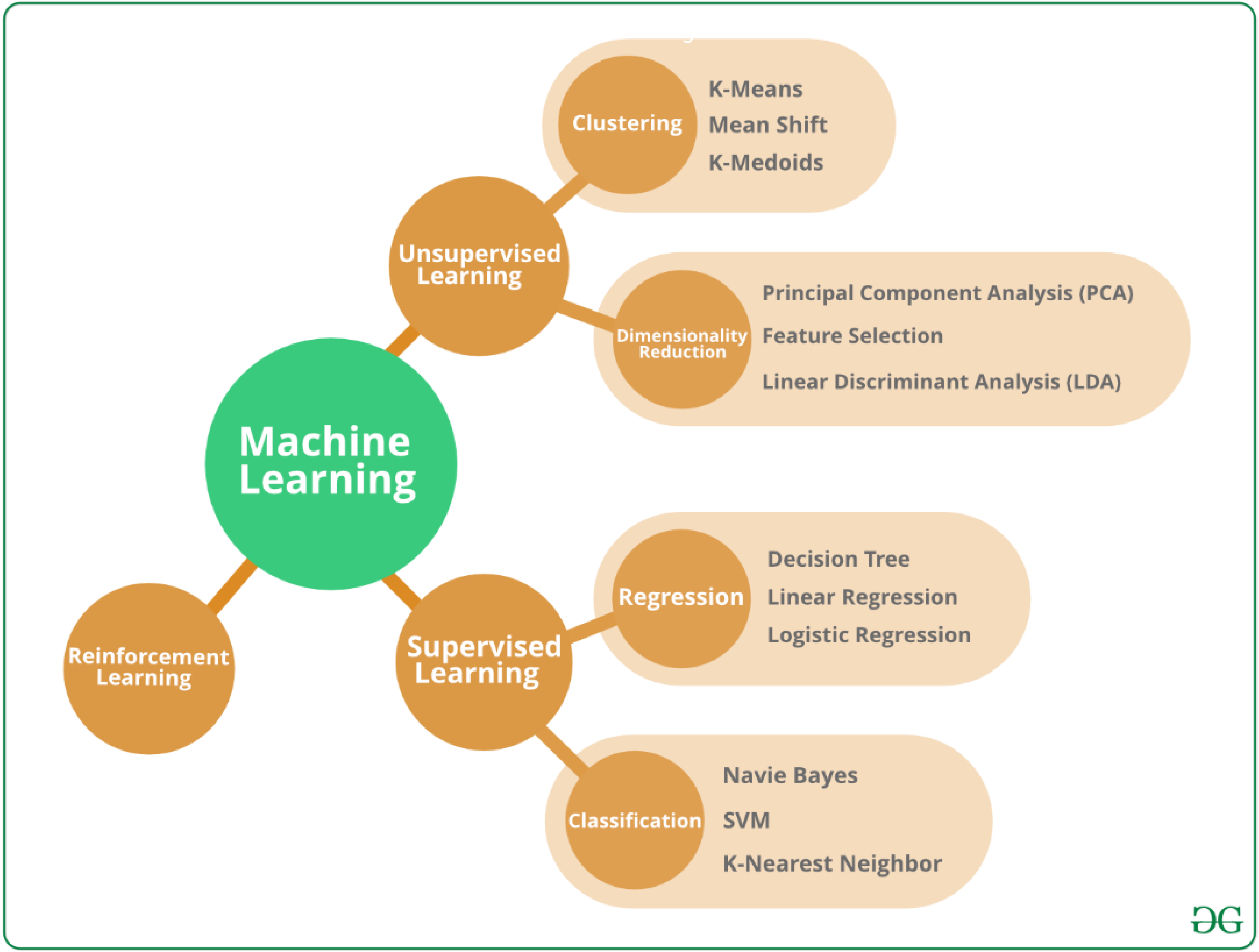
Recent Advances and Challenges

Panos M. Pardalos
www.ise.ufl.edu/pardalos
https://nnov.hse.ru/en/latna/

XVI Summer School on Operations Research, Data, and Decision Making,

ORA 2024, May 23-24, 2024.

Machine Learning

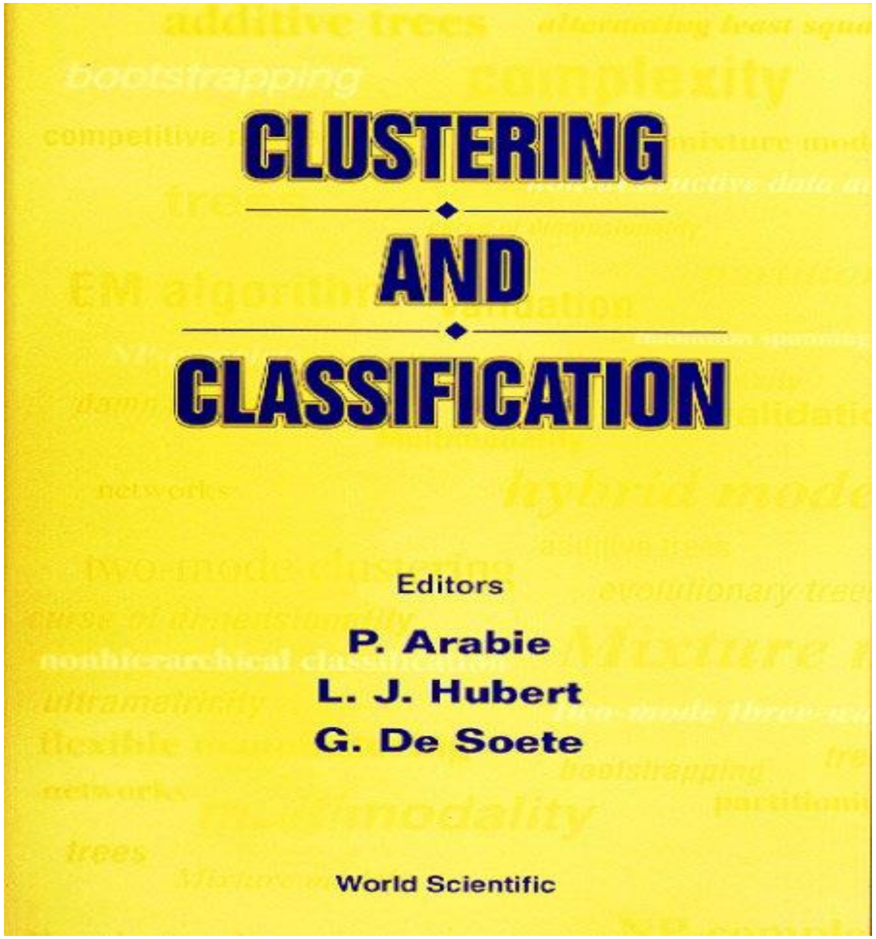

Machine Learning
Unsupervised Learning
Clustering
K-Means
Mean Shift
K-Medoids
Dimensionality Reduction
Principal Component Analysis (PCA)
Feature Selection
Linear Discriminant Analysis (LDA)
Supervised Learning
Regression
Decision Tree
Linear Regression
Logistic Regression
Classification
Navie Bayes
SVM
K-Nearest Neighbor
Reinforcement Learning

Classification and Clustering in Data Analysis

- Classification **(supervised learning)** uses **predefined classes** in which objects are assigned, while clustering (**unsupervised learning) identifies similarities between objects**, which it **groups** according to those characteristics in common and which differentiate them from other groups of objects. These groups are known as "**clusters**".

Clustering and Classification

(P Arabie, L J Hubert, and G De Soete https://doi.org/10.1142/1930 | January 1996)

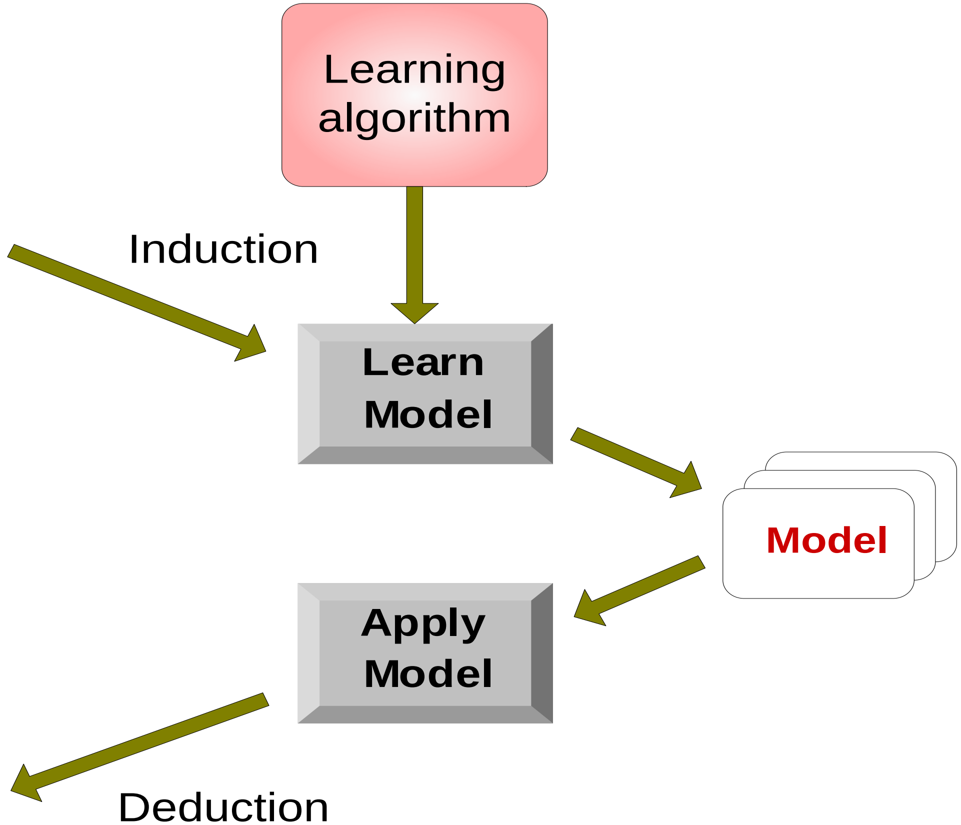


Supervised Learning

General Approach for Building Classification Model

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

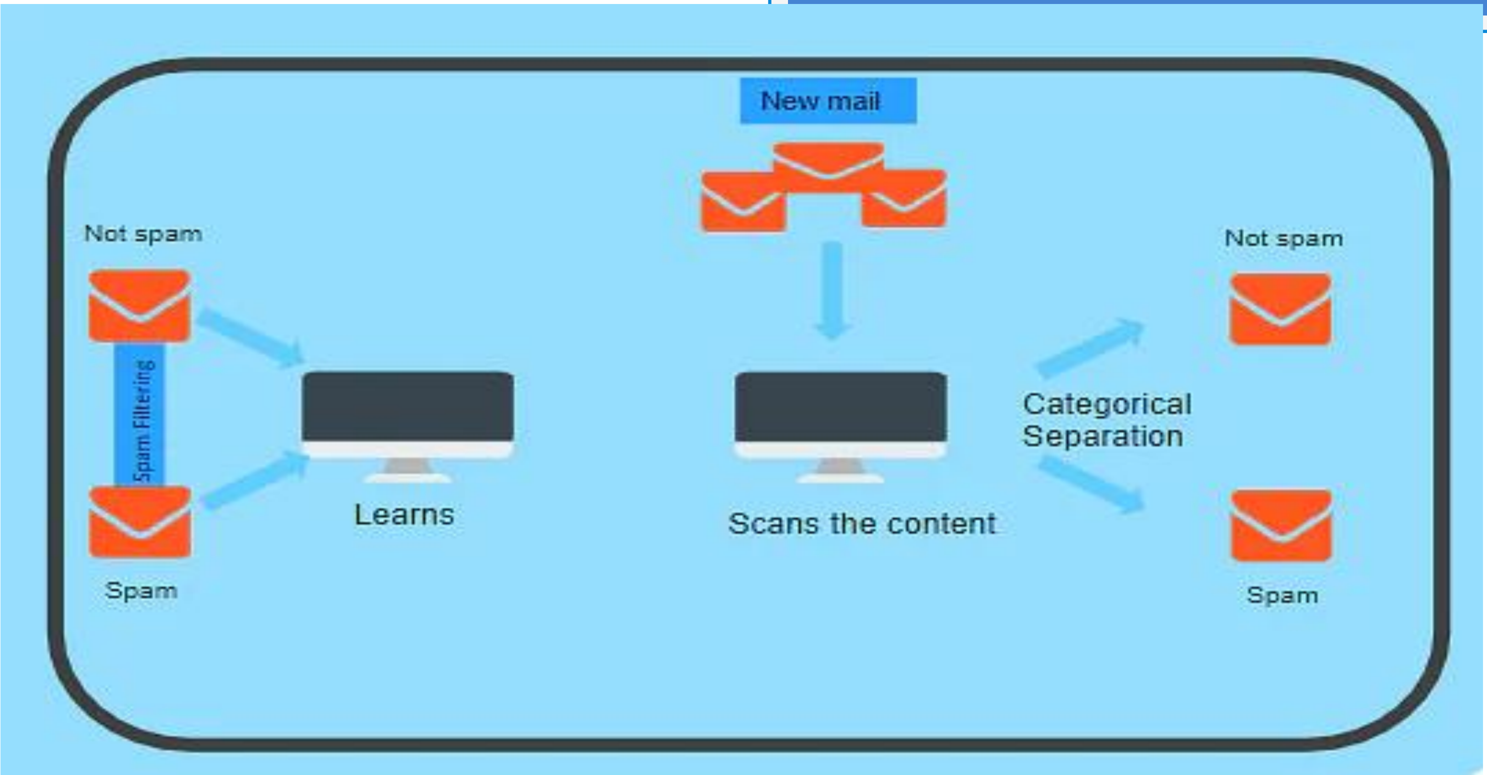
Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set


EMAIL FILTER
inbox
Spam



In order to predict whether a mail is spam or not, we need to first teach the machine what a spam mail is. This is done based on a lot of spam filters - reviewing the content of the mail, reviewing the mail header and so on.

Based on the content, label, and the spam score of the new incoming mail, the algorithm decides whether it should land in the inbox or spam folder.

Applications of Classification Algorithms

- Speech recognition
- Face recognition
- Handwriting recognition
- Biometric identification
- Document classification
- Fraud detection in finance
- Biomedicine

Computer Methods and Programs in Biomedicine 141 (2017) 19–26

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm

Zeinab Arabasadi [a], Roohallah Alizadehsani [b,*], Mohamad Roshanzamir [c], Hossein Moosaei [d], Ali Asghar Yarifard [a]

[a] *Department of Computer Engineering, University of Bojnord, Bojnord, Iran*
[b] *Department of Computer Engineering, Sharif University of Technology, Azadi Ave, Tehran, Iran*
[c] *Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran*
[d] *Department of Mathematics, Faculty of Science, University of Bojnord, Iran*

A R T I C L E I N F O

Article history:
Received 11 September 2016
Revised 18 December 2016
Accepted 12 January 2017

Keywords:
Cardiovascular disease

A B S T R A C T

Cardiovascular disease is one of the most rampant causes of death around the world and was deemed as a major illness in Middle and Old ages. Coronary artery disease, in particular, is a widespread cardiovascular malady entailing high mortality rates. Angiography is, more often than not, regarded as the best method for the diagnosis of coronary artery disease; on the other hand, it is associated with high costs and major side effects. Much research has, therefore, been conducted using machine learning and data mining so as to seek alternative modalities. Accordingly, we herein propose a highly accurate hybrid method for the diagnosis of coronary artery disease. As a matter of fact, the proposed method is able to increase

Arabasadi, Z., Alizadehsani, R., Roshanzamir, M., **Hossein Moosaei**, & Yarifard, A. A. (2017). Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm. *Computer methods and programs in biomedicine*, *141*, 19-26.

The dataset recorded 303 patients, each of which has 54 features.

Feature type	Feature name	Range
Demographic	Age	*30–86*
	Weight	*48–120*
	Sex	*Male, Female*
	BMI (Body Mass Index Kg/m^2)	*18–41*
	DM (Diabetes Mellitus)	*Yes, No*
	HTN (Hypertension)	*Yes, No*
	Current smoker	*Yes, No*
	Ex-smoker	*Yes, No*
	FH (Family History)	*Yes, No*
	Obesity	*Yes if MBI > 25, No otherwise*
	CRF (Chronic Renal Failure)	*Yes, No*
	CVA (Cerebrovascular Accident)	*Yes, No*
	Airway disease	*Yes, No*
	Thyroid disease	*Yes, No*
	CHF (Congestive Heart Failure)	*Yes, No*
	DLP (Dyslipidemia)	*Yes, No*
Symptom and examination	BP (Blood Pressure mm Hg)	*90–190*
	PR (Pulse Rate ppm)	*50–110*
	Edema	*Yes, No*
	Weak peripheral pulse	*Yes, No*
	Lung rales	*Yes, No*
	Systolic murmur	*Yes, No*
	Diastolic murmur	*Yes, No*
	Typical chest pain	*Yes, No*
	Dyspnea	*Yes, No*
	Function class	*1, 2, 3, 4*
	Atypical	*Yes, No*
	Nonanginal chest pain	*Yes, No*
	Exertional chest pain	*Yes, No*
	Low Th Ang (low-Threshold angina)	*Yes, No*
ECG	Rhythm	*Sin, AF*
	Q wave	*Yes, No*
	ST elevation	*Yes, No*

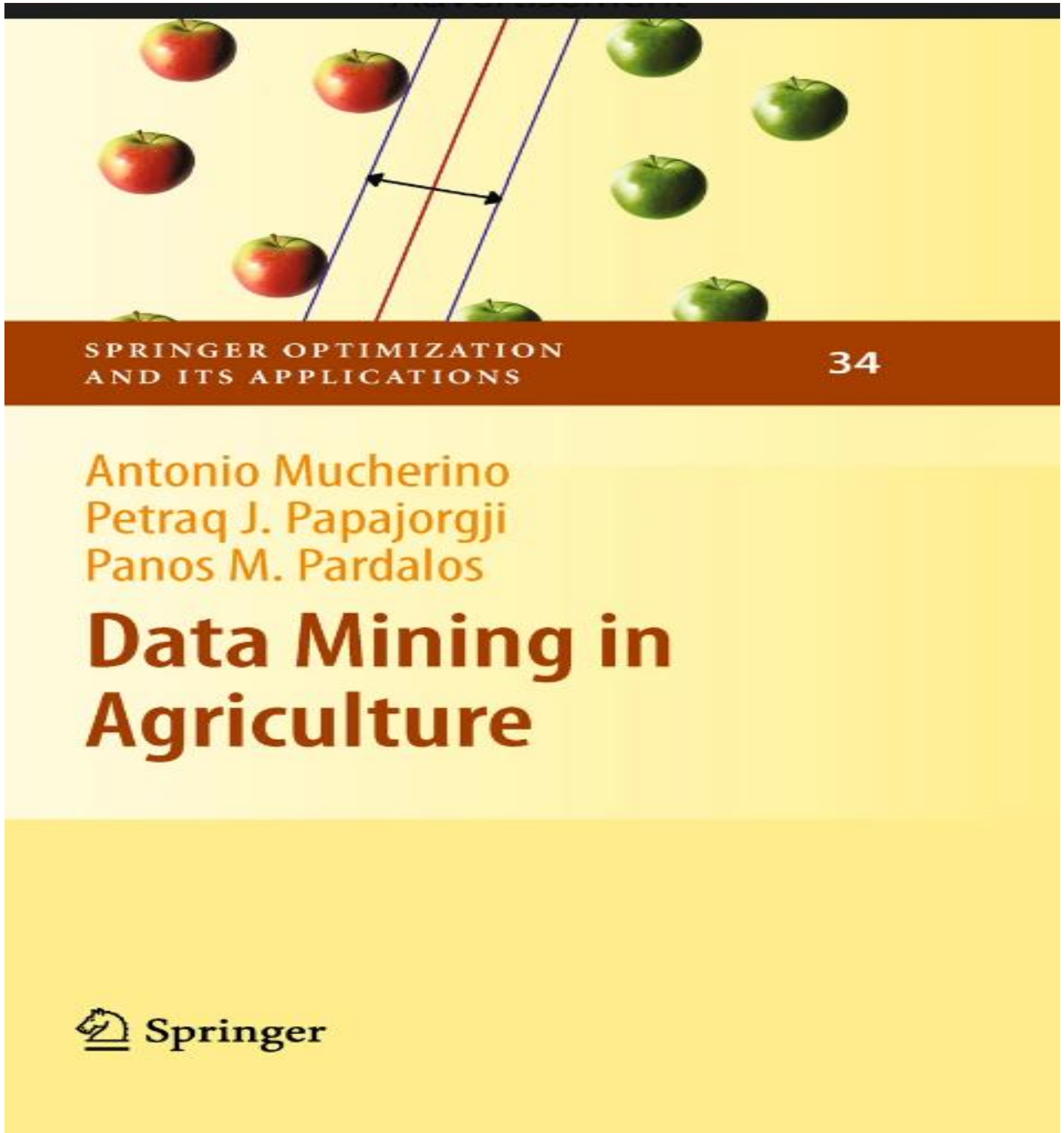
1	HB	K	Na	WBC	Lymph	Neu	PLT	EF-TT	ion R	VHD	Cath
2	15.6	4.7	141	5700	39	52	261	50	0	N	Cad
3	13.9	4.7	156	7700	38	55	165	40	4	N	Cad
4	13.5	4.7	139	7400	38	60	230	40	2	mild	Cad
5	12.1	4.4	142	13000	18	72	742	55	0	Severe	Normal
6	13.2	4	140	9200	55	39	274	50	0	Severe	Normal
7	15.6	4.2	141	7300	26	66	194	50	0	N	Cad
8	14.1	4.8	139	9400	58	33	292	40	4	mild	Cad
9	16.1	4.3	142	12200	25	74	410	45	4	mild	Cad
10	11.6	3.4	139	5100	49	50	370	50	0	N	Normal
11	13.9	4.6	140	4900	55	42	380	40	2	N	Cad
12	14.5	4.2	142	5800	44	52	201	50	0	mild	Cad
13	10	4.3	128	11000	31	66	290	50	3	mild	Cad
14	12.3	4.3	148	11300	25	70	380	25	4	Moderate	Cad
15	14.3	4.5	139	8100	31	65	254	55	2	mild	Cad
16	12.9	4.3	139	6400	60	39	217	55	0	mild	Cad
17	13.3	4.7	146	12100	30	70	280	30	0	Moderate	Cad
18	11.4	4.6	148	7800	48	50	199	35	0	Severe	Normal
19	13	4.6	141	4900	35	60	194	60	0	N	Normal
20	13.1	3.5	140	3700	30	68	180	55	0	N	Cad
21	12.4	3.8	145	5300	45	50	227	50	1	N	Cad
22	15.4	4.3	142	6500	40	60	184	50	0	N	Cad
23	10	4.3	143	5600	34	60	194	55	0	N	Cad
24	14.9	3.6	135	7600	32	66	184	30	0	N	Cad
25	13.5	4.9	138	9600	28	72	190	50	0	N	Cad
26	12.7	4.6	138	5800	31	60	180	50	0	N	Cad
27	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

The last column shows whether the person is healthy or sick.

- Rehman, Mujeeb Ur, et al. "Future forecasting of COVID-19: a supervised learning approach." *Sensors* 21.10 (2021): 3322.
- Ye, Qinghao, et al. "Robust weakly supervised learning for COVID-19 recognition using multi-center CT images." *Applied Soft Computing* 116 (2022): 108291.
- Guleria, Kalpna, et al. "Breast cancer prediction and classification using supervised learning techniques." *Journal of Computational and Theoretical Nanoscience* 17.6 (2020): 2519-2522.
- Chitra, R., and V. Seenivasagam. "Heart disease prediction system using supervised learning classifier." *Bonfring International Journal of Software Engineering and Soft Computing* 3.1 (2013): 01-07.

Classification Techniques

- Neural Networks
- Random Forest
- Decision Trees
- Nearest Neighbor
- Boosted Trees
- Linear Classifiers: Logistic Regression, Naïve Bayes Classifier
- **Support Vector Machines**


SPRINGER OPTIMIZATION
AND ITS APPLICATIONS
34
Antonio Mucherino
Petraq J. Papajorgji
Panos M. Pardalos
Data Mining in
Agriculture
Springer


Information Science and Statistics
Ingo Steinwart • Andreas Christmann
Support Vector Machines
Springer

Support Vector Machine (SVM)

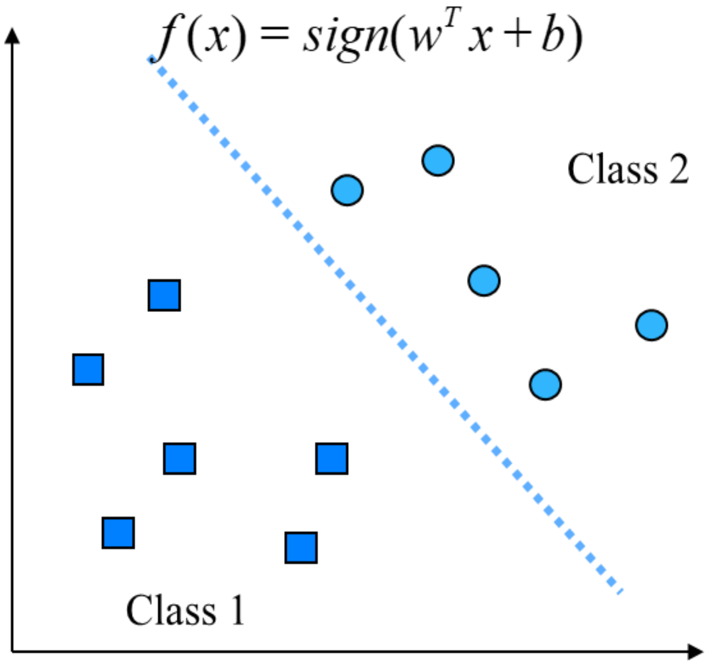
What is a good Decision Boundary?

- Consider a two-class, linearly separable classification problem. Construct the hyperplane

$$w^T x + b = 0, \qquad x \in R^n$$

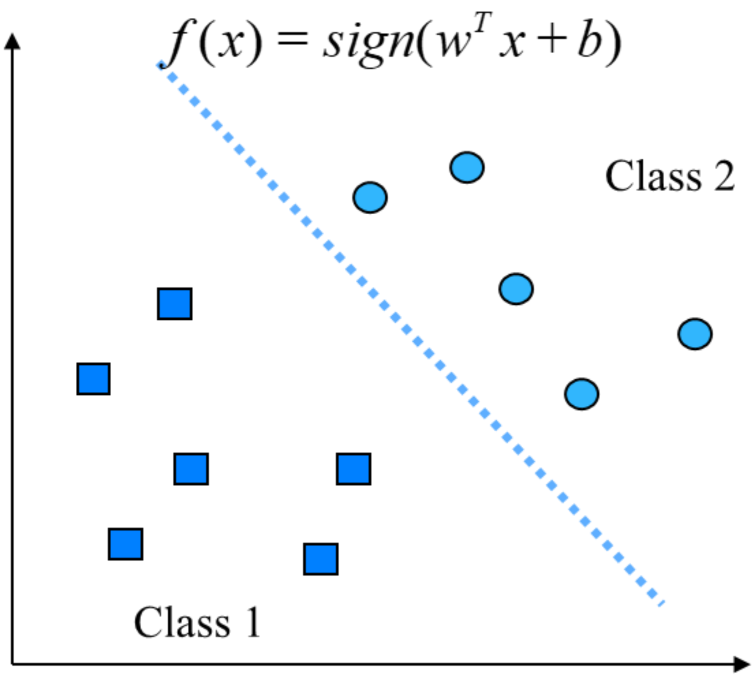
- to make

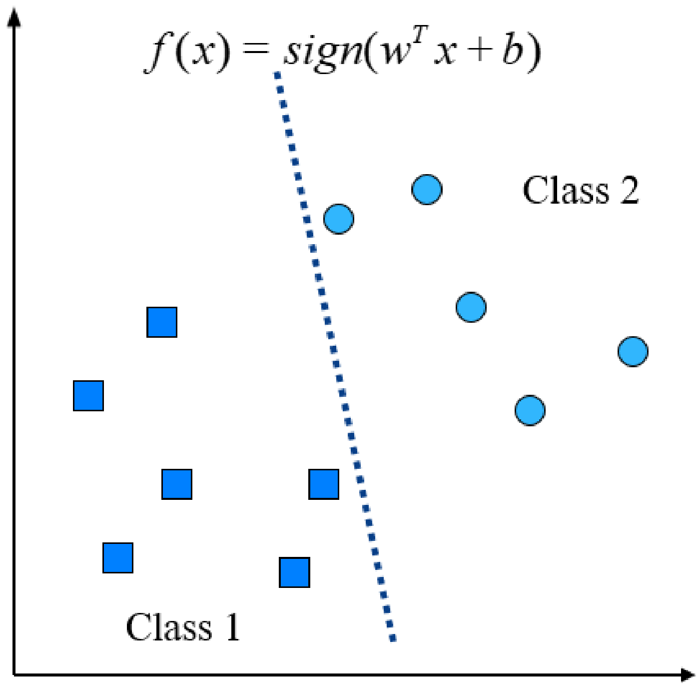
$$w^T x_i + b > 0, \qquad for \quad y_i = +1$$
$$w^T x_i + b < 0, \qquad for \quad y_i = -1$$



- Many decision boundaries! Are all decision boundaries equally good?

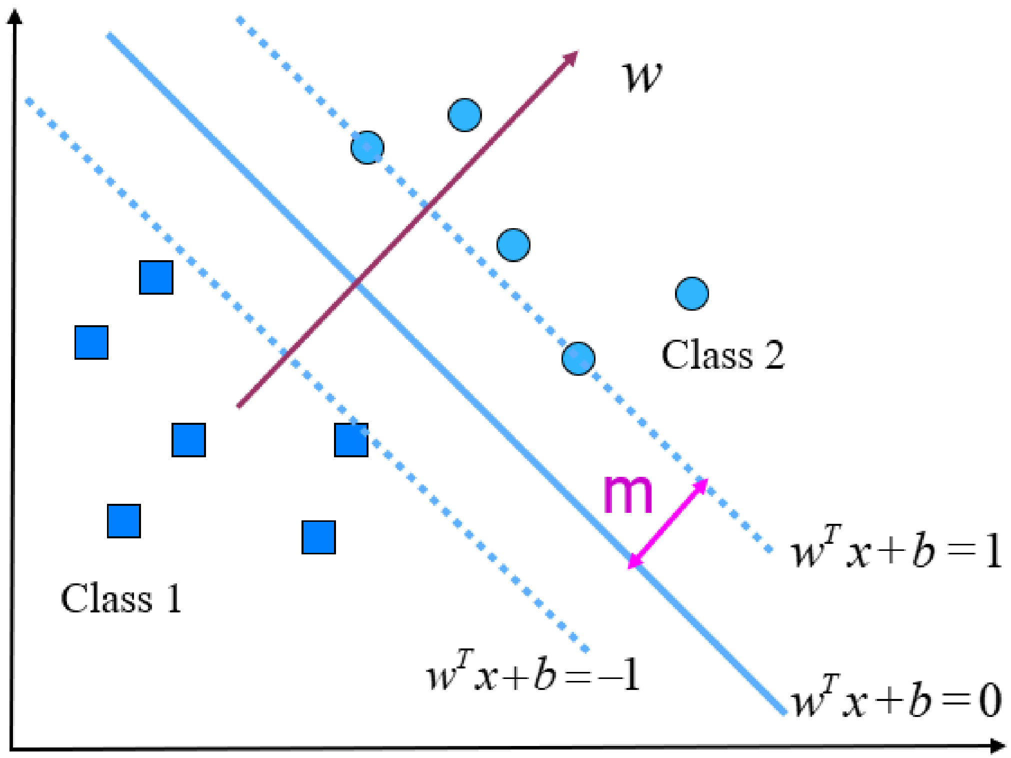
Examples of Bad Decision Boundaries





Optimal separating hyperplane

- The optimal separating hyperplane



- For the hyperplane, it can be proved that the margin m is

$$m = \frac{1}{\|w\|^2}$$

Hence, maximizing margin is equivalent to minimizing the square of the norm of w.

Finding the optimal decision boundary

- Let $\{x_1, \ldots, x_n\}$ be our data set and let $y_i \in \{1, -1\}$ be the class label of x_i

- The optimal decision boundary should classify all points correctly

$$\Longrightarrow \quad y_i(w^T x_i + \mathrm{b}) \geq 1, \ \ \forall\, i$$

- The decision boundary can be found by solving the following constrained optimization problem

$$\begin{aligned} &minimize \quad \frac{1}{2}\|w\|^2 \\ &subject\ to\ y_i(w^T x_i + b) \geq 1 \qquad \forall i \end{aligned}$$

Lagrangian of the optimization problem

$$\textit{minimize} \quad \frac{1}{2}\|w\|^2$$
$$\textit{subject to } y_i(w^T x_i + b) \geq 1 \qquad \forall i$$

- The Lagrangian is

$$L = \frac{1}{2} w^T w + \sum_{i=1}^{n} \alpha_i (1 - y_i (w^T x_i + b))$$

- Setting the gradient of L w.r.t. w and be to zero, we have

$$w + \sum_{i=1}^{n} \alpha_i (-y_i) x_i = 0 \Rightarrow w = \sum_{i=1}^{n} \alpha_i y_i x_i$$
$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

The Dual Problem

- If we substitute $w = \sum_{i=1}^{n} \alpha_i y_i x_i$ into Lagrangian L, we have

$$L = \frac{1}{2} \sum_{i=1}^{n} \alpha_i y_i x_i^T \sum_{j=1}^{n} \alpha_j y_j x_j + \sum_{i=1}^{n} \alpha_i \left(1 - y_i \left(\sum_{j=1}^{n} \alpha_j y_j x_j^T x_i + b \right) \right)$$

$$= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \alpha_i y_i \sum_{j=1}^{n} \alpha_j y_j x_j^T x_i - b \underbrace{\sum_{i=1}^{n} \alpha_i y_i}_{0}$$

$$= -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^{n} \alpha_i$$

- Note that $\sum_{i=1}^{n} \alpha_i y_i = 0$, and the data points appear in terms of their inner product; this is a quadratic function of α_i only.

The Dual Problem

- The dual problem is therefore:

$$\text{maxmize} \quad W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\text{subject to} \quad \alpha_i \geq 0, \qquad \sum_{i=1}^{n} \alpha_i y_i = 0$$

The Dual Problem

$$\text{minimize} \quad W(\alpha) = \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j - \sum_{i=1}^{n}\alpha_i$$

$$\text{subject to} \quad \alpha_i \geq 0, \qquad \sum_{i=1}^{n}\alpha_i y_i = 0$$

- This is a quadratic programming (QP) problem, and therefore a global minimum of α_i can always be found

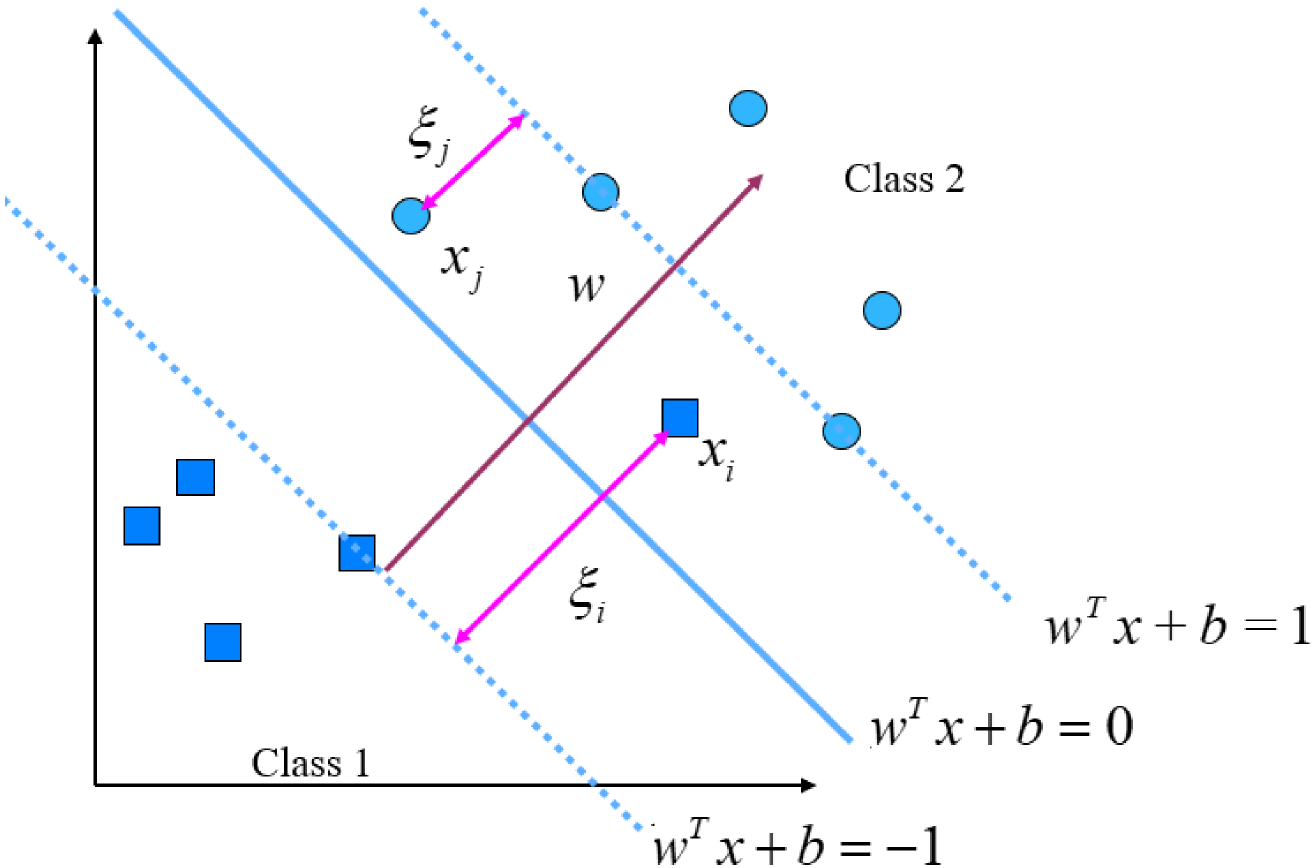
- w can be recovered by $w = \sum_{i=1}^{n}\alpha_i y_i x_i$, and

$$b = y_k - \sum_{i=1}^{n}\alpha_i y_i x_i^T x_k \qquad \text{for any} \quad \alpha_k > 0$$

- so the decision function can be written $\quad f(x) = sign\left(\sum_{i=1}^{n}\alpha_i y_i x_i^T x + b\right)$

The use of slack variables

- We allow "errors" ξ_i in classification for noisy data



Soft Margin Hyperplane

- The use of slack variables ξ_i enable the soft margin classifier

$$\begin{cases} w^T x_i + b \geq 1 - \xi_i & y_i = 1 \\ w^T x_i + b \leq -1 + \xi_i & y_i = -1 \\ \xi_i \geq 0 & \forall i \end{cases}$$

  - ξ_i are "slack variables" in optimization
  - Note that $\xi_i = 0$ if there is no error for x_i

- The objective function $$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

  C : tradeoff parameter between error and margin

- The primal optimization problem becomes

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{subject to} \quad y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

Dual Soft-Margin Optimization Problem

- The dual of this new constrained optimization problem is

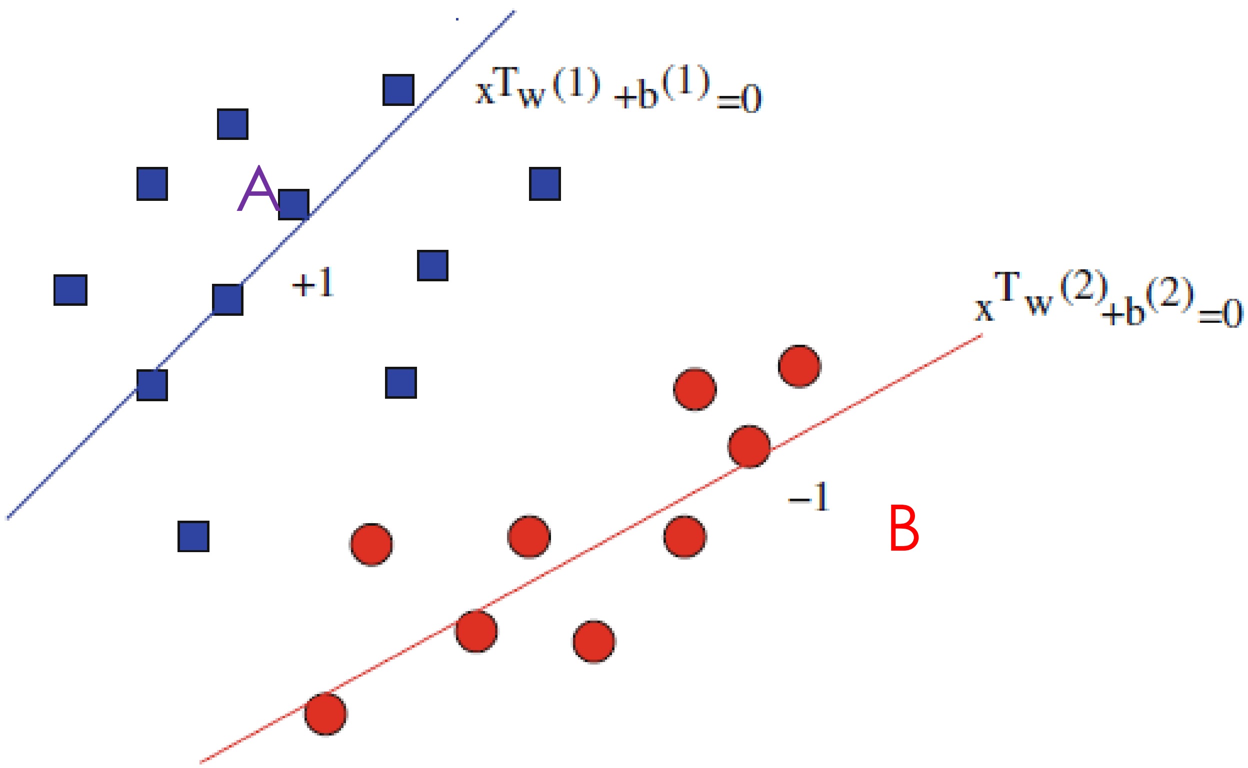
$$maxmize \;\; W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$subject \;\; to \;\; C \geq \alpha_i \geq 0, \;\; \sum_{i=1}^{n} \alpha_i y_i = 0$$

- w can be recovered as $w = \sum_{i=1}^{n} \alpha_i y_i x_i$

- This is very similar to the optimization problem in the hard-margin case, except that there is an upper bound C on α_i now.
- Once again, a QP solver can be used to find α_i

Proximal Support Vector Machine



The algorithm finds two non-parallel hyperplanes one for each class, each hyperplane should be as close as possible to one class and as far as possible from the other class.

The following decision rule can be used to allocate a new data point $x \in \mathbb{R}^n$ to the class $i \in \{+1, -1\}$

class $$i = \arg\min \frac{\left| w_k^T x + b_k \right|}{\|w_k\|^2}, \quad k = 1, 2.$$

IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 28, NO. 1, JANUARY 2006 69

Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues

Olvi L. Mangasarian and Edward W. Wild

Abstract—A new approach to support vector machine (SVM) classification is proposed wherein each of two data sets are proximal to one of two distinct planes that are *not parallel* to each other. Each plane is generated such that it is closest to one of the two data sets and as far as possible from the other data set. Each of the two nonparallel proximal planes is obtained by a single MATLAB command as the eigenvector corresponding to a smallest eigenvalue of a generalized eigenvalue problem. Classification by proximity to two distinct nonlinear surfaces generated by a nonlinear kernel also leads to two simple generalized eigenvalue problems. The effectiveness of the proposed method is demonstrated by tests on simple examples as well as on a number of public data sets. These examples show the advantages of the proposed approach in both computation time and test set correctness.

Index Terms—Support vector machines, proximal classification, generalized eigenvalues.

1 Introduction

Support vector machines (SVMs) [23], [4], [27] constitute the method of choice for classification problems while the generalized eigenvalue problem [22], [5] is a simple problem of classical linear algebra solvable by a single command of MATLAB [17] or Scilab [24] or by using standard linear algebra software such LAPACK [1]. In proximal support vector classification [7], [25], [6], two *parallel* planes are generated such that each plane is closest to one of two data sets

variation and maximizing between-class variation of various protein folds.

This work is organized as follows: In Section 2, we briefly describe the general classification problem and our proximal multiplane linear kernel formulation as a generalized eigenvalue problem. In Section 3, we extend our proximal results to a proximal multisurface nonlinear kernel formula-

$$\min \frac{\|AW^1 + b^1\|}{\|BW^1 + b^1\|}$$

$$\min \frac{\|BW^2 + b^2\|}{\|AW^2 + b^2\|}$$

We introduce the Tikhonov regularization term, a widely-utilized technique for least squares and mathematical programming problems. This regularization diminishes the norm of the problem variables (w, b), which determine the proximal planes. Consequently, by introducing a nonnegative parameter δ, we modify our problems as follows:

$$\min \frac{\|AW^1 + b^1\| + \delta\|(W^1, b^1)\|}{\|BW^1 + b^1\|}$$

$$\min \frac{\|BW^2 + b^2\| + \delta\|(W^2, b^2)\|}{\|AW^2 + b^2\|}$$

$$G := [A \quad -e]'[A \quad -e] + \delta I,$$

$$H := [B \quad -e]'[B \quad -e], z := \begin{bmatrix} w \\ \gamma \end{bmatrix},$$

$$\min_{z \neq 0} r(z) := \frac{z'Gz}{z'Hz},$$

where G and H are symmetric matrices . The objective function is known as the Rayleigh quotient.

Theorem. (Rayleigh Quotient properties).

Let G and H be arbitrary symmetric matrices in $R^{(n+1)\times(n+1)}$. When H in positive definite, the Rayleigh quotient of (7) enjoys the following properties:

1. (**Boundedness**) The Rayleigh quotient ranges over the interval $[\lambda_1, \lambda_{n+1}]$ as Z ranges over the unit sphere, where λ_1 *and* λ_{n+1} are the minimum and maximum eigenvalues of the generalized eigenvalue

$$Gz = \lambda Hz, \quad z \neq 0.$$

2. (**stationarity**)

$$\nabla r(z) = 2\frac{(Gz - r(z)Hz)}{z'Hz} = 0$$

Thus, r(z) is stationary at and only at the eigenvectors of the above generalized eigenvalue problem.

Optimization Methods and Software
Vol. 22, No. 1, February 2007, 73–81

Taylor & Francis
Taylor & Francis Group

A classification method based on generalized eigenvalue problems

M. R. GUARRACINO*†, C. CIFARELLI‡, O. SEREF§ and P. M. PARDALOS§

†High Performance Computing and Networking Institute, National Research Council, Italy
‡Department of Statistic, Probability and Applied Statistics, University of Rome 'La Sapienza', Italy
§Center for Applied Optimization, University of Florida, Gainesville, FL, 32611-6595, USA

(Received 14 July 2005; revised 12 April 2006; in final form 18 May 2006)

Binary classification refers to supervised techniques that split a set of points in two classes, with respect to a training set of points whose membership is known for each class. Binary classification plays a central role in the solution of many scientific, financial, engineering, medical and biological problems. Many methods with good classification accuracy are currently available. This work shows how a binary classification problem can be expressed in terms of a generalized eigenvalue problem. A new regularization technique is proposed, which gives results that are comparable to other techniques in use, in terms of classification accuracy. The advantage of this method relies in its lower computational complexity with respect to the existing techniques based on generalized eigenvalue problems. Finally, the method is compared with other methods using benchmark data sets.

Keywords: Classification; Binary classification; Generalized Eigenvalue problem

$$\min \frac{\|AW^1 + b^1\| + \delta\|(W^1, b^1)\|}{\|BW^1 + b^1\|}$$

$$\min \frac{\|BW^2 + b^2\| + \delta\|(W^2, b^2)\|}{\|AW^2 + b^2\|}$$

THEOREM . *Consider the generalized eigenvalue problem $Gx = \lambda Hx$ and the transformed $G^*x = \lambda H^*x$ defined by*

$$G^* = \tau_1 G - \delta_1 H, \qquad H^* = \tau_2 H - \delta_2 G$$

for each choice of scalars τ_1, τ_2, δ_1 and δ_2, such that the 2×2 matrix

$$\Omega = \begin{pmatrix} \tau_2 & \delta_1 \\ \delta_1 & \tau_1 \end{pmatrix}$$

is nonsingular. Then the problem $G^*x = \lambda H^*x$ has the same eigenvectors of the problem $Gx = \lambda Hx$. An associated eigenvalue λ^* of the transformed problem is related to an eigenvalue λ of the original problem by

$$\lambda = \frac{\tau_2 \lambda^* + \delta_1}{\tau_1 + \delta_2 \lambda^*}$$

In the linear case Theorem can be applied. By setting $\tau_1 = \tau_2 = 1$ and $\hat{\delta}_1 = \delta_1, \hat{\delta}_2 = -\delta_2$, the regularized problem becomes

$$\min_{w,\gamma \neq 0} \frac{\|Aw - e\gamma\|^2 + \hat{\delta}_1 \|Bw - e\gamma\|^2}{\|Bw - e\gamma\|^2 + \hat{\delta}_2 \|Aw - e\gamma\|^2}.$$

If $\hat{\delta}_1$ and $\hat{\delta}_2$ are non-negative, Ω is non-degenerate. The spectrum is now shifted and inverted so that the minimum eigenvalue of the original problem becomes the maximum of the regularized one, and the maximum becomes the minimum eigenvalue. Choosing the eigenvectors related to the new minimum and maximum eigenvalue, we still obtain the same ones of the original problem.

Expert Systems With Applications 42 (2015) 9183–9191

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Sparse Proximal Support Vector Machines for feature selection in high dimensional datasets

Vijay Pappu [a], Orestis P. Panagopoulos [b], Petros Xanthopoulos [b,*], Panos M. Pardalos [a]

[a] *Department of Industrial Engineering, University of Florida, 401 Weil Hall, Gainesville, FL 32608, USA*
[b] *Department of Industrial Engineering and Management Systems, University of Central Florida, 12800 Pegasus Dr., P.O. Box 162993, Orlando, FL 32816, USA*

A R T I C L E I N F O

Keywords:
Embedded feature selection
Sparsity
Regularization
Class-specific feature selection
High dimensional datasets

A B S T R A C T

Classification of High Dimension Low Sample Size (HDLSS) datasets is a challenging task in supervised learning. Such datasets are prevalent in various areas including biomedical applications and business analytics. In this paper, a new embedded feature selection method for HDLSS datasets is introduced by incorporating sparsity in Proximal Support Vector Machines (PSVMs). Our method, called Sparse Proximal Support Vector Machines (sPSVMs), learns a sparse representation of PSVMs by first casting it as an equivalent least squares problem and then introducing the l_1-norm for sparsity. An efficient algorithm based on alternating optimization techniques is proposed. sPSVMs remove more than 98% of features in many high dimensional datasets without compromising on generalization performance. Stability in the feature selection process of sPSVMs is also studied and compared with other univariate filter techniques. Additionally, sPSVMs offer the advantage of interpreting the selected features in the context of the classes by inducing class-specific *local* sparsity instead of *global* sparsity like other embedded methods. sPSVMs appear to be robust with respect to data dimensionality. Moreover, sPSVMs are able to perform feature selection and classification in one step, elimi-

Classification of High Dimension Low Sample Size (HDLSS) datasets is a challenging task in supervised learning. Such datasets are prevalent in various areas including biomedical applications and business analytics. In this paper, a new embedded feature selection method for HDLSS datasets is introduced by incorporating sparsity in Proximal Support Vector Machines (PSVMs).

Theorem . *Consider a real matrix* $\boldsymbol{X} \in \Re^{n\times p}$ *with rank* $r \leq \min(n, p)$. *Let matrices* $\boldsymbol{V} \in \Re^{p\times p}$ *and* $\boldsymbol{D} \in \Re^{p\times p}$ *satisfy the following relation:*

$$\boldsymbol{V}^T(\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{V} = \boldsymbol{D}$$

where, $\boldsymbol{D} = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_r^2, 0, 0, \ldots, 0)_{p\times p}$. *Assume* $\sigma_1^2 \geq \sigma_2^2 \geq \cdots \geq \sigma_r^2$. *For the following optimization problem,*

$$\underset{\boldsymbol{\alpha}, \hat{\boldsymbol{\beta}} \in \Re^p}{\text{minimize}} \quad \|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{\alpha}\hat{\boldsymbol{\beta}}^T\|_F^2 + \mu\hat{\boldsymbol{\beta}}^T\hat{\boldsymbol{\beta}}$$

$$\text{subject to} \quad \boldsymbol{\alpha}^T\boldsymbol{\alpha} = 1$$

$\hat{\boldsymbol{\beta}}_{opt}$ *is proportional to* $\boldsymbol{v}_1$, *where* $\boldsymbol{v}_1$ *is the eigenvector corresponding to the largest eigenvalue* σ_1^2 *and* $\mu \in \Re_+$.

Using Theorem , we now establish that the proximal hyperplanes P_1 and P_2 can be obtained via the least-squares approach. Let the Cholesky decomposition of the matrices $\boldsymbol{H}_2$ and $\boldsymbol{G}_1$ be given by:

$$\boldsymbol{H}_2 = \boldsymbol{U}_2^T\boldsymbol{U}_2, \quad \boldsymbol{G}_1 = \boldsymbol{U}_1^T\boldsymbol{U}_1 \tag{1}$$

where $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are upper triangular matrices.

Using (1) in $\mathbf{GEV}(\boldsymbol{H}_2, \boldsymbol{G}_1)$,

$$\boldsymbol{U}_2^T\boldsymbol{U}_2\boldsymbol{z} = \lambda\boldsymbol{U}_1^T\boldsymbol{U}_1\boldsymbol{z}(\boldsymbol{U}_2\boldsymbol{U}_1^{-1})^T(\boldsymbol{U}_2\boldsymbol{U}_1^{-1})\boldsymbol{U}_1\boldsymbol{z} = \lambda\boldsymbol{U}_1\boldsymbol{z}$$

$$(\boldsymbol{U}_2\boldsymbol{U}_1^{-1})^T(\boldsymbol{U}_2\boldsymbol{U}_1^{-1})\boldsymbol{y} = \lambda\boldsymbol{y} \tag{2}$$

where $\boldsymbol{U}_1\boldsymbol{z} = \boldsymbol{y}$.

The optimal eigenvector corresponding to proximal hyperplane P_1 can be found by the following relation:

$$\boldsymbol{z}_{opt} = \boldsymbol{U}_1^{-1}\hat{\boldsymbol{y}}$$

where $\hat{\boldsymbol{y}}$ is the eigenvector corresponding to the maximum eigenvalue of the symmetric eigenvalue problem given in (2).

By substituting $\boldsymbol{X} = \boldsymbol{U}_2\boldsymbol{U}_1^{-1}$, $\hat{\boldsymbol{\beta}} = \boldsymbol{U}_1\boldsymbol{\beta}$, and re-arranging the terms, the following least-squares optimization problem is obtained:

$$\begin{aligned} \underset{\boldsymbol{\alpha},\boldsymbol{\beta}}{\text{minimize}} \quad & \|\boldsymbol{U}_2\boldsymbol{U}_1^{-1} - \boldsymbol{U}_2\boldsymbol{\beta}\boldsymbol{\alpha}^T\|_F^2 + \mu\boldsymbol{\beta}^T\boldsymbol{G}_1\boldsymbol{\beta} \\ \text{subject to} \quad & \boldsymbol{\alpha}^T\boldsymbol{\alpha} = 1 \end{aligned} \tag{4}$$

By Theorem , the optimal solution for (4) $\boldsymbol{\beta}_{opt}$ is proportional to $\boldsymbol{z}_1$, the eigenvector corresponding to the largest eigenvalue of **GEV**($\boldsymbol{H}_2$, $\boldsymbol{G}_1$).

The following algorithm summarizes the steps needed to solve for the optimal hyperplane P_1 in PSVMs using the least squares (LS) approach:

Similarly, the hyperplane P_2 can be obtained from Algorithm 1 with the input parameters $(\boldsymbol{H}_1, \boldsymbol{G}_2)$.

Algorithm 1 PSVMs-via-LS $(\boldsymbol{H}_2, \boldsymbol{G}_1)$.

1. Initialize $\boldsymbol{\beta}$.
2. Find the upper triangular matrix $\boldsymbol{U}_1$ from the Cholesky decomposition of $\boldsymbol{G}_1$.
3. Find $\boldsymbol{\alpha}$ from the following relation:

$$\boldsymbol{\alpha} = \frac{\boldsymbol{U}_1^{-T}\boldsymbol{H}_2\boldsymbol{\beta}}{\|\boldsymbol{U}_1^{-T}\boldsymbol{H}_2\boldsymbol{\beta}\|}$$

4. Find $\boldsymbol{\beta}$ as follows:

$$\boldsymbol{\beta} = (\boldsymbol{H}_2 + \mu\boldsymbol{G}_1)^{-1}\boldsymbol{H}_2\boldsymbol{U}_1^{-1}\boldsymbol{\alpha}$$

5. Alternate between 3 and 4 until convergence.

Sparsity is induced in PSVMs by adding an l1-norm term to the objective function given in (4). The resulting optimization problem is given by:

$$\underset{\boldsymbol{\alpha},\boldsymbol{\beta}}{\text{minimize}} \quad \|\boldsymbol{U}_2\boldsymbol{U}_1^{-1} - \boldsymbol{U}_2\boldsymbol{\beta}\boldsymbol{\alpha}^T\|_F^2 + \mu\boldsymbol{\beta}^T\boldsymbol{G}_1\boldsymbol{\beta} + \delta\|\boldsymbol{\beta}\|_1$$

$$\text{subject to} \quad \boldsymbol{\alpha}^T\boldsymbol{\alpha} = 1$$

where the parameter δ controls the level of sparsity in the coefficient vector $\boldsymbol{\beta}$.

Algorithm 2 sPSVMs $(\boldsymbol{H}_2, \boldsymbol{G}_1)$.

1. Initialize $\boldsymbol{\beta}$
2. Find $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ that satisfy,

$$\boldsymbol{G}_1 = \boldsymbol{U}_1^T \boldsymbol{U}_1, \quad \boldsymbol{H}_2 = \boldsymbol{U}_2^T \boldsymbol{U}_2$$

3. Find $\boldsymbol{\alpha}$ from the following equation:

$$\boldsymbol{\alpha} = \frac{\boldsymbol{U}_1^{-T} \boldsymbol{H}_2 \boldsymbol{\beta}}{\|\boldsymbol{U}_1^{-T} \boldsymbol{H}_2 \boldsymbol{\beta}\|}$$

4. Solve the following LASSO regression problem to obtain $\boldsymbol{\beta}$:

$$\underset{\boldsymbol{\beta}}{\text{minimize}} \quad \|\boldsymbol{y} - \boldsymbol{W}\boldsymbol{\beta}\|^2 + \delta \|\boldsymbol{\beta}\|_1$$

where W and y are given by:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{U}_2 \\ \sqrt{\mu}\boldsymbol{U}_1 \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{U}_2 \boldsymbol{U}_1^{-1} \boldsymbol{\alpha} \\ 0 \end{bmatrix}$$

5. Alternate between 3 and 4 until convergence.

Twin Support Vector Machines (TWSVM)

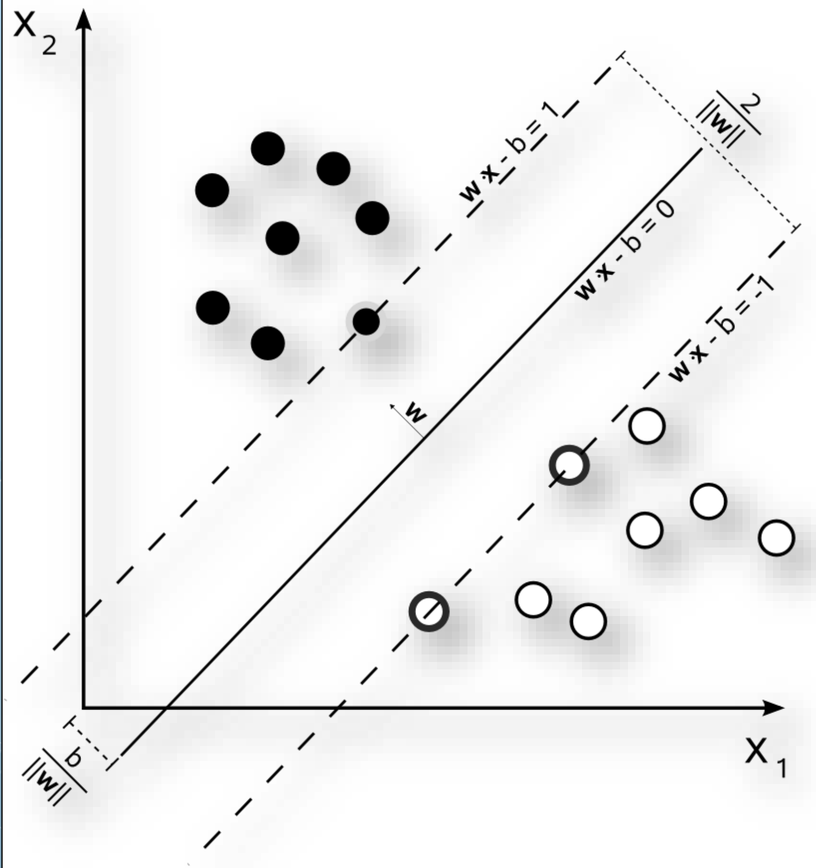
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE,

Twin Support Vector Machines for Pattern Classification

Jayadeva, *Senior Member, IEEE*,
R. Khemchandani, *Student Member, IEEE*,
and
Suresh Chandra

Abstract—We propose Twin SVM, a binary SVM classifier that determines two nonparallel planes by solving two related SVM-type problems, each of which is smaller than in a conventional SVM. The Twin SVM formulation is in the spirit of proximal SVMs via generalized eigenvalues. On several benchmark data sets, Twin SVM is not only fast, but shows good generalization. Twin SVM is also useful for automatically discovering two-dimensional projections of the data.

Standard SVM :



$$\min_{w,r} \frac{1}{2} w^T w + \nu e^T r,$$

$$\textit{subject to}\ \ (Aw - e\gamma) + r \geq e,$$

$$(Bw - e\gamma) - r \leq -e,$$

$$r \geq 0.$$

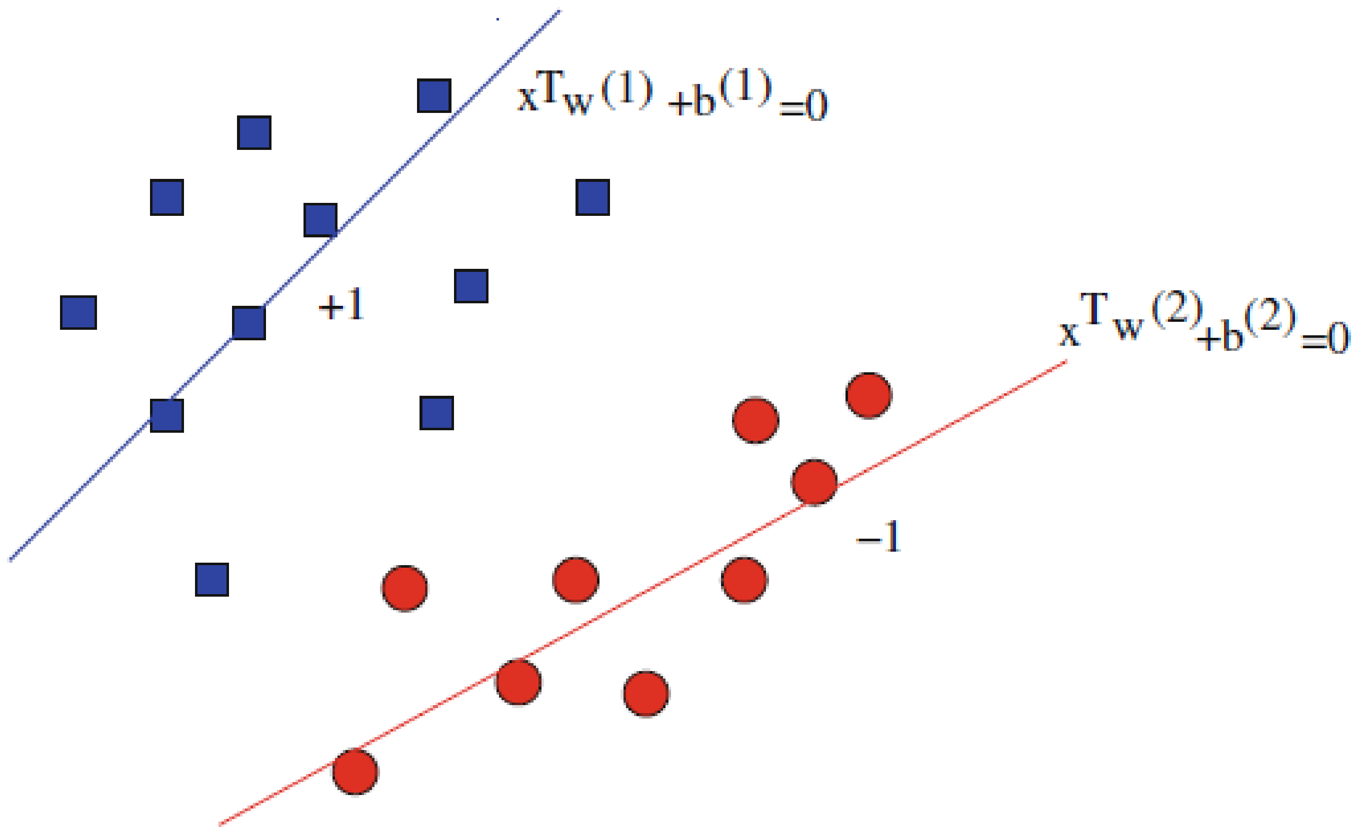
Why TWSVM?

This quadratic programming problem (QPP) is expensive to solve for large dimensions because all data points appear in the constraints.

How does it works ?

Instead of solving one large QPP, TWSVM solve two smaller QPP each of them has the formulation of standard SVM except that not all data patterns appear in the constraint at the same time.

The algorithm finds two non-parallel hyperplanes one for each class, each hyperplane should be as close as possible to one class and as far as possible from the other class.


$x^T w^{(1)} + b^{(1)} = 0$
+1
$x^T w^{(2)} + b^{(2)} = 0$
−1

Linear Classifier

TWSVM is obtained by solving the following pair of QPPs:

$$(TWSVM1) \quad \underset{w^{(1)},\, b^{(1)},\, q}{Min} \quad \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) + c_1 e_2^T q$$

$$subject\ to \quad -(Bw^{(1)} + e_2 b^{(1)}) + q \geq e_2, \quad q \geq 0,$$

$$(TWSVM2) \quad \underset{w^{(2)},\, b^{(2)},\, q}{Min} \quad \frac{1}{2}(Bw^{(2)} + e_2 b^{(2)})^T (Bw^{(2)} + e_2 b^{(2)}) + c_2 e_1^T q$$

$$subject\ to \quad (Aw^{(2)} + e_1 b^{(2)}) + q \geq e_1, \quad q \geq 0,$$

The first term of the objective function represents the sum of square distance from the hyperplane to each pattern of one class, therefore minimizing it keeps the hyperplane close to the patterns of one class.

The constraints require the hyper plane to be far from the other class patterns at least with distance 1.

The second term of the objective function minimize the sum of error variables to minimize miss classification of patterns belongs to other class.

The Wolfe dual can be obtain as follows

$$\max_{\alpha} e_2^T\alpha - \frac{1}{2}\alpha^T G^{(H^T H)^{-1}} G^T\alpha, \quad G = [B \quad e_2] \quad and \quad H = [A \quad e_1]$$

$$subject\ to\ \ 0 \le \alpha \le c_1$$

$$u = -(H^T H)^{-1} G^T\alpha \quad where \quad u = [w_1^T, \qquad b_1]^T.$$

$$\max_{\alpha} e_1^T\gamma - \frac{1}{2}\gamma^T P^{(Q^T Q)^{-1}} P^T\gamma, \quad P = [A \quad e_1] \quad and \quad Q = [B \quad e_2]$$

$$subject\ to\ \ 0 \le \gamma \le c_2$$

$$v = (Q^T Q)^{-1} P^T\gamma \quad where \quad v = [w_2^T, \qquad b_2]^T$$

Inference with the Universum

Jason Weston JASONW@NEC-LABS.COM
Ronan Collobert RONAN@COLLOBERT.COM
NEC Labs America, Princeton NJ, USA.

Fabian Sinz FABEE@TUEBINGEN.MPG.DE
NEC Labs America, Princeton NJ, USA; and
Max Planck Insitute for Biological Cybernetics, Tuebingen, Germany.

Léon Bottou LEON@BOTTOU.ORG
Vladimir Vapnik VLAD@NEC-LABS.COM
NEC Labs America, Princeton NJ, USA.

Abstract

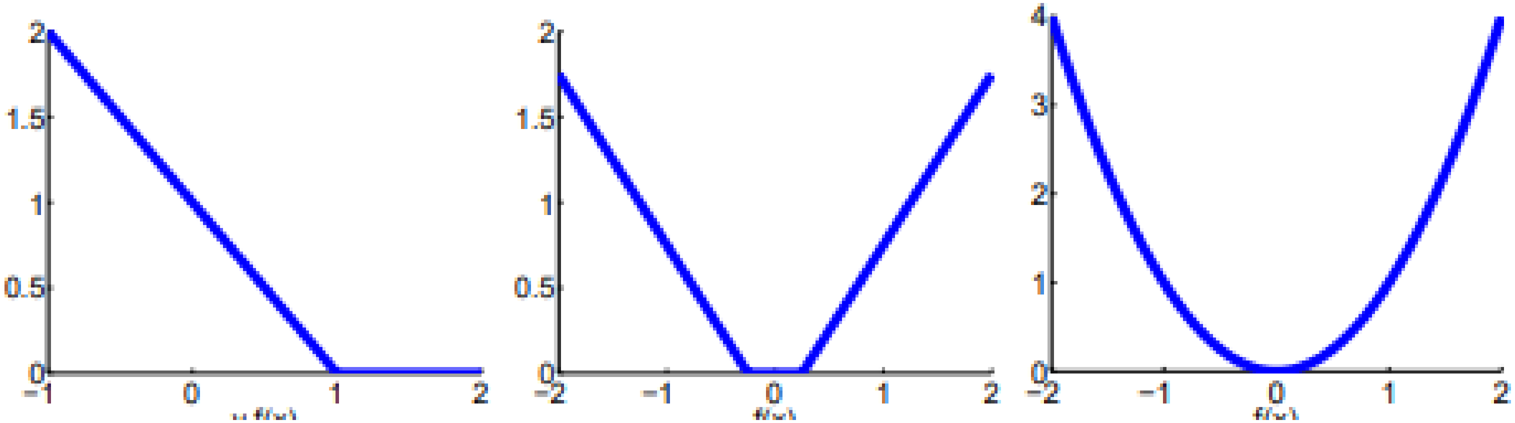
In this paper we study a new framework introduced by Vapnik (1998) and Vapnik (2006) that is an alternative capacity concept to the large margin approach. In the particular case of binary classification, we are given a set of labeled examples, and a collection of "non-examples" that do not belong

not to belong to either class

$$x_1^*, \ldots, x_{|\mathfrak{U}|}^*, \quad x^* \in R^d \tag{1}$$

The set $\mathfrak{U}$ is called the *Universum*. It contains data that belongs to the *same domain* as the problem of interest and is expected to represent meaningful information related to the pattern recognition task at hand.

Figure 1. From left to right, the Hinge loss and the ε-insensitive and L_2 losses. The ε-insensitive loss is a linear combination $U[t] = H_{-\varepsilon}[t] + H_{-\varepsilon}[-t]$ of two Hinge loss functions $H_{-\varepsilon}[t] = \max\{0, t - \varepsilon\}$. Here it is shown with $\varepsilon = 0.25$. The L_2 loss is a simple quadratic function.



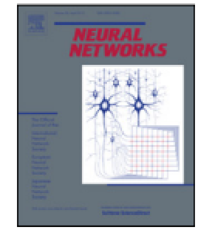
Twin support vector machine with universum data (UTSVM)

Neural Networks 36 (2012) 112–119

Contents lists available at SciVerse ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Twin support vector machine with Universum data

Zhiquan Qi [a], Yingjie Tian [a,*], Yong Shi [a,b,**]

[a] Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing 100190, China

[b] College of Information Science & Technology, University of Nebraska at Omaha, Omaha, NE 68182, USA

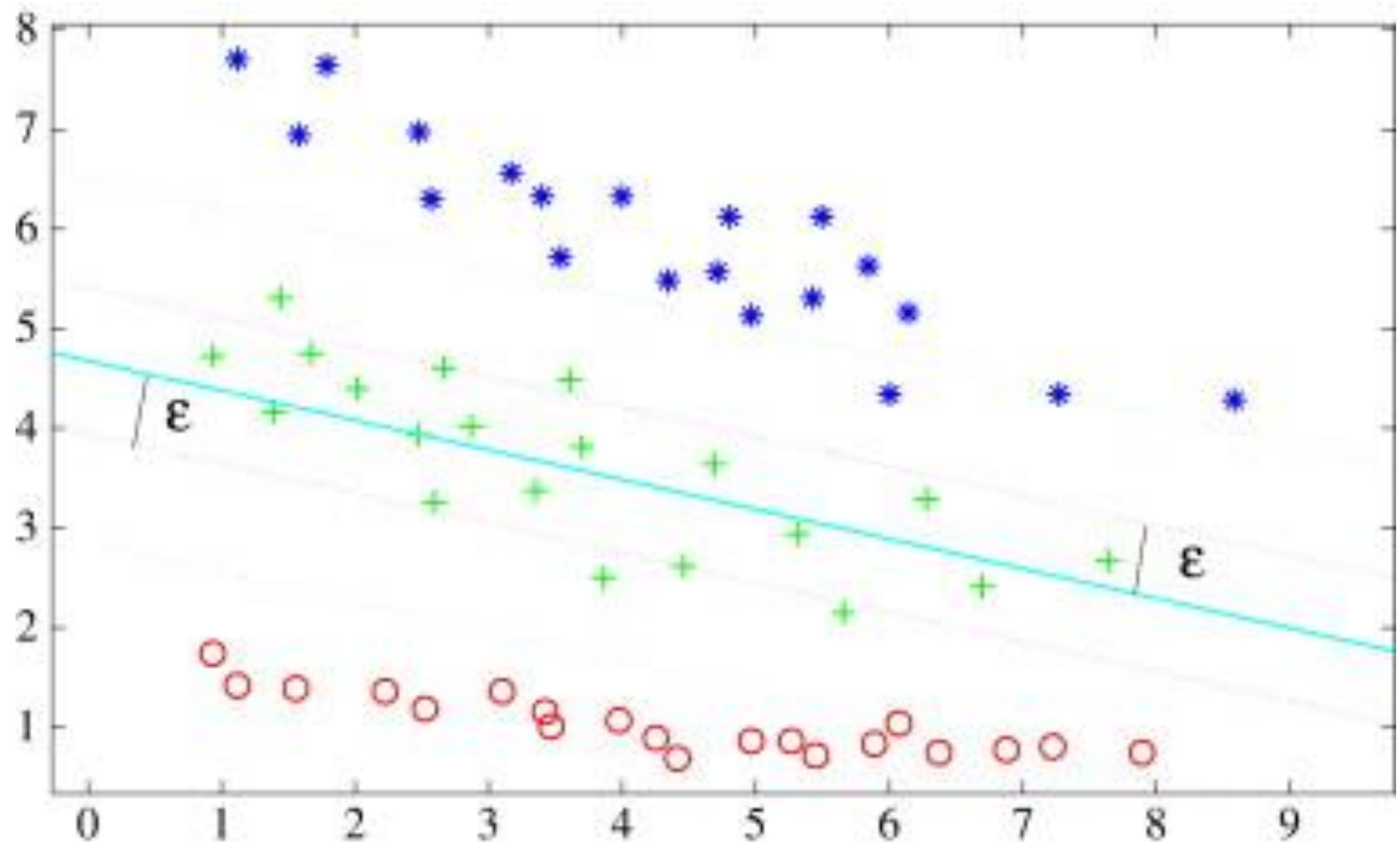
ARTICLE INFO

Article history:
Received 30 May 2012
Received in revised form 20 August 2012
Accepted 3 September 2012

Keywords:
Classification
Twin support vector machine
Universum

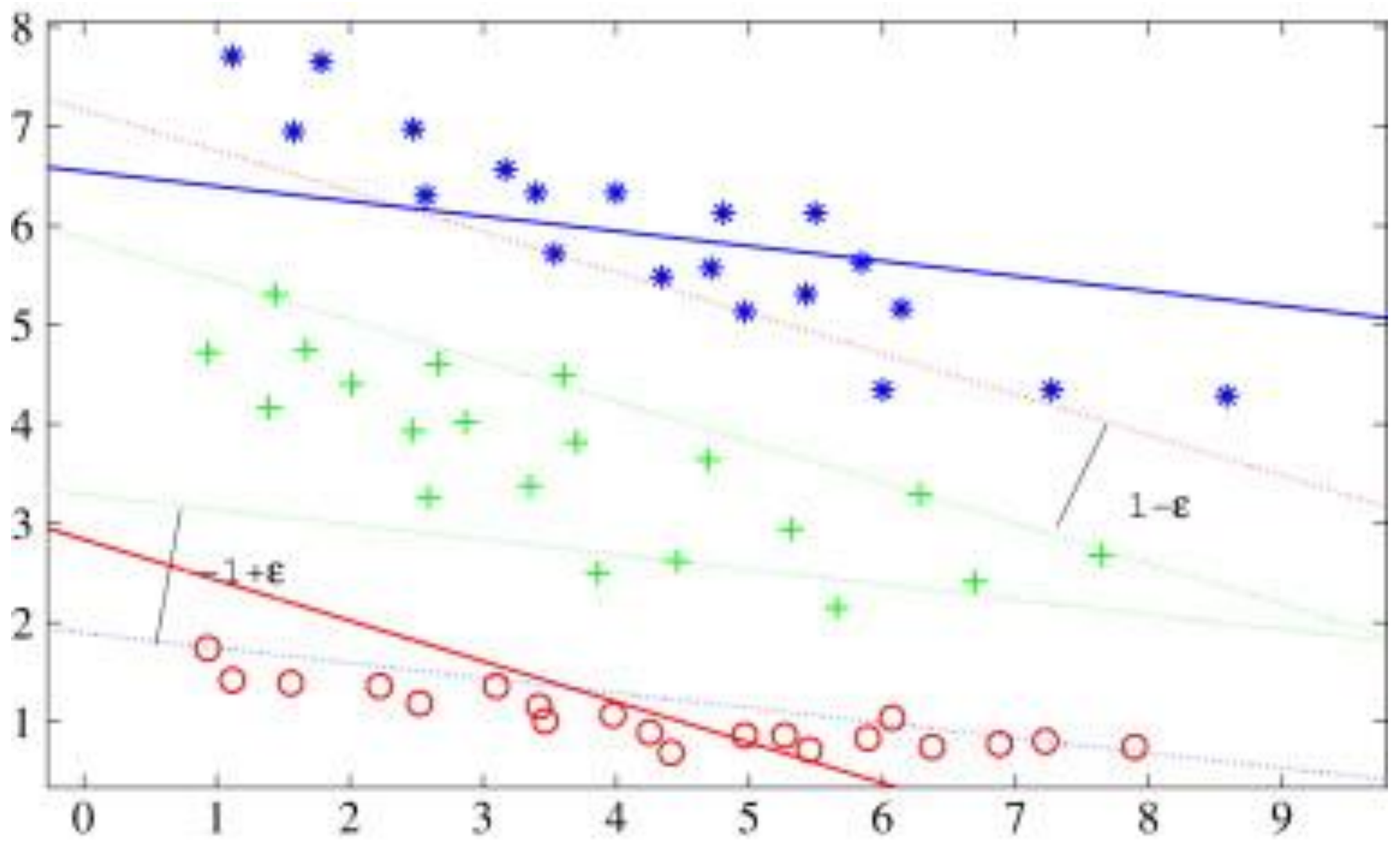
ABSTRACT

The Universum, which is defined as the sample not belonging to either class of the classification problem of interest, has been proved to be helpful in supervised learning. In this work, we designed a new Twin Support Vector Machine with Universum (called $\mathfrak{U}$-TSVM), which can utilize Universum data to improve the classification performance of TSVM. Unlike $\mathfrak{U}$-SVM, in $\mathfrak{U}$-TSVM, Universum data are located in a nonparallel insensitive loss tube by using two Hinge Loss functions, which can exploit these prior knowledge embedded in Universum data more flexible. Empirical experiments demonstrate that $\mathfrak{U}$-TSVM can directly improve the classification accuracy of standard TSVM that use the labeled data alone and is superior to $\mathfrak{U}$-SVM in most cases.



Support vector machine with Universum data

(USVM)



Twin support vector machine with Universum data (UTSVM)

Twin bounded support vector machine with universum data (UTBSVM)

- Training data $\tilde{T}$ ：

$$\tilde{T} = T \bigcup U,$$

$$T = \{(\mathrm{x}_1, \mathrm{y}_1), \ldots, (\mathrm{x}_n, \mathrm{y}_n)\} \in (\mathrm{R}^m \times \{\pm 1\})^n,$$

$$U = \{\mathrm{x}_1^*, \ldots, \mathrm{x}_u^*\}.$$

Here, $U \in R^{u \times m}$ denotes the universum class, and each row of the matrix U represents an universum sample.

- Learning the UTBSVM can be formulated as an optimization:

$$\begin{aligned}
\min_{w_1, b_1, \xi_1, \psi_1} \quad & \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 + \frac{c_1}{2} e_2^t \xi_1 + \frac{c_2}{2}(\|w_1\|^2 + b_1^2) + \frac{c_3}{2} e_u^t \psi_1 \\
s.t. \quad & -(Bw_1 + e_2 b_1) + \xi_1 \geq e_2, \\
& (Uw_1 + e_u b_1) + \psi_1 \geq (-1 + \varepsilon) e_u, \\
& \xi_1, \psi_1 \geq 0,
\end{aligned} \tag{1}$$

$$\begin{aligned}
\min_{w_2, b_2, \xi_2, \psi_2} \quad & \frac{1}{2}\|Bw_2 + e_2 b_2\|^2 + \frac{c_4}{2} e_1^t \xi_2 + \frac{c_2}{2}(\|w_2\|^2 + b_2^2) + \frac{c_6}{2} e_u^t \psi_2 \\
s.t. \quad & (Aw_2 + e_1 b_2) + \xi_2 \geq e_1, \\
& -(Uw_2 + e_u b_2) + \psi_2 \geq (-1 + \varepsilon) e_u, \\
& \xi_2, \psi_2 \geq 0,
\end{aligned} \tag{2}$$

Challenges

Twin Support Vector Machines (TSVM) and

Sparse Optimization for Feature Selection

Expert Systems With Applications 248 (2024) 123378

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

ELSEVIER

Sparse least-squares Universum twin bounded support vector machine with adaptive L_p-norms and feature selection

Hossein Moosaei [a,c,*], Fatemeh Bazikar [b], Milan Hladík [c], Panos M. Pardalos [d,e]

[a] *Department of Informatics, Faculty of Science, Jan Evangelista Purkyně University, Ústí nad Labem, Czech Republic*
[b] *Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran*
[c] *Department of Applied Mathematics, School of Computer Science, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic*
[d] *Center for Applied Optimization, Department of Industrial and Systems Engineering, University of Florida, Gainesville, 32611, USA*
[e] *LATNA, Higher School of Economics, Russia*

ARTICLE INFO

Keywords:
Universum
Twin bounded support vector machine
Least-squares twin bounded support vector machine with Uiversum
p-norm
Feature selection

ABSTRACT

In data analysis, when attempting to solve classification problems, we may encounter a large number of features. However, not all features are relevant for the current classification, and including irrelevant features can occasionally degrade learning performance. As a result, selecting the most relevant features is critical, especially for high-dimensional data sets in classification problems. Feature selection is an effective method for resolving this issue. It attempts to represent the original data by extracting relevant features containing useful information. In this research, our aim is to propose a *p*-norm least-squares Universum twin bounded support vector machine (LS_p-UTBSVM) to perform classification and feature selection at the same time. Indeed,

$$\min_{w_1,b_1\xi_1,\psi} \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 + \frac{c_1}{2}\|\xi_1\|^2 + \frac{c_3}{2}\left(\|w_1\|_p^p + b_1^2\right) + \frac{c_u}{2}\|\psi\|^2$$

$$\text{s.t.} \quad -\left(Bw_1 + e_2 b_1\right) + \xi_1 = e_2,$$

$$(Uw_1 + e_u b_1) + \psi = (-1+\varepsilon)e_u, \qquad (1)$$

$$\min_{w_2,b_2,\xi_2,\psi^*} \frac{1}{2}\|Bw_2 + e_2 b_2\|^2 + \frac{c_2}{2}\|\xi_2\|^2 + \frac{c_4}{2}\left(\|w_2\|_p^p + b_2^2\right) + \frac{c_u^*}{2}\|\psi^*\|^2$$

$$\text{s.t.} \quad \left(Aw_2 + e_1 b_2\right) + \xi_2 = e_1,$$

$$-(Uw_2 + e_u b_2) + \psi^* = (-1+\varepsilon)e_u. \qquad (2)$$

We reformulate problems (1) and (2) to the following unconstrained optimization problems

$$\min_{w_1,b_1} \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 + \frac{c_1}{2}\|e_2 + (Bw_1 + e_2 b_1)\|^2 + \frac{c_3}{2}(\|w_1\|_p^p + b_1^2) + \frac{c_u}{2}\|(-1+\varepsilon)e_u - (Uw_1 + e_u b_1)\|^2, \tag{3}$$

$$\min_{w_2,b_2} \frac{1}{2}\|Bw_2 + e_2 b_2\|^2 + \frac{c_2}{2}\|e_1 - (Aw_2 + e_1 b_2)\|^2 + \frac{c_4}{2}(\|w_2\|_p^p + b_2^2) + \frac{c_u^*}{2}\|(-1+\varepsilon)e_u + (Uw_2 + e_u b_2)\|^2. \tag{4}$$

Here, we find lower bounds for the absolute values of non-zero components of the optimal solution. More precisely, we find such lower and upper bounds that each component of the optimal solution lying inside the bounds must be 0.

Theorem. Let (w_1^*, b_1^*) be a local optimal solution of problem (1). Then $w_{1i}^* = 0$ if $w_{1i}^* \in (-I_i, I_i)$, where

$$I_i = \left[\frac{\frac{c_3}{2}p(1-p)}{e_i^T \tilde{A}^T \tilde{A} e_i + c_1 e_i^T \tilde{B}^T \tilde{B} e_i + c_u e_i^T \widetilde{U}^T \widetilde{U} e_i p}\right]^{\frac{1}{2-p}}, \qquad i = 1,2,\dots,n,$$

e_i is the ith column of the identity matrix, $\tilde{A}$ is a submatrix of A composed of the columns corresponding to the non-zero components of w_1^* and, $\tilde{B}$ and $\widetilde{U}$ can be described analogously.

Theorem. Assume (w_2^*, b_2^*) is a local optimal solution of problem (2). If $w_{2i}^* \in (-E_i, E_i)$, where

$$E_i = \left[\frac{\frac{c_4}{2}p(1-p)}{e_i^T\tilde{B}^T\tilde{B}e_i + c_1 e_i^T\tilde{A}^T\tilde{A}e_i + c_u^* e_i^T\widetilde{U}^T\widetilde{U}e_i}\right]^{\frac{1}{2-p}}, \qquad i = 1,2,\dots,n,$$

Then $w_{2i}^* = 0$.

Not that the terms $||w_1||_p^p$ and $||w_2||_p^p$ in the objective functions not only are non-smooth, but also are the sources of non-convexity for problems (1) and (2) and also (3) and (4). So, it is not an easy task to obtain the global solutions of these problems. To resolve the issue of non-smooth terms, we approximate $||w_1||_p^p$ $= \sum_{i=1}^{n} |w_{1i}|^p$ by $\sum_{i=1}^{n} (|w_{1i}| + \varepsilon_0)^p$

and $||w_2||_p^p = \sum_{i=1}^{n} |w_{2i}|^p$ by $\sum_{i=1}^{n} (|w_{2i}| + \varepsilon_0)^p$, where ε_0>0 is a very small number. Therefor, the problems (3) and (4) are differentiable.

But, because of the terms $\sum_{i=1}^{n}(|w_{1i}| + \varepsilon_0)^p$ and $\sum_{i=1}^{n}(|w_{2i}| + \varepsilon_0)^p$ for $0 < p < 1$, the problems (3) and (4) are still non-convex. To overcome this defect, the non-convex terms $\sum_{i=1}^{n}(|w_{1i}| + \varepsilon_0)^p$ and $\sum_{i=1}^{n}(|w_{2i}| + \varepsilon_0)^p$ are replaced by the convex terms $\|\beta \otimes w_1\|_2^2$ and $\left\|\tilde{\beta} \otimes w_2\right\|_2^2$, where β and $\tilde{\beta}$ can be adjusted to fit the approximation.

So, we obtain the convex programming problems

$$\min_{w_1,b_1} \frac{1}{2}\|Aw_1 + e_1b_1\|^2 + \frac{c_1}{2}\|e_2 + (Bw_1 + e_2b_1)\|^2 + \frac{c_3}{2}(\|\beta \otimes w_1\|^2 + b_1^2) + \frac{c_u}{2}\|(-1+\varepsilon)e_u - (Uw_1 + e_ub_1)\|^2, \quad (5)$$

and

$$\min_{w_2,b_2} \frac{1}{2}\|Bw_2 + e_2b_2\|^2 + \frac{c_2}{2}\|e_1 - (Aw_2 + e_1b_2)\|^2 + \frac{c_4}{2}(\|\tilde{\beta} \otimes w_2\|^2 + b_2^2) + \frac{c_u^*}{2}\|(-1+\varepsilon)e_u + (Uw_2 + e_ub_2)\|^2. \quad (6)$$

The problems (5) and (6) can be solved by solving a systems of equations.

Twin Support Vector Machines (TSVM) and Multi-task learning

Annals of Mathematics and Artificial Intelligence
https://doi.org/10.1007/s10472-023-09877-8

Check for updates

An improved multi-task least squares twin support vector machine

Hossein Moosaei[1,2] · Fatemeh Bazikar[3] · Panos M. Pardalos[4]

Accepted: 1 June 2023

Abstract

In recent years, multi-task learning (MTL) has become a popular field in machine learning and has a key role in various domains. Sharing knowledge across tasks in MTL can improve the performance of learning algorithms and enhance their generalization capability. A new approach called the multi-task least squares twin support vector machine (MTLS-TSVM) was recently proposed as a least squares variant of the direct multi-task twin support vector machine (DMTSVM). Unlike DMTSVM, which solves two quadratic programming problems, MTLS-TSVM solves two linear systems of equations, resulting in a reduced computational time. In this paper, we propose an enhanced version of MTLS-TSVM called the improved multi-task least squares twin support vector machine (IMTLS-TSVM). IMTLS-

Twin Support Vector Machines (TSVM)

and

Imbalanced Data

Neural Networks 157 (2023) 125–135

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

ELSEVIER

Inverse free reduced universum twin support vector machine for imbalanced data classification

Hossein Moosaei [a,b,*], M.A. Ganaie [c,d], Milan Hladík [b], M. Tanveer [c]

[a] Department of Informatics, Faculty of Science, Jan Evangelista Purkyně University, Ústí nad Labem, Czech Republic
[b] Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
[c] Department of Mathematics, Indian Institute of Technology Indore, Simrol, Indore, 453552, India
[d] Department of Robotics, University of Michigan, Ann Arbor, MI, 48109, USA

ARTICLE INFO

Article history:
Received 8 March 2022
Revised and accepted 4 October 2022
Available online 15 October 2022

Keywords:
Universum
Class-imbalanced
Twin support vector machine
Universum twin support vector machine

ABSTRACT

Imbalanced datasets are prominent in real-world problems. In such problems, the data samples in one class are significantly higher than in the other classes, even though the other classes might be more important. The standard classification algorithms may classify all the data into the majority class, and this is a significant drawback of most standard learning algorithms, so imbalanced datasets need to be handled carefully. One of the traditional algorithms, twin support vector machines (TSVM), performed well on balanced data classification but poorly on imbalanced datasets classification. In order to improve the TSVM algorithm's classification ability for imbalanced datasets, recently, driven by the universum twin support vector machine (UTSVM), a reduced universum twin support vector machine for class imbalance learning (RUTSVM) was proposed. The dual problem and finding classifiers involve

Twin Support Vector Machines (TSVM)

and

Optimization Methods

Neural Processing Letters (2021) 53:1545–1564
https://doi.org/10.1007/s11063-021-10464-3

Generalized Twin Support Vector Machines

H. Moosaei[1] · S. Ketabchi[2] · M. Razzaghi[2] · M. Tanveer[3]

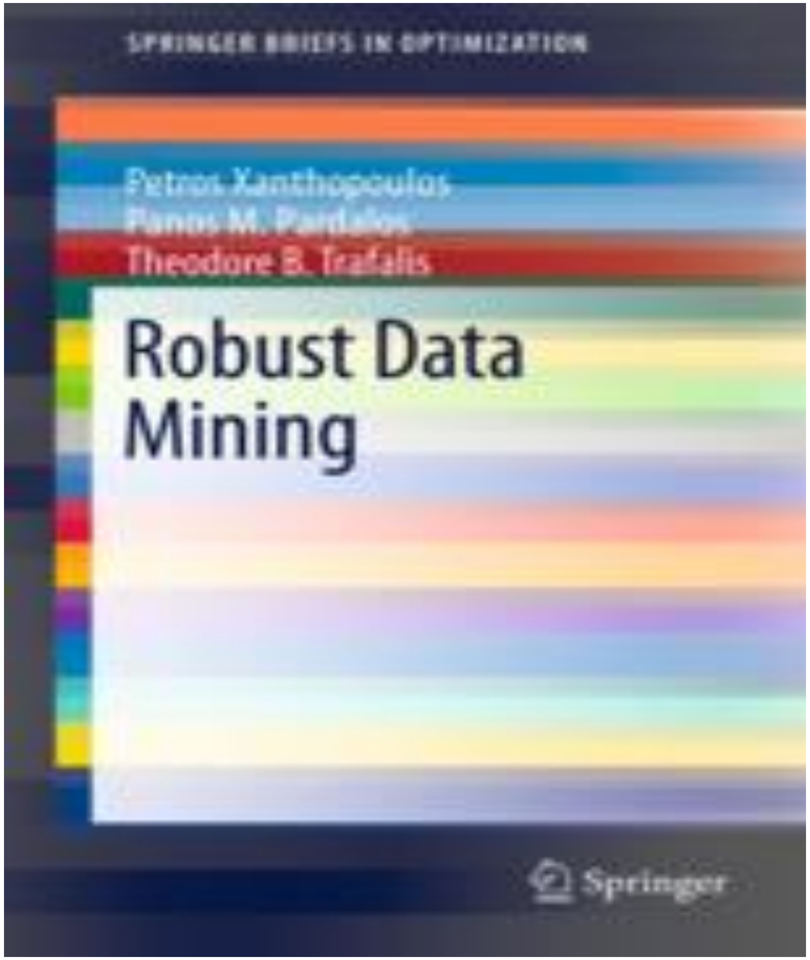
Accepted: 9 February 2021 / Published online: 6 March 2021

Abstract
In this paper, we propose two efficient approaches of twin support vector machines (TWSVM). The first approach is to reformulate the TWSVM formulation by introducing L_1 and L_∞ norms in the objective functions, and convert into linear programming problems termed as LTWSVM for binary classification. The second approach is to solve the primal TWSVM, and convert into completely unconstrained minimization problem. Since the objective function is convex, piecewise quadratic but not twice differentiable, we present an efficient algorithm using the generalized Newton's method termed as GTWSVM. Computational comparisons of the proposed LTWSVM and GTWSVM on synthetic and several real-world benchmark datasets exhibits significantly better performance with remarkably less computational time in comparison to relevant baseline methods.

Keywords Support vector machines · Twin support vector machines · Linear programming · Unconstrained minimization problem · Generalized Newton-Armijo method

Robust TVSVM

Wang, X., Pardalos, P.M. A Survey of Support Vector Machines with Uncertainties. *Ann. Data. Sci.* **1,** 293–309 (2014). https://doi.org/10.1007/s40745-014-0022-8



Pattern Recognition 46 (2013) 305–316

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Robust twin support vector machine for pattern classification

Zhiquan Qi, Yingjie Tian*, Yong Shi*

Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing 100190, China

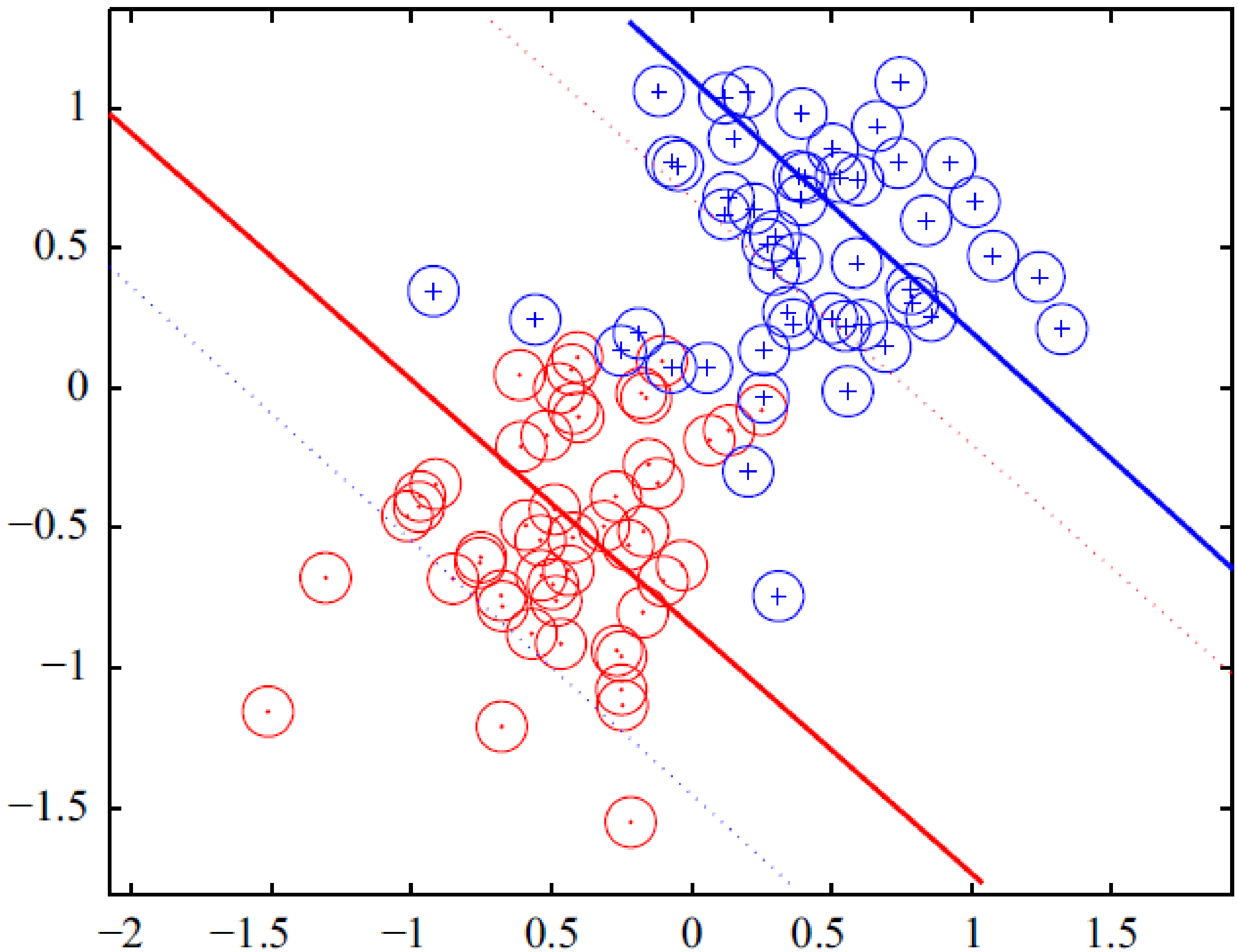
A R T I C L E I N F O

Article history:
Received 27 December 2011
Received in revised form
22 June 2012
Accepted 27 June 2012
Available online 4 July 2012

Keywords:
Classification
Twin support vector machine
Second order cone programming
Robust

A B S T R A C T

In this paper, we proposed a new robust twin support vector machine (called $\mathcal{R}$-TWSVM) via second order cone programming formulations for classification, which can deal with data with measurement noise efficiently. Preliminary experiments confirm the robustness of the proposed method and its superiority to the traditional robust SVM in both computation time and classification accuracy. Remarkably, since there are only inner products about inputs in our dual problems, this makes us apply kernel trick directly for nonlinear cases. Simultaneously we does not need to solve the extra inverse of matrices, which is totally different with existing TWSVMs. In addition, we also show that the TWSVMs are the special case of our robust model and simultaneously give a new dual form of TWSVM by degenerating R-TWSVM, which successfully overcomes the existing shortcomings of TWSVM.


1
0.5
0
−0.5
−1
−1.5
−2
−1.5
−1
0.5
0
0.5
1
1.5

Twin Support Vector Machines (TSVM)

and

Multi-class data sets

Cogn Comput (2013) 5:580–588
DOI 10.1007/s12559-012-9179-7

A Twin Multi-Class Classification Support Vector Machine

Yitian Xu · Rui Guo · Laisheng Wang

Received: 16 September 2011 / Accepted: 1 August 2012 / Published online: 21 August 2012

Abstract Twin support vector machine (TSVM) is a novel machine learning algorithm, which aims at finding two nonparallel planes for each class. In order to do so, one needs to resolve a pair of smaller-sized quadratic programming problems rather than a single large one. Classical TSVM is proposed for the binary classification problem. However, multi-class classification problem is often met in our real world. For this problem, a new multi-

learning technique. Compared with other machine learning approaches like artificial neural networks [2], SVM has many advantages. First, SVM solves a QPP, assuring that once an optimal solution is obtained, it is the unique (global) solution. Second, SVM derives a sparse and robust solution by maximizing the margin between the two classes. Third, SVM implements the structural risk minimization principle rather than the empirical risk minimization

Twin-KSVC could be considered as a novel multi-class categorization depending on TWSVM (Xu et al., 2013). The approach employs ternary outputs of {-1, 0,+1} to assess all of the training data in a ‘‘1-versus-1-versus-rest’’ framework. Two non-parallel hyperplanes for classes +1 and -1 are created by addressing two quadratic programming problems, and the remaining sample data sets are labeled as 0.

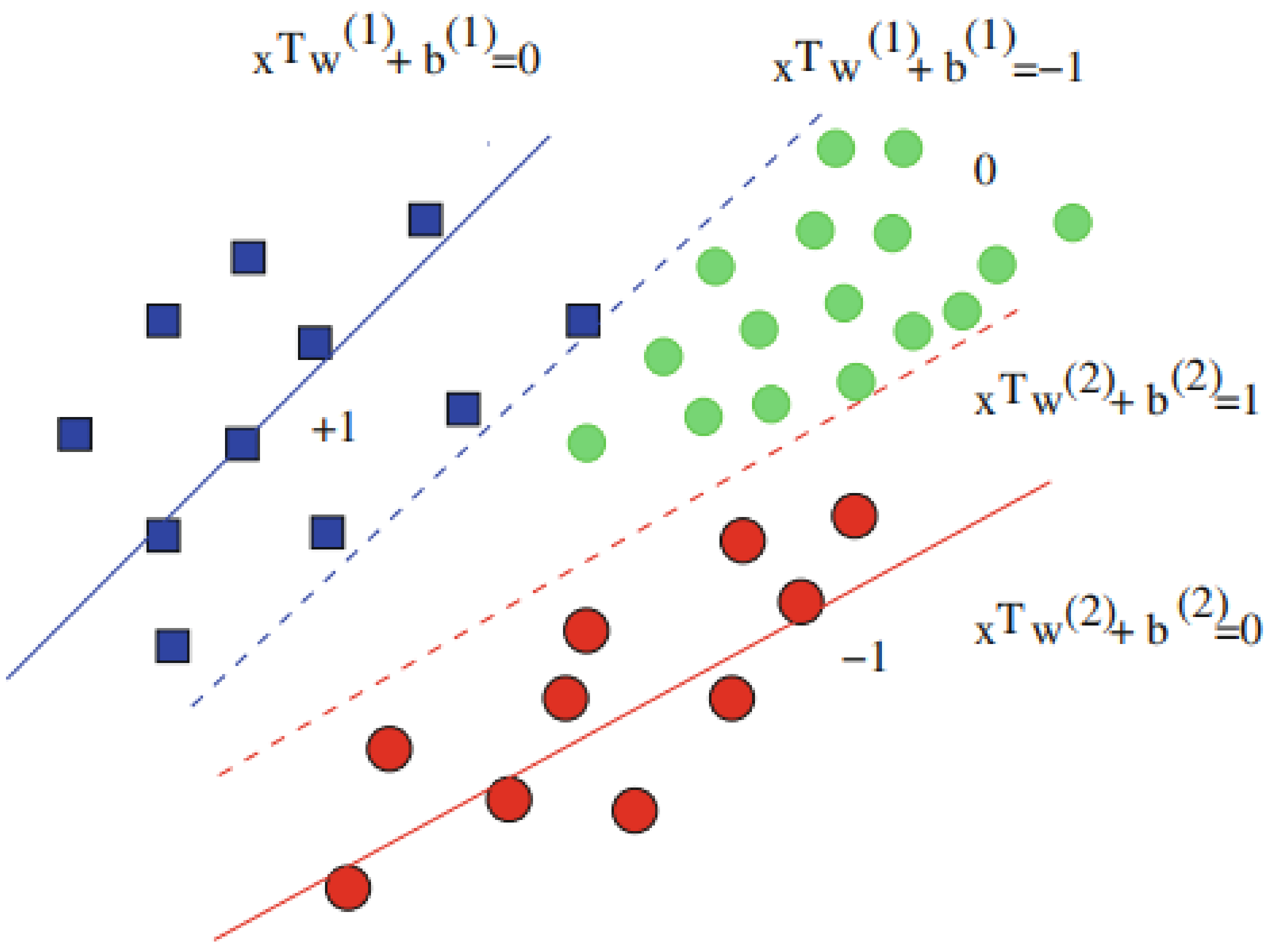


Fig. 3 Illustration of Twin-KSVC

demonstration of the Twin-KSVC technique is shown in Fig. 3 In the Twin-KSVC, two non-parallel hyperplanes are searched:

$$x^T w_1 + b_1 = 0, \quad x^T w_2 + b_2 = 0.$$

Assuming three data matrices, $A_{m_1 \times n}$, $B_{m_2 \times n}$ and $C_{m_3 \times n}$ with class labels +1, −1 and 0 correspondingly, is identical to the preceding subsection. Solving the subsequent pair of QPPs yields the Twin-KSVC classifiers:

$$\begin{aligned} \min_{w_1, b_1, q_1, q_2} \quad & \frac{1}{2} \| A w_1 + e_1 b_1 \|^2 + c_1 e_2^T q_1 + c_2 e_3^T q_2, \\ \text{subject to} \quad & -(B w_1 + e_2 b_1) + q_1 \geq e_2, \\ & -(C w_1 + e_3 b_1) + q_2 \geq e_3 (1 - \epsilon), \\ & q_1 \geq 0, \; q_2 \geq 0, \end{aligned} \qquad (1)$$

and

$$\begin{aligned} \min_{w_2, b_2, q_3, q_4} \quad & \frac{1}{2} \| B w_2 + e_2 b_2 \|^2 + c_3 e_1^T q_3 + c_4 e_3^T q_4, \\ \text{subject to} \quad & A w_2 + e_1 b_2 + q_3 \geq e_1, \\ & C w_2 + e_3 b_2 + q_4 \geq e_3 (1 - \epsilon), \\ & q_3 \geq 0, \; q_4 \geq 0. \end{aligned} \qquad (2)$$

where $c_1, c_2, c_3, c_4 \geq 0$ considers as regularization parameters, e_1, e_2, e_3 and e_4 are vectors of one's of proper dimension, q_1, q_2, q_3, and q_4 are slack variables, and ϵ is a parameter with a positive value.

For Twin-KSVC and NTW-KSVC linear versions:

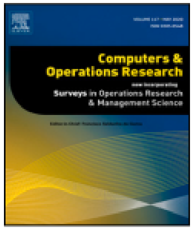
$$f(x_i) = \begin{cases} +1, & x_i^T w_1 + b_1 > -1 + \epsilon, \\ -1, & x_i^T w_2 + b_2 < 1 - \epsilon, \\ 0, & \text{otherwise.} \end{cases}$$

Computers & Operations Research 160 (2023) 106370

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor



Newton-based approach to solving K-SVCR and Twin-KSVC multi-class classification in the primal space

Hossein Moosaei [a,b,*], Milan Hladík [b], Mohamad Razzaghi [c], Saeed Ketabchi [c]

[a] Department of Informatics, Faculty of Science, Jan Evangelista Purkyně University, Ústí nad Labem, Czech Republic
[b] Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
[c] Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran

ARTICLE INFO

Keywords:
Multi-class classification
Support vector machine
Twin SVM
K-SVCR
Twin-KSVC
Newton's method

ABSTRACT

Multi-class classification is an important problem in machine learning, which often occurs in the real world and is an ongoing research issue. Support vector classification-regression machine for k-class classification (K-SVCR) and twin k-class support vector classification (Twin-KSVC) are two novel machine learning methods for multi-class classification problems. This paper presents novel methods to solve the primal problems of K-SVCR and Twin-KSVC, known as NK-SVCR and NTW-KSVC, respectively. The proposed methods evaluate all training data into a "1-versus-1-versus-rest" structure, so it generates ternary outputs $\{-1, 0, +1\}$. The primal problems are reformulated as unconstrained optimization problems so that the objective functions are only once differentiable, not twice, therefore an extension of the Newton-Armijo algorithm is adopted for finding their solution. To test the efficiency and validity of the proposed methods, we compare the classification accuracy and learning time of these methods with K-SVCR and Twin-KSVC on the United States Postal Service (USPS) handwriting digital data sets and several University of California Irvine (UCI) benchmark data sets.

Twin-KSVC problems (1) and (2) can be rewritten as follows:

$$\min_{q_1, q_2, y_1} \frac{1}{2}\|T_1 y_1\|^2 + c_1\|q_1\|^2 + c_2\|q_2\|^2,$$
$$\text{subject to} \quad S_1 y_1 + e_2 \leq q_1, \quad S_2 y_1 + e_3(1-\epsilon) \leq q_2, \quad q_1, q_2 \geq 0. \tag{3}$$

$$\min_{q_3, q_4, y_2} \frac{1}{2}\|T_2 y_2\|^2 + c_3\|q_3\|^2 + c_4\|q_4\|^2,$$
$$\text{subject to} \quad S_3 y_2 + e_1 \leq q_3, \quad S_4 y_2 + e_3(1-\epsilon) \leq q_4, \quad q_3, q_4 \geq 0. \tag{4}$$

where $T_1 = [A \ e_1]$, $S_1 = [B \ e_2]$, $S_2 = [C \ e_3]$, and $y_1 = [w_1; \ b_1]$. Analogously, $T_2 = [B \ e_2]$, $S_3 = [-A \ -e_1]$, $S_4 = [-C \ -e_3]$, and $y_2 = [w_2; \ b_2]$. For the optimal solution of problem (1) we have $q_1 = (S_1 y_1 + e_2)_+$ and $q_2 = (S_2 y_1 + e_3(1-\epsilon))_+$ (Lee and Mangasarian, 2001b; Mangasarian and Musicant, 1999).

Therefore we can substitute them in the objective function. Also, q_3 and q_4 can be substituted in a similar way. Then, problems (3) and (4) will be equivalent to the following unconstrained minimization problems:

$$\min_{y_1} \psi_1(y_1) = \min_{y_1} \frac{1}{2}\|T_1 y_1\|^2 + c_1\|(S_1 y_1 + e_2)_+\|^2 + c_2\|(S_2 y_1 + e_3(1-\epsilon))_+\|^2, \qquad (5)$$

and

$$\min_{y_2} \psi_2(y_2) = \min_{y_2} \frac{1}{2}\|T_2 y_2\|^2 + c_3\|(S_3 y_2 + e_1)_+\|^2 + c_4\|(S_4 y_2 + e_3(1-\epsilon))_+\|^2. \qquad (6)$$

As the objective functions of the above problems are only once differentiable we will use Generalized Newton's Method to solve them.

Neural Networks 166 (2023) 471–486

ELSEVIER

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Sparse solution of least-squares twin multi-class support vector machine using ℓ_0 and ℓ_p-norm for classification and feature selection

Hossein Moosaei [a,c,*], Milan Hladík [b,c]

[a] Department of Informatics, Faculty of Science, Jan Evangelista Purkyně University, Ústí nad Labem, Czech Republic
[b] Department of Applied Mathematics, School of Computer Science, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
[c] Department of Econometrics, Prague University of Economics and Business, Czech Republic

A R T I C L E   I N F O

Article history:
Received 25 November 2022
Received in revised form 22 June 2023
Accepted 26 July 2023
Available online 1 August 2023

Keywords:
Multi-class classification
Twin k-class support vector classification
Least-squares
Cardinality-constrained optimization problem

A B S T R A C T

In the realm of multi-class classification, the twin K-class support vector classification (Twin-KSVC) generates ternary outputs $\{-1, 0, +1\}$ by evaluating all training data in a "1-versus-1-versus-rest" structure. Recently, inspired by the least-squares version of Twin-KSVC and Twin-KSVC, a new multi-class classifier called improvements on least-squares twin multi-class classification support vector machine (ILSTKSVC) has been proposed. In this method, the concept of structural risk minimization is achieved by incorporating a regularization term in addition to the minimization of empirical risk. Twin-KSVC and its improvements have an influence on classification accuracy. Another aspect influencing classification accuracy is feature selection, which is a critical stage in machine learning, especially when working with high-dimensional datasets. However, most prior studies have not addressed this crucial aspect. In this study, motivated by ILSTKSVC and the cardinality-constrained optimization

Exploring Novel Methods Inspired by Twin Support Vector Machines (TSVM)

Neural Comput & Applic (2014) 24:1207–1220
DOI 10.1007/s00521-012-1306-6

ORIGINAL ARTICLE

Twin support vector hypersphere (TSVH) classifier for pattern recognition

Xinjun Peng · Dong Xu

Received: 14 January 2012 / Accepted: 7 December 2012 / Published online: 5 February 2013

Abstract Motivated by the support vector data description, a classical one-class support vector machine, and the twin support vector machine classifier, this paper formulates a twin support vector hypersphere (TSVH) classifier, a novel binary support vector machine (SVM) classifier that determines a pair of hyperspheres by solving two related SVM-type quadratic programming problems, each of which is smaller than that of a conventional SVM, which means that this TSVH is more efficient than the classical

powerful method in machine learning algorithms. Within a few years after its introduction, the SVM has already outperformed most other systems in a wide variety of applications. These include a wide spectrum of research areas, ranging from pattern recognition [5, 6], text categorization [7], biomedicine [8], brain–computer interface [9], and financial applications [10].

The theory of SVM proposed by Vapnik et al. is based on the structural risk minimization (SRM) principle [1–4]. In

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 19, 2020, accepted April 19, 2020, date of publication April 27, 2020, date of current version May 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990611

Twin Hyper-Ellipsoidal Support Vector Machine for Binary Classification

ZEINAB EBRAHIMPOUR[1,2], WANGGEN WAN[1,2], (Senior Member, IEEE), ARASH SIOOFY KHOOJINE[3], AND LI HOU[4]
[1]School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
[2]Institute of Smart City, Shanghai University, Shanghai 200444, China
[3]School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai 200400, China
[4]School of Information Engineering, Huangshan University, Huangshan 245041, China

Corresponding author: Zeinab Ebrahimpour (z_ebrahimpour@shu.edu.cn)

This work was supported in part by the Science and Technology Commission of Shanghai Municipality under Grant 18510760300, in part by the Anhui Natural Science Foundation under Grant 1908085MF178, and in part by the Anhui Excellent Young Talents Support Program Project under Grant gxyqZD2019069.

ABSTRACT In this paper, a twin hyper-ellipsoidal support vector machine (TESVM) for binary classification of data is presented. Similar to twin support SVM(TWSVM) and twin hypersphere SVM (THSVM), as in the literature, our proposed method finds two hyper-ellipsoidals by solving two related SVM-type quadratic programming problem (QPPs), each of which is smaller than that of the classical SVM, causing it to achieve higher speed. The main idea of this paper is to employ Mahalanobis distance-based kernels for two classes of data in the THSVM algorithm to improve its generalization performance. Since the kernel used in SVM, TWSVM, and THSVM is based on Euclidean distance, it is assumed that the data points have

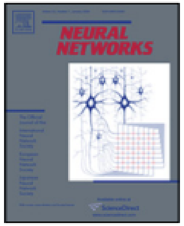
Twin SVM for regression

Neural Networks 23 (2010) 365–372

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



TSVR: An efficient Twin Support Vector Machine for regression

Peng Xinjun *

Department of Mathematics, Shanghai Normal University, 200234, PR China
Scientific Computing Key Laboratory of Shanghai Universities, 200234, PR China

ARTICLE INFO

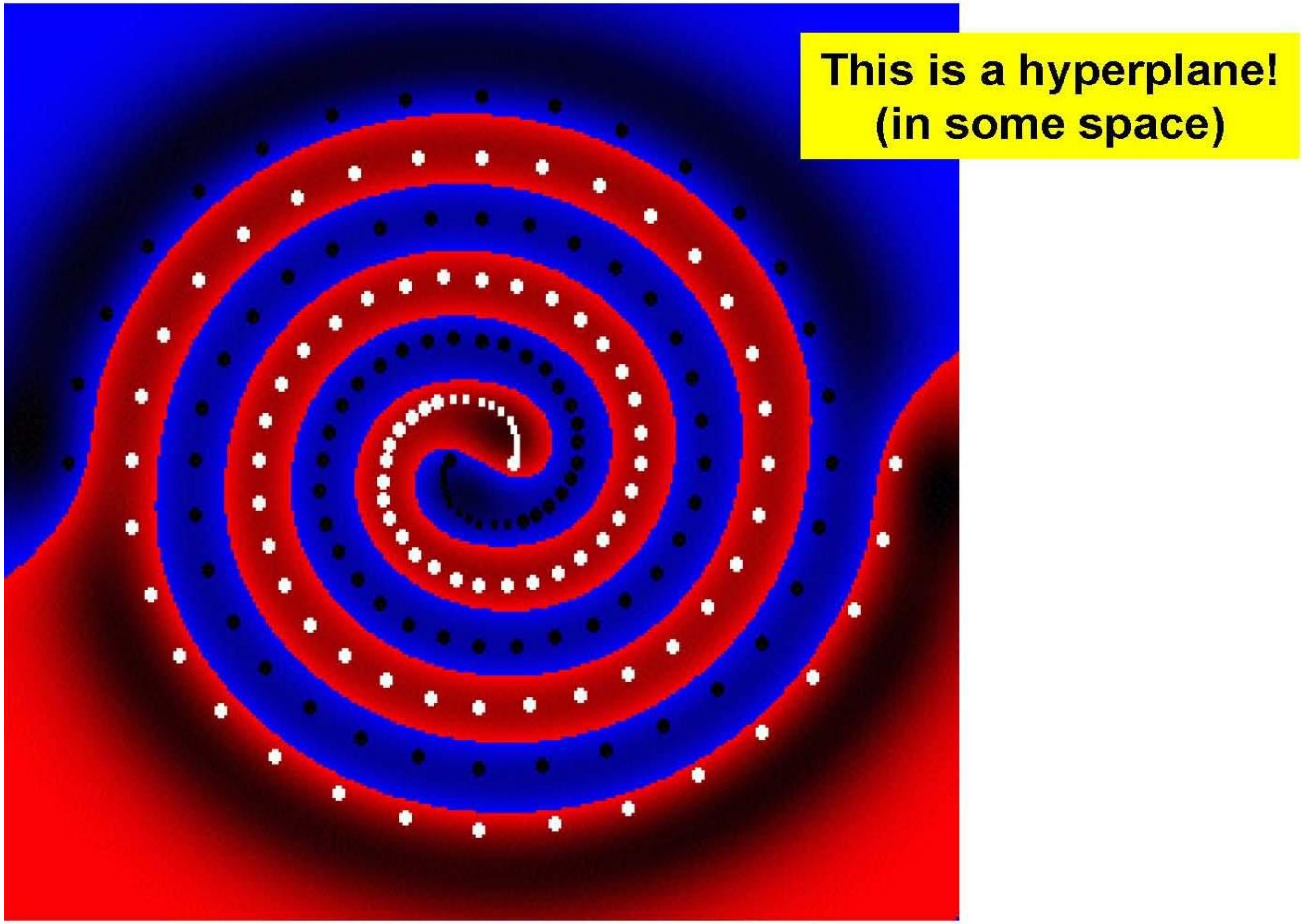
Article history:
Received 17 April 2009
Received in revised form 1 July 2009
Accepted 6 July 2009

Keywords:
Machine learning
Support vector machine
Regression
Nonparallel planes
ϵ-insensitive bound

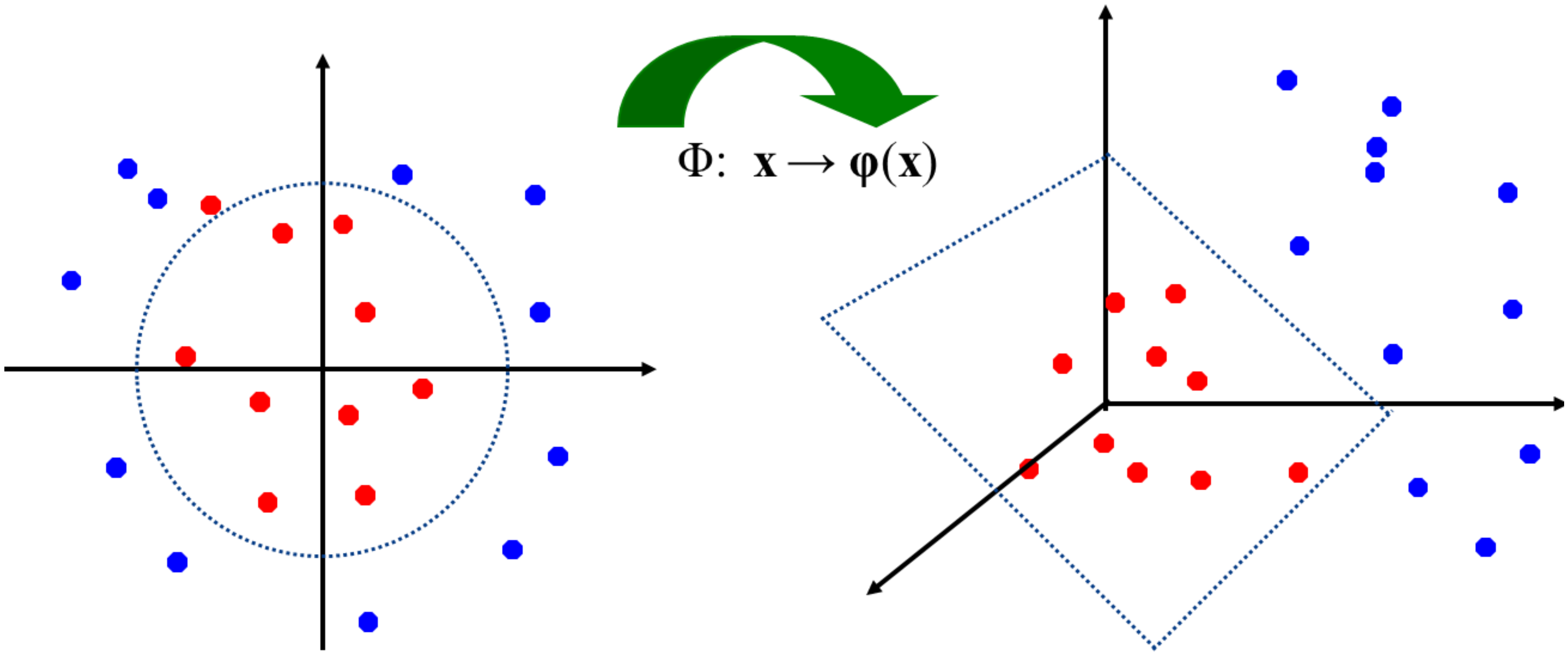
ABSTRACT

The learning speed of classical Support Vector Regression (SVR) is low, since it is constructed based on the minimization of a convex quadratic function subject to the pair groups of linear inequality constraints for all training samples. In this paper we propose Twin Support Vector Regression (TSVR), a novel regressor that determines a pair of ϵ-insensitive up- and down-bound functions by solving two related SVM-type problems, each of which is smaller than that in a classical SVR. The TSVR formulation is in the spirit of Twin Support Vector Machine (TSVM) via two nonparallel planes. The experimental results on several artificial and benchmark datasets indicate that the proposed TSVR is not only fast, but also shows good generalization performance.

Nonlinear separable problems



Non-linear SVMs: Feature spaces



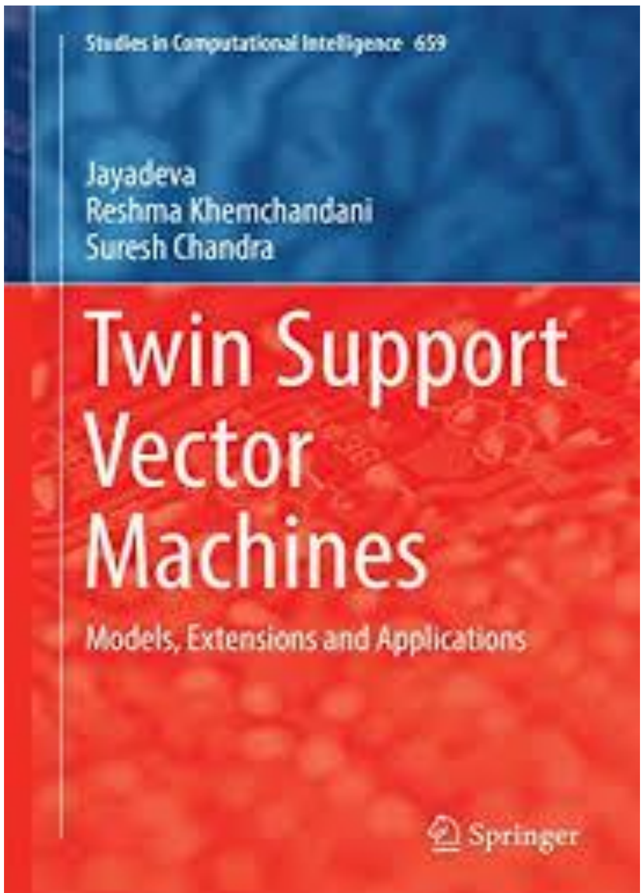

John Shawe-Taylor
and Nello Cristianini
Kernel Methods
for Pattern Analysis

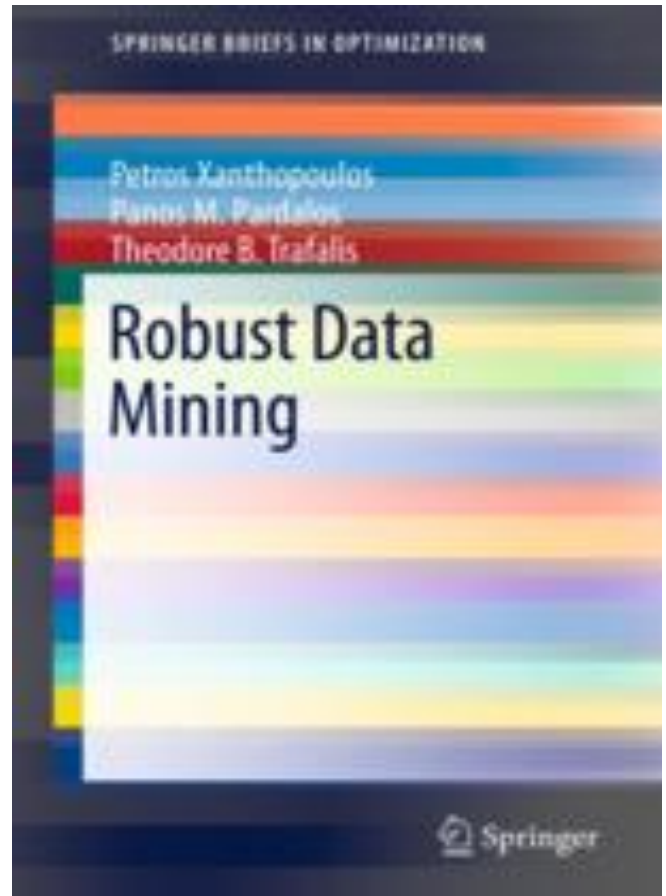
Challenging issues with TSVM

1. "Exploring Innovative Approaches for Data Separation"
2. "Introducing an Efficient Optimization Model for Enhanced Performance"
3. "Addressing Existing Challenges with Novel Solutions"
4. "Extending Binary Classification Methods to Multi-class Classification"
5. "Utilizing Sparse Solutions for Feature Selection"
6. "Dealing with Unbalanced Data and Structural Datasets"
7. "Tackling Multi-label Classification and Semi-supervised Learning"
8. "Handling Massive Datasets with TSVM"

Many Models of SVM

Wang, X., Pardalos, P.M. A Survey of Support Vector Machines with Uncertainties. *Ann. Data. Sci.* **1,** 293–309 (2014). https://doi.org/10.1007/s40745-014-0022-8





Resources: Datasets

- UCI Repository:
  http://www.ics.uci.edu/~mlearn/MLRepository.html
- UCI KDD Archive:
  http://kdd.ics.uci.edu/summary.data.application.html
- Statlib: http://lib.stat.cmu.edu/
- Delve: http://www.cs.utoronto.ca/~delve/

Journals

- Journal of Machine Learning Research Machine Learning
- IEEE Transactions on Neural Networks
- IEEE Transactions on Pattern Analysis and Machine Intelligence
- Annals of Statistics
- Journal of the American Statistical Association
- …

Resources: Conferences

- International Conference on the Dynamics of Information Systems (DIS)
- International Conference on Machine Learning (ICML)
- European Conference on Machine Learning (ECML)
- Neural Information Processing Systems (NIPS)
- International Joint Conference on Artificial Intelligence (IJCAI)
- ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)
- IEEE Int. Conf. on Data Mining (ICDM)

The End ...

Thank you!

Appendix

Optimization

Optimization Problem

- Finding the minimizer of a function subject to constraints:

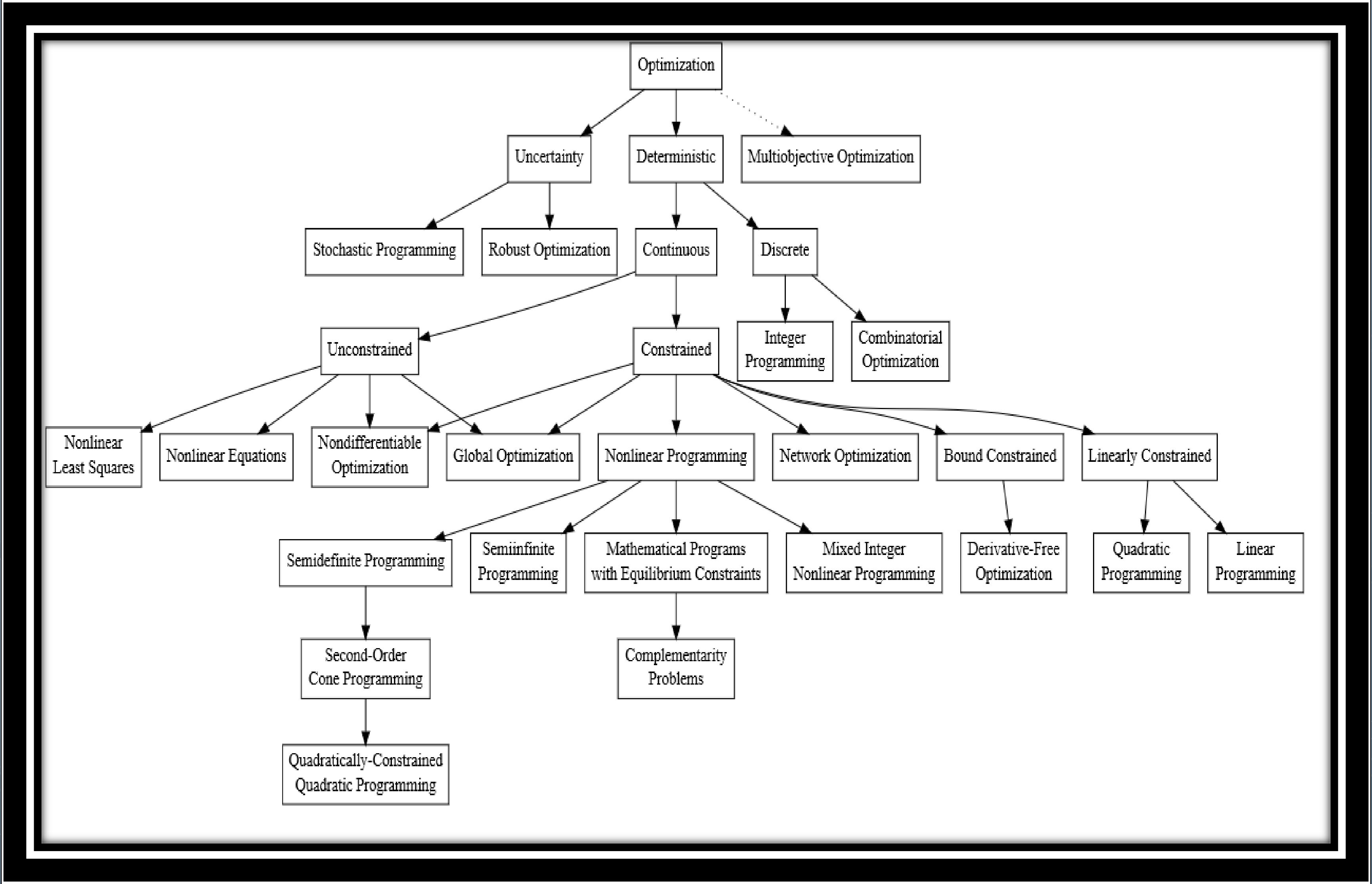
$$\begin{aligned} \underset{x}{\text{minimize}}\ & f_0(x) \\ \text{s.t.}\ & f_i(x) \leq 0,\ i = \{1, \dots, k\} \\ & h_j(x) = 0,\ j = \{1, \dots, l\} \end{aligned}$$

Some different types

of

optimization problems?

Optimization Taxonomy



Applications of Optimization?

- Transportation
- Resource Allocation
- Portfolio Management
- Economics
- Manufacturing System
- Medical Science
- **Data Mining**

Karush-Kuhn-Tucker Optimality Conditions

Optimality Criteria

- <u>Big question</u>: *How do we know that we have found the "optimum" for min f(x)?*

<u>Answer</u>: Test the solution for the "necessary and sufficient conditions"

Optimality Conditions – Unconstrained Case

- Let x* be the point that we think is the minimum for f(x)

Necessary condition (for optimality):

$$\nabla f(x^*) = 0$$

- A point that satisfies the necessary condition is a stationary point It can be a minimum, maximum, or saddle point

- *How do we know that we have a minimum?*
- Answer: Sufficiency Condition:

  The sufficient conditions for x* to be a strict local minimum are:

$$\nabla f(x^*) = 0$$

$\nabla^2 f(x^*)$ is positive definite

Constrained Case – KKT Conditions

- To proof a claim of optimality in constrained minimization (or maximization), we have to check the found point with respect to the (Karesh) Kuhn Tucker conditions.
- Kuhn and Tucker extended the Lagrangian theory to include the general classical single-objective nonlinear programming problem:

minimize $f(x)$

Subject to $g_j(x) \geq 0$ for $j = 1, 2, ..., J$

$h_k(\mathbf{x}) = 0$ for $k = 1, 2, ..., K$

$x = (x_1, x_2, ..., x_N)$

Necessary KKT Conditions

For the problem:

Min f(x)

s.t. $g(x) \leq 0$

(n variables, m constraints)

The necessary conditions are:

$\nabla f(x) + \Sigma \mu_i g_i(x) = 0$ (optimality)

$g_i(x) \leq 0$ for i = 1, 2, ..., m (feasibility)

$\mu_i g_i(x) = 0$ for i = 1, 2, ..., m (complementary slackness condition)

$\mu_i \geq 0$ for i = 1, 2, ..., m (non-negativity)

Note that the first condition gives n equations.

Necessary KKT Conditions (General Case)

- For general case (n variables, M Inequalities, L equalities):

  Min f(x)

  s.t.

  $g_{i}(x) \leq 0$ for i = 1, 2, ..., M

  $h_j(x) = 0$ for j = 1, 2, ..., L

- In all this, the assumption is that $\nabla g_j(x^*)$ for j belonging to active constraints and $\nabla h_k(x^*)$ for k = 1, ...,K are linearly independent

- The necessary conditions are:

  $\nabla f(x) + \sum \mu_i \nabla g_i(x) + \sum \lambda_j \nabla h_j(x) = 0$ (optimality)

  $g_i(x) \leq 0$ for i = 1, 2, ..., M (feasibility)

  $h_j(x) = 0$ for j = 1, 2, ..., L (feasibility)

  $\mu_i g_i(x) = 0$ for i = 1, 2, ..., M (complementary slackness condition)

  $\mu_i \geq 0$ for i = 1, 2, ..., M (non-negativity)

  (Note: λ_j is unrestricted in sign)

Restating the Optimization Problem

- Kuhn Tucker Optimization Problem: Find vectors $X_{(N\times1)}$, $\mu_{(1\times M)}$ and $\lambda_{(1\times K)}$ that satisfy:

  $\nabla f(x) + \Sigma\, \mu_i \nabla g_i(x) + \Sigma\, \lambda_j \nabla h_j(x) = 0$ (optimality)

  $g_i(x) \leq 0$ for i = 1, 2, ..., M (feasibility)

  $h_j(x) = 0$ for j = 1, 2, ..., L (feasibility)

  $\mu_i\, g_{i(x)} = 0$ for i = 1, 2, ..., M (complementary slackness condition)

  $\mu_i \geq 0$ for i = 1, 2, ..., M (non-negativity)

- If x* is an optimal solution to NLP, then there exists a (μ*, λ*) such that (x*, μ*, λ*) solves the Kuhn–Tucker problem.

- Above equations not only give the necessary conditions for optimality, but also provide a way of finding the optimal point.

Limitations

- Necessity theorem helps identify points that are not optimal. A point is not optimal if it does not satisfy the Kuhn–Tucker conditions.

- On the other hand, not all points that satisfy the Kuhn-Tucker conditions are optimal points.

- The Kuhn–Tucker sufficiency theorem gives conditions under which a point becomes an optimal solution to a single-objective NLP.

Sufficiency Condition

- Sufficient conditions that a point x* is a strict local minimum of the NLP problem, where f, g_j, and h_k are twice differentiable functions are that

  1) The necessary KKT conditions are met.

  2) The Hessian matrix $\nabla^2 L(x*) = \nabla^2 f(x*) + \Sigma \mu_i \nabla^2 g_i(x*) + \Sigma l_j \nabla^2 h_j(x*)$ is positive definite on a subspace of R^n as defined by the condition:

  $y^T \nabla^2 L(x^*)\, y \geq 0$ is met for every vector $y_{(1xN)}$ satisfying:

  $\nabla g_j(\mathbf{x^*})y < 0$ for j belonging to $I_1 = \{ j \mid g_j(\mathbf{x^*}) = 0, u_j^* > 0\}$ (active constraints)

  $\nabla h_k(\mathbf{x^*})y = 0$ for k = 1, ..., K

  $y \quad 0$

KKT Sufficiency Theorem (Special Case)

- Consider the classical single objective NLP problem.

  minimize $f(x)$

  Subject to $g_j(x) \leq 0$ for $j = 1, 2, \ldots, J$

  $h_k(\mathbf{x}) = 0$ for $k = 1, 2, \ldots, K$

- Let the objective function $f(x)$ be convex, the inequality constraints $g_j(x)$ be all convex functions for $j = 1, \ldots, J$, and the equality constraints $h_k(x)$ for $k = 1, \ldots, K$ be linear.
- If this is true, then the necessary KKT conditions are also sufficient.
- Therefore, in this case, if there exists a solution x^* that satisfies the KKT necessary conditions, then x^* is an optimal solution to the NLP problem.
- In fact, it is a global optimum.

Dual Problem

Generalized Lagrangian Function

- Consider the general (primal) optimization problem

$$\begin{aligned} &minimize \;\; f(w) \\ &subject\ to \;\; g_i(w) \le 0, i = 1, \cdots, k \\ &\qquad\qquad\quad h_j(w) = 0, \; j = 1, \cdots, m \end{aligned}$$

where the functions $f, g_i, i = 1, \cdots, k, and \; h_i, i = 1, \cdots, m,$ are defined on a domain Ω. The generalized Lagrangian was defined as

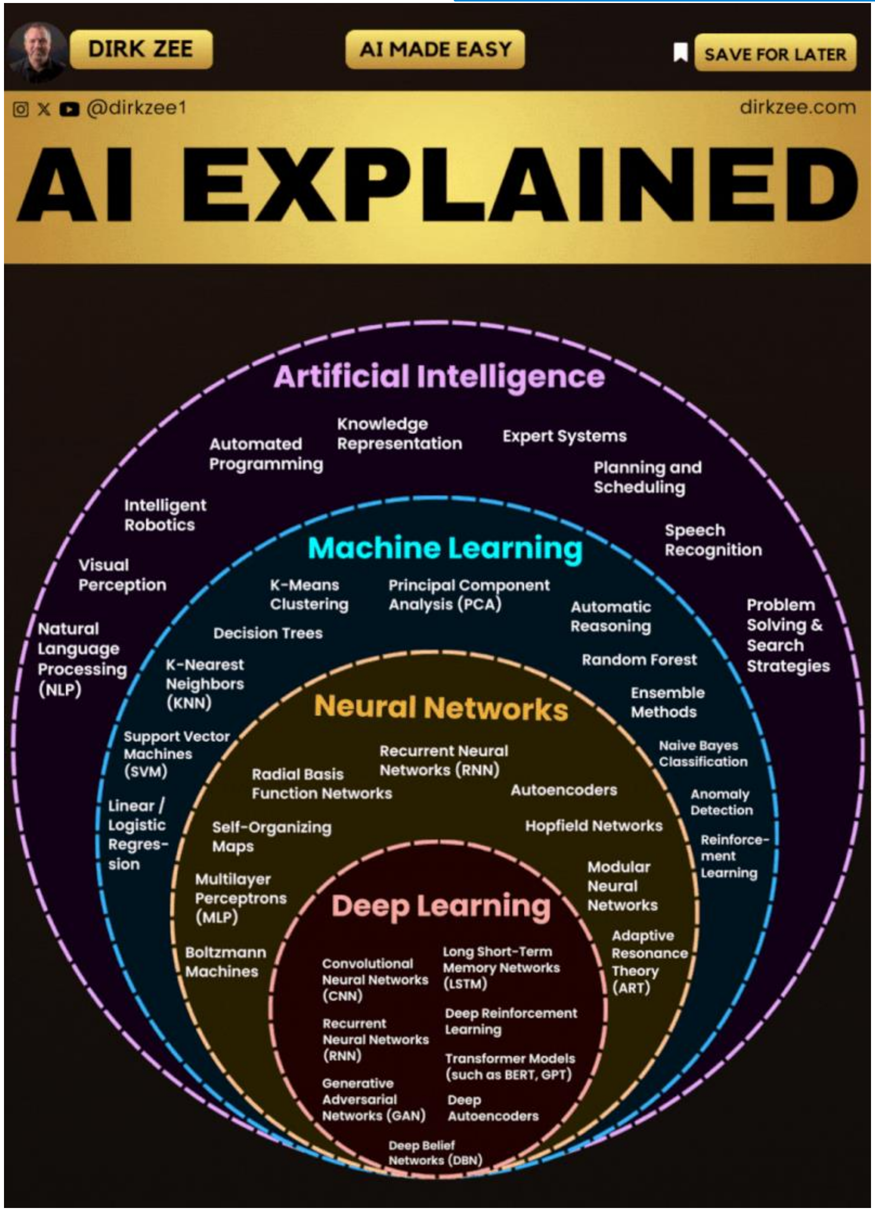
$$\begin{aligned} L(w, \alpha, \beta) &= f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{j=1}^{m} \beta_j h_j(w) \\ &= f(w) + \alpha^T g(w) + \beta^T h(w) \end{aligned}$$

Dual Problem and Strong Duality Theorem

- Given the primal optimization problem, the dual problem of it was defined as

$$\begin{aligned} &maximize \;\; \theta(\alpha,\beta) = \inf_{w\in\Omega} \; L(w,\alpha,\beta) \\ &subject \;\; to \qquad \alpha>0 \end{aligned}$$

- **Strong Duality Theorem**: Given the primal optimization problem, where the domain Ω is convex and the constraints g_i *and* h_i are affine functions. Then the optimum of the primal problem occurs at the same values as the optimum of the dual problem .


DIRK ZEE
AI MADE EASY
SAVE FOR LATER
@dirkzee1
dirkzee.com
AI EXPLAINED
Artificial Intelligence
Automated Programming
Knowledge Representation
Expert Systems
Planning and Scheduling
Intelligent Robotics
Speech Recognition
Visual Perception
Problem Solving & Search Strategies
Natural Language Processing (NLP)
Machine Learning
K-Means Clustering
Principal Component Analysis (PCA)
Automatic Reasoning
Decision Trees
Random Forest
K-Nearest Neighbors (KNN)
Ensemble Methods
Support Vector Machines (SVM)
Naive Bayes Classification
Anomaly Detection
Linear / Logistic Regres-sion
Reinforce-ment Learning
Neural Networks
Recurrent Neural Networks (RNN)
Radial Basis Function Networks
Autoencoders
Self-Organizing Maps
Hopfield Networks
Multilayer Perceptrons (MLP)
Modular Neural Networks
Boltzmann Machines
Adaptive Resonance Theory (ART)
Deep Learning
Convolutional Neural Networks (CNN)
Long Short-Term Memory Networks (LSTM)
Deep Reinforcement Learning
Recurrent Neural Networks (RNN)
Transformer Models (such as BERT, GPT)
Generative Adversarial Networks (GAN)
Deep Autoencoders
Deep Belief Networks (DBN)

Jose C. Principe:

Cycles in Neural Network Research

- How to manage expectations?