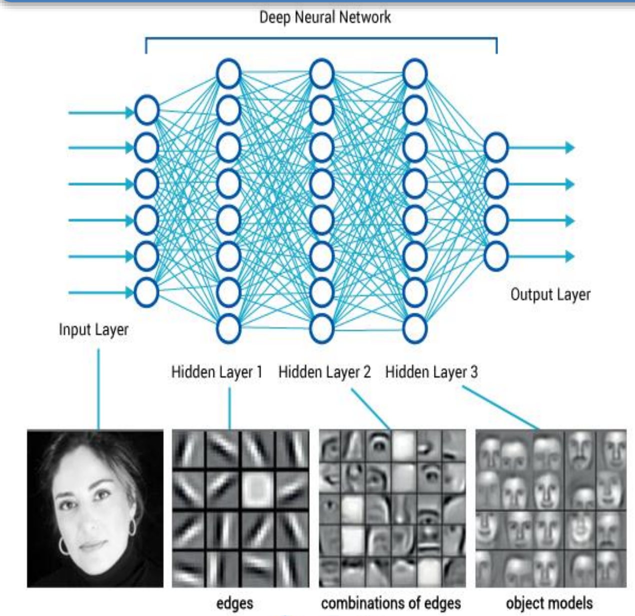
Machine learning analysis of images

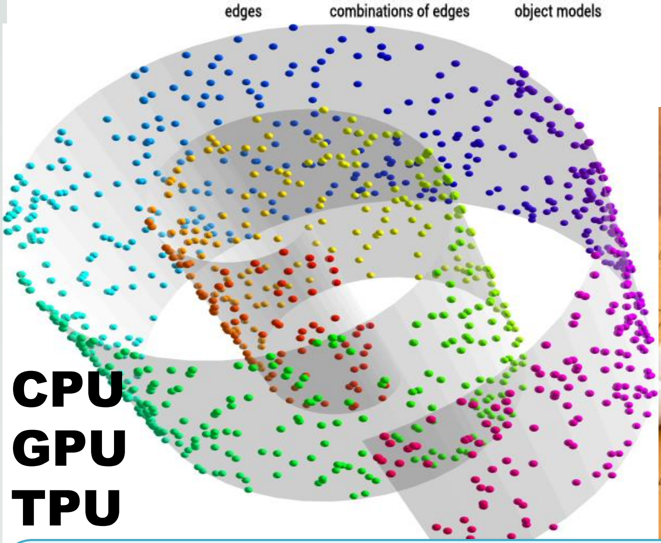
Guowei Wei

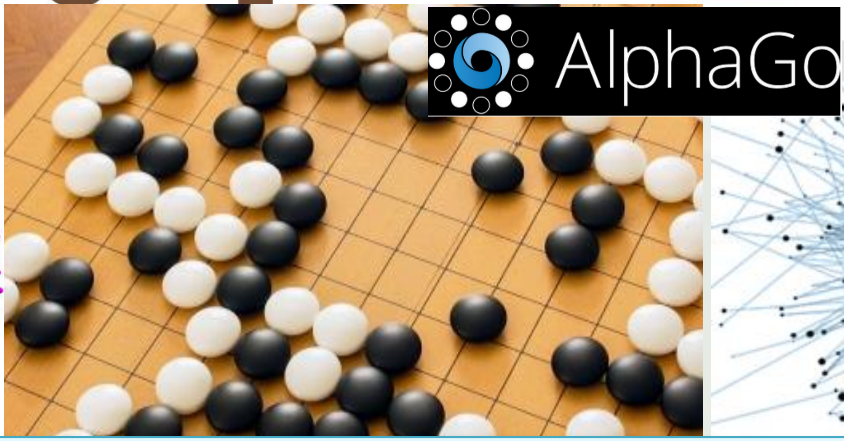
Department of Mathematics
Michigan State University

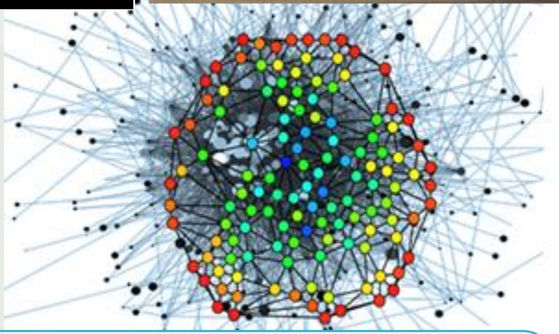
In collaboration with Anne Gelb, Weihong Guo and Duc Nguyen

Welcome to big-data era









CPU
GPU
TPU

Half of all jobs will be done by robots in the near future.

US have closed about 6,000 stores so far this year, compared to 1153 last year.

Why “learn”?

- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- There is no need to learn to calculate your GPA
- Learning is used when:
  - Human expertise does not exist
  - Humans are unable to explain their expertise (face recognition)
  - Solution changes in time (stock returns)
  - Solution needs to be adapted to particular cases (student loan management)

What we talk about when we talk about "learning"

- Learning general models from particular examples (data)
- Data is cheap and abundant; knowledge is expensive and scarce.
- Example in retail: Customer transactions to consumer behavior:

  People who watched "The Godfather" also watch "Casino"

- Build a model that is *a good and useful approximation* to the data.

What is Machine Learning?

- Machine Learning is the study of algorithms that improve their performance at some tasks with experience.
- Optimize a performance criterion using example data.
- Role of Statistics: Inference from a sample
- Role of computer science: Efficient algorithms to
  - Solve the optimization problem (It is mathematics too)
  - Representing and evaluating the model for inference using mathematics.

Growth of Machine Learning

- **Machine learning is preferred approach to**
  - Speech recognition, Natural language processing
  - Computer vision
  - Medical outcomes analysis
  - Bioinformatics
  - Banking, loan management, insurance policy,
  - …
- **This trend is accelerating**
  - Improved machine learning algorithms
  - Improved computer speed
  - Accumulated data sets (big data)
  - The desire to make more money with less effort ☺

Applications

Retail: Market basket analysis, Customer relationship management (CRM)

Finance: Credit scoring, fraud detection

Manufacturing: Optimization, troubleshooting

Medicine: Medical diagnosis, optimal treatment

Telecommunications: Quality of service optimization

Bioinformatics: Motifs, alignment, protein-drug binding

Web mining: Search engines

Image analysis: Face recognition

Character recognition: Different handwriting styles.

Speech recognition: Transfer spoken language into text

...

Face Recognition

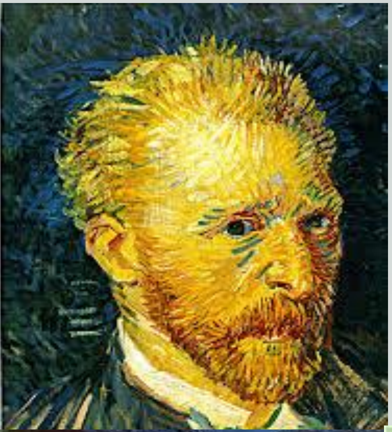
Training examples of a person

Test images

AT&T Laboratories, Cambridge UK
http://www.uk.research.att.com/facedatabase.html

Painting authenticity
(Van Gogh)



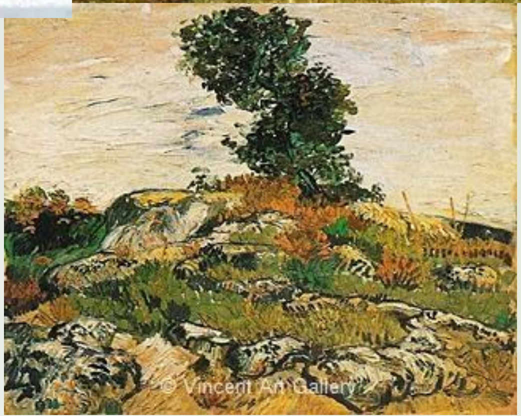




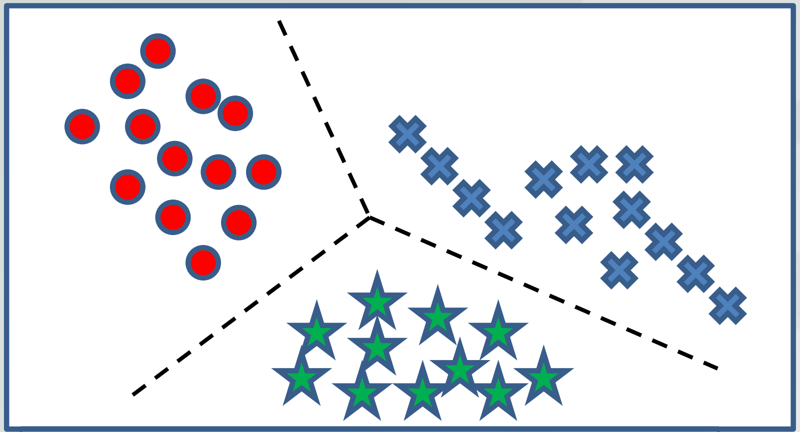




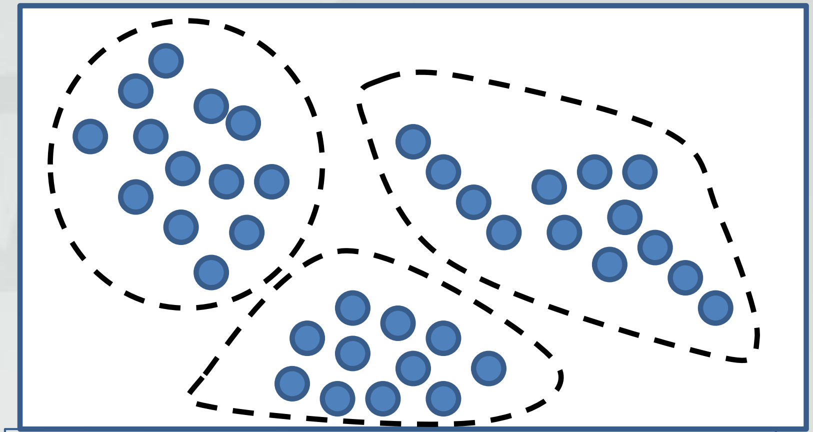


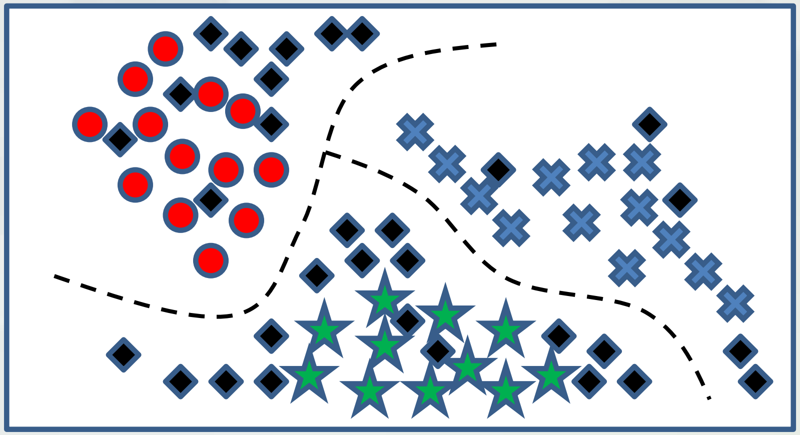
Algorithms



Supervised learning



Unsupervised learning

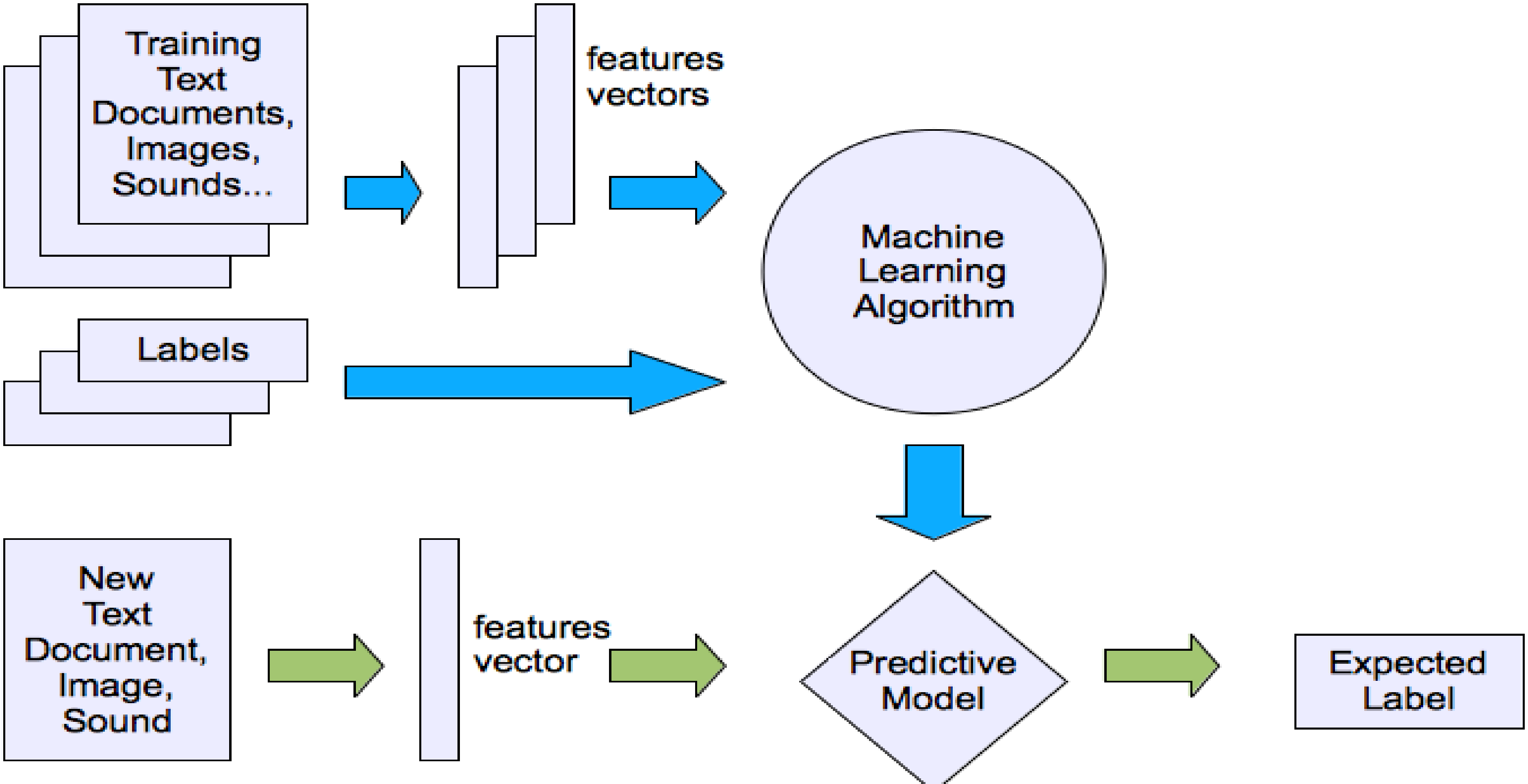


Semi-supervised learning

Machine learning structure

- **Supervised learning**

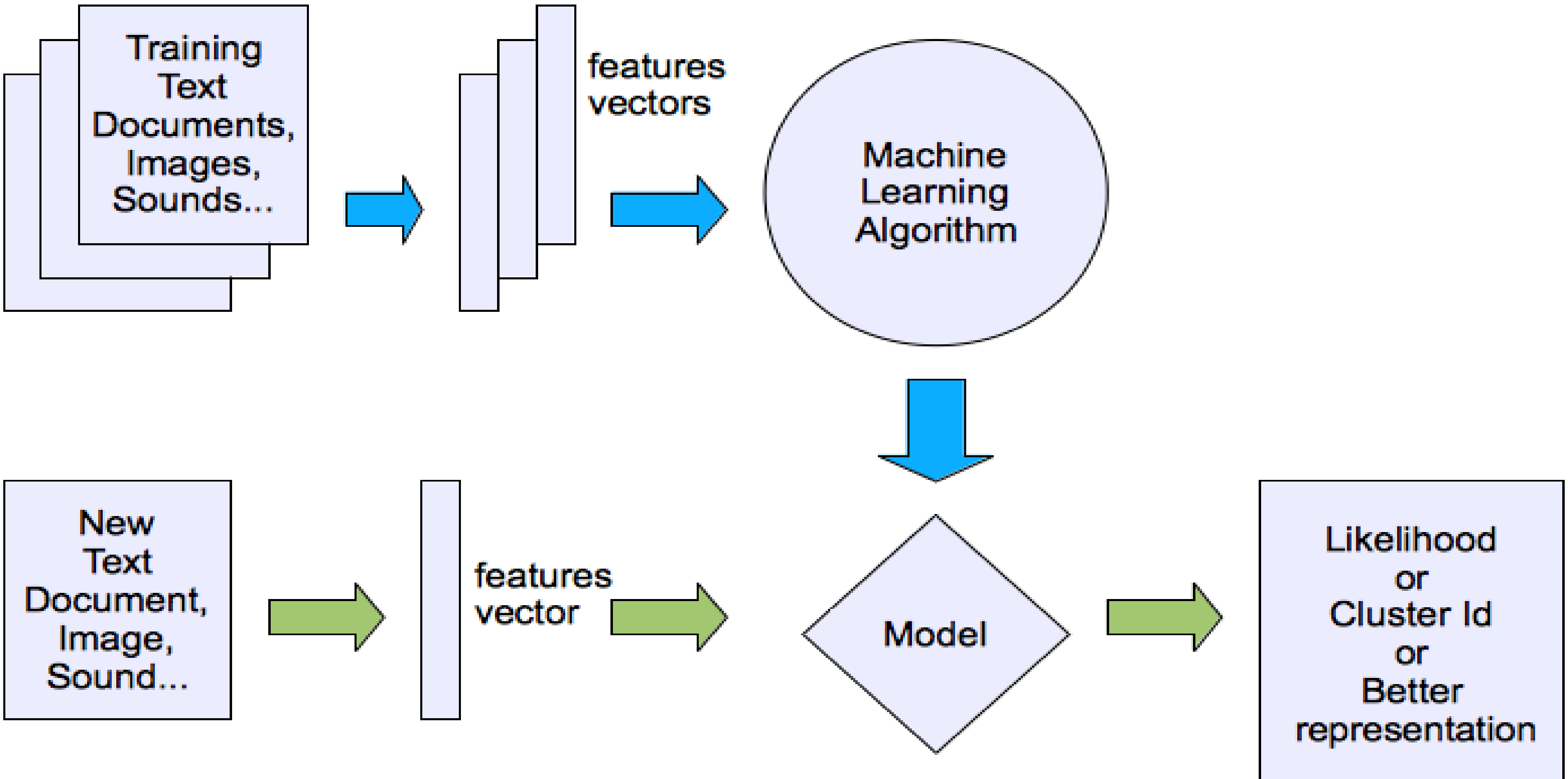
Yi-Fan Chang



Machine learning structure

- **Unsupervised learning**

Yi-Fan Chang



Learning techniques

- Decision trees
- Sets of rules / Logic programs
- Instances
- Graphical models (Bayes/Markov nets)
- Neural networks
- Support vector machines
- Model ensembles
- Etc.

Learning techniques

- Supervised learning categories and techniques
  - **Linear classifier** (numerical functions)
  - **Parametric** (Probabilistic functions)
    - Naïve Bayes, Gaussian discriminant analysis (GDA), Hidden Markov models (HMM), Probabilistic graphical models
  - **Non-parametric** (Instance-based functions)
    - *K*-nearest neighbors, Kernel regression, Kernel density estimation, Local regression
  - **Non-metric** (Symbolic functions)
    - Classification and regression tree, decision tree
  - **Aggregation**
    - Bagging (bootstrap + aggregation), Gradient boost trees, Random forest

Learning techniques

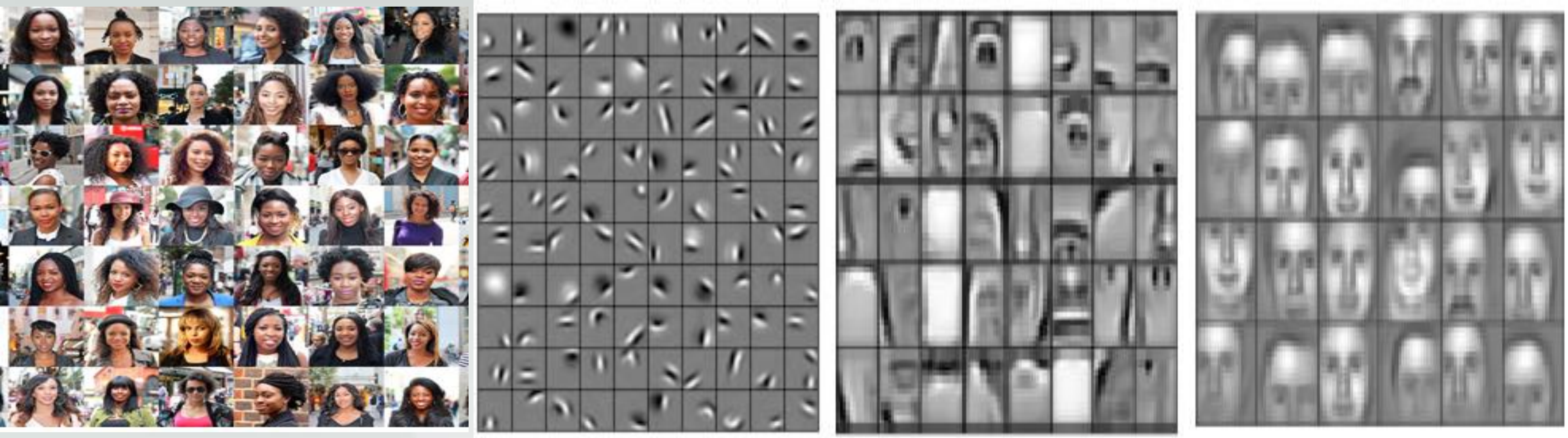
- Unsupervised learning categories and techniques
  - **Clustering**
    - K-means clustering
    - Spectral clustering (Graph Laplacian)
  - **Density Estimation**
    - Gaussian mixture model (GMM)
    - Graphical models
  - **Dimensionality reduction**
    - Principal component analysis (PCA)
    - Factor analysis

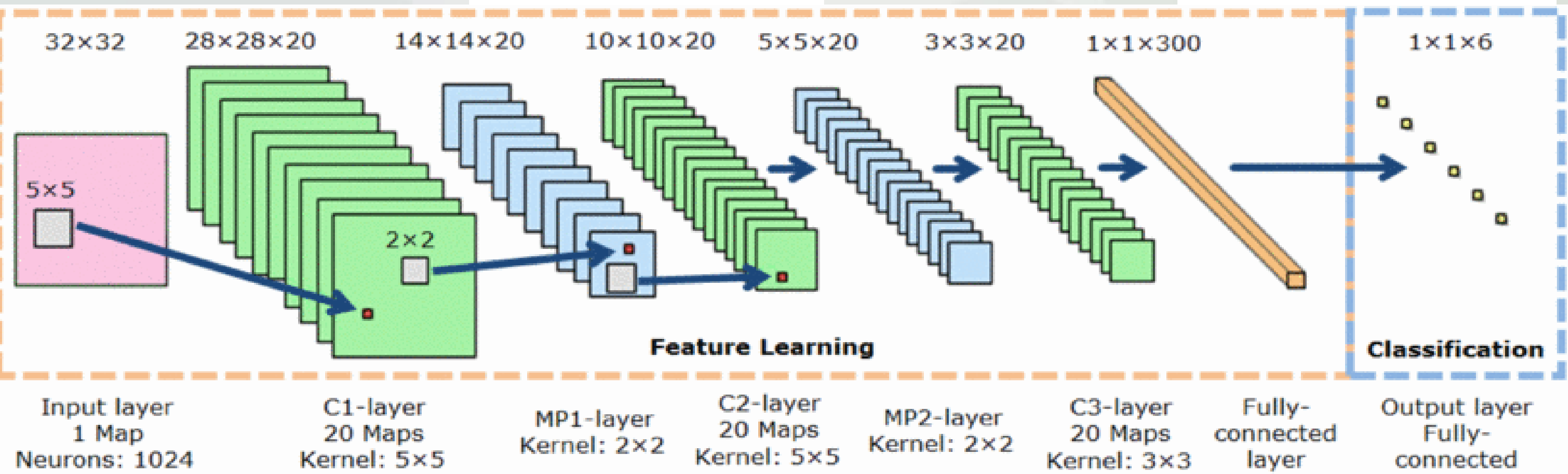
Optimization

- **Combinatorial optimization**
  - E.g.: Greedy search
- **Convex optimization**
  - E.g.: Gradient descent
- **Constrained optimization**
  - E.g.: Linear programming

Deep learning

Fukushima (1980) – Neo-Cognitron; LeCun (1998) – Convolutional Neural Networks (CNN);…





Manifold regularizer for semi-supervised learning

Goal

Labeled data: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)$

Unlabeled data: $\mathbf{x}_{l+1}, \ldots, \mathbf{x}_{l+u}$

Estimate a learner $f: \mathbb{R}^n \rightarrow \mathbb{R}$

Methods

a. **Learning without marginal distribution**
b. **Learning with marginal distribution**
c. **Learning with data-dependent kernel**
d. **Learning with multiscale information**

(Belkin et. al., JMLR 06)

Manifold regularizer (cont'd)

Methods

a. Learning without marginal distribution

$$f^* = \underset{f \in \mathcal{H}}{\operatorname{argmin}} \underbrace{\frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f)}_{\text{Loss function}} + \underbrace{\gamma \|f\|_{\mathcal{H}}^2}_{\text{Penalty term}} \qquad (1)$$

where $\mathcal{H}$ is a Reproducing Kernel Hilbert Space (RKHS) associated with a kernel K.

Common choice for loss function V :

- Squared loss $V = (y - f(\mathbf{x}))^2$ (Regularized Least Square (RLS))
- Hinge loss $V = \max[0, 1 - yf(\mathbf{x})]$ (Support Vector Machine (SVM))

(Belkin et. al., JMLR 06)

Manifold regularizer (cont'd)

Methods

a. Learning without marginal distribution

$$f^* = \underset{f \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f) + \gamma ||f||_{\mathcal{H}}^2$$

The classical Representer Theorem states that

$$f^*(\mathbf{x}) = \sum_{i}^{l} \alpha_i K(\mathbf{x}_i, \mathrm{x})$$

Common choice of kernel K :

- Polynomial kernel: $K(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^T \mathbf{z} + c)^d$
- Radial basis function kernel: $K(\mathbf{x}, \mathbf{z}) = e^{-\gamma ||\mathbf{x} - \mathbf{z}||^2}$

(Belkin et. al., JMLR 06)

Manifold regularizer (cont'd)

Methods

b. Learning with marginal distribution

$$f^* = \underset{f \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{l} \underbrace{\sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f)}_{\text{Loss function}} + \underbrace{\gamma_A ||f||_{\mathcal{H}}^2}_{\text{Penalty term}} + \underbrace{\gamma_I \, ||f||_I^2}_{\text{Additional term for instrinsic geometry}} \quad (2)$$

$||f||_I^2$ can be approximated as

$$||f||_I^2 = \frac{1}{(u+l)^2} \sum_{i,j=1}^{l+u} \left(f(x_i) - f\left(x_j\right)\right) W_{ij} = \frac{1}{(u+l)^2} \mathbf{f}^T \boldsymbol{L}^p \mathbf{f}$$

- $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_{l+u})]^T$
- Graph Laplacian $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$,
- $\boldsymbol{D}$ is diagonal: $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{W}_{ij}$

(Belkin et. al., JMLR 06)

Manifold regularizer (cont'd)

Methods

b. Learning with marginal distribution

$$f^* = \underset{f \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f) + \gamma_A ||f||_{\mathcal{H}}^2 \quad + \gamma_I \, ||f||_I^2$$

The Representer Theorem gives

$$f^*(\mathrm{x}) = \sum_{i=1}^{l+u} \alpha_i K(\mathbf{x}_i, \mathrm{x})$$

(Belkin et. al., JMLR 06)

Manifold regularizer (cont'd)

Methods

c. Learning with data-dependent kernel

$$f^* = \underset{f \in \widetilde{\mathcal{H}}}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f) + \gamma_A ||f||_{\widetilde{\mathcal{H}}}^2 \qquad (3)$$

The minimizer admits

$$f^*(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i \widetilde{K}(\mathbf{x}_i, \mathbf{x})$$

Warped kernel $\widetilde{K}$ defined by

$$\widetilde{K}(\mathbf{x}, \mathbf{z}) = K(\mathbf{x},\mathbf{z}) - \boldsymbol{K}_{\mathbf{X}}^T(\boldsymbol{I} + \boldsymbol{MK})^{-1}\boldsymbol{MK}_{\mathbf{Z}}$$

$$\boldsymbol{K}_{\mathbf{X}} = [K(\mathbf{x}, \mathbf{x}_1), \ldots, K(\mathbf{x}, \mathbf{x}_{l+u})]^T, \boldsymbol{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$$

$\boldsymbol{M}$ is a symmetric positive semi-definite matrix

(Sindhwani et. al., ICML 05)

Manifold regularizer (cont'd)

Methods

c. Learning with data-dependent kernel

$$f^* = \underset{f \in \widetilde{\mathcal{H}}}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f) + \gamma_A ||f||^2_{\widetilde{\mathcal{H}}} \qquad (3)$$

$$\widetilde{K}(\mathbf{x},\mathbf{z}) = K(\mathbf{x},\mathbf{z}) - \boldsymbol{K}_{\mathbf{x}}^T(\boldsymbol{I} + \boldsymbol{M}\boldsymbol{K})^{-1}\boldsymbol{M}\boldsymbol{K}_{\mathbf{z}}$$

By setting $\boldsymbol{M} = \frac{\gamma_I}{\gamma_A}\boldsymbol{L}^p$, one can reconstruct Eq. (2).

Graph Laplacian $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$,

$$\boldsymbol{D} = \text{diag}\{\boldsymbol{D}_{ii} = \sum_j \boldsymbol{W}_{ij}\}$$

(Sindhwani et. al., ICML 05)

Multiscale graph Laplacian based manifold learning

Methods

d. Learning multiscale information

$$f^* = \underset{f \in \widetilde{\mathcal{H}}}{\operatorname{argmin}} \frac{1}{l} \sum_{i=1}^{l} V(\mathbf{x}_i, y_i, f) + \gamma_A ||f||_{\widetilde{\mathcal{H}}}^2$$

$$f^*(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i \widetilde{K}(\mathbf{x}_i, \mathbf{x})$$

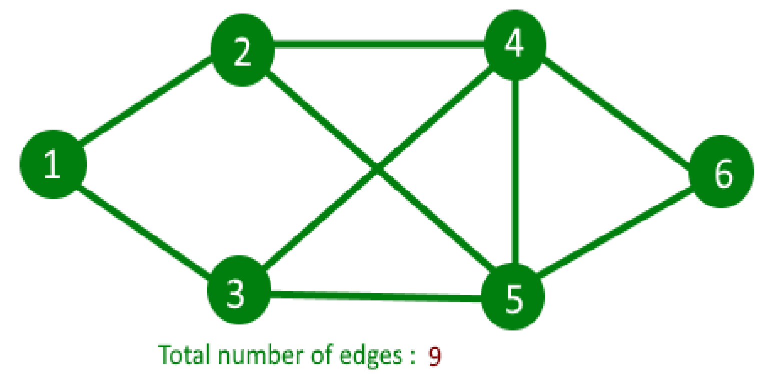
$$\widetilde{K}(\mathbf{x,z}) = K(\mathbf{x,z}) - \boldsymbol{K}_{\mathbf{X}}^T(\boldsymbol{I} + \boldsymbol{MK})^{-1}\boldsymbol{MK}_{\mathbf{Z}}$$

Here M is multiscale kernel.

(Sindhwani et. al., ICML 05)

Multiscale graph

Given graph $G = (V, E)$ *with V* being a set of vertices or nodes or points and E a set of *edges*.



Single scale graph:

$V=\{1,2,3,4,5,6\}$

$E=\{W12, W13, W24, W25, W34, W35, W45, W46, W56\}$

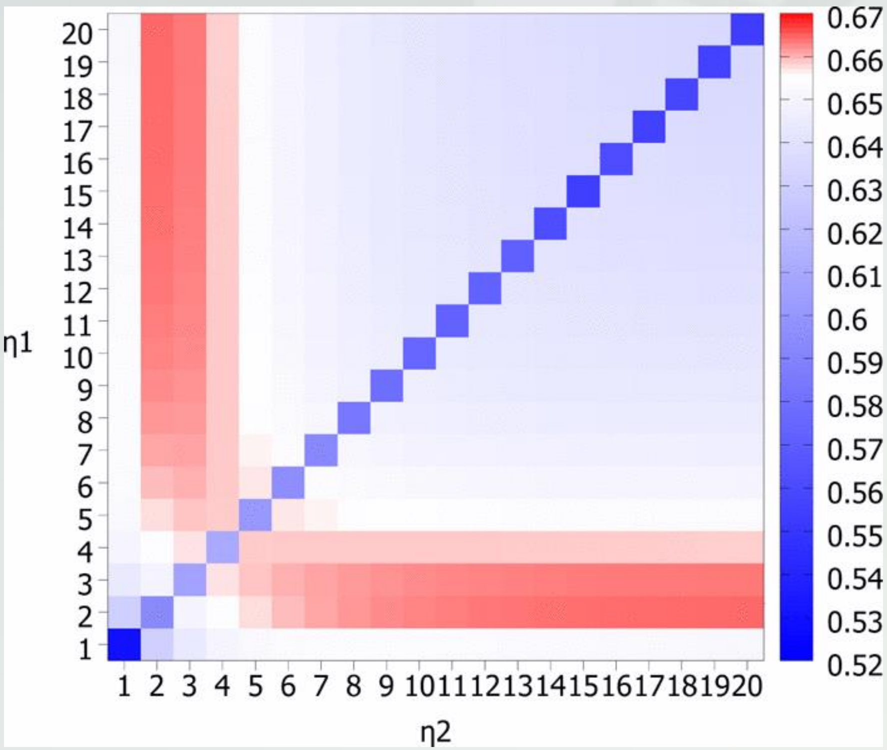
$$[\boldsymbol{W}]_{ij} = e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{2\sigma^2}}$$

(Opron, Xia, Wei, JCP 2015)

Multiscale graph Laplacian

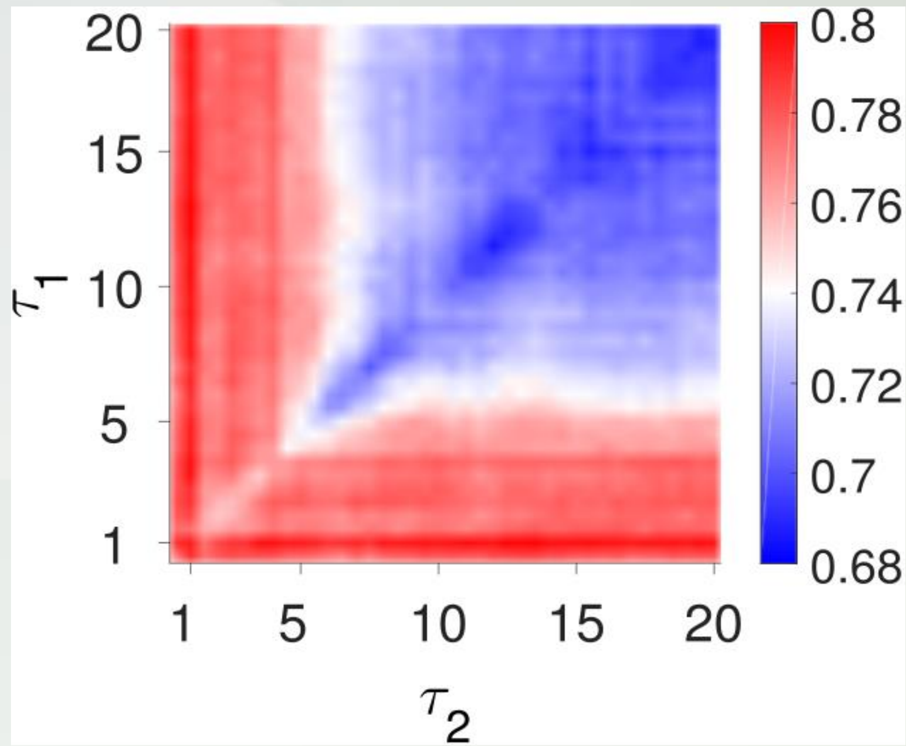
Multiscale graph approach improves performance in protein B-factor prediction and protein-drug binding affinity prediction

B-factor correlation coeff.



(Opron, Xia, Wei, JCP 2015)

Binding affinity correlation coeff.

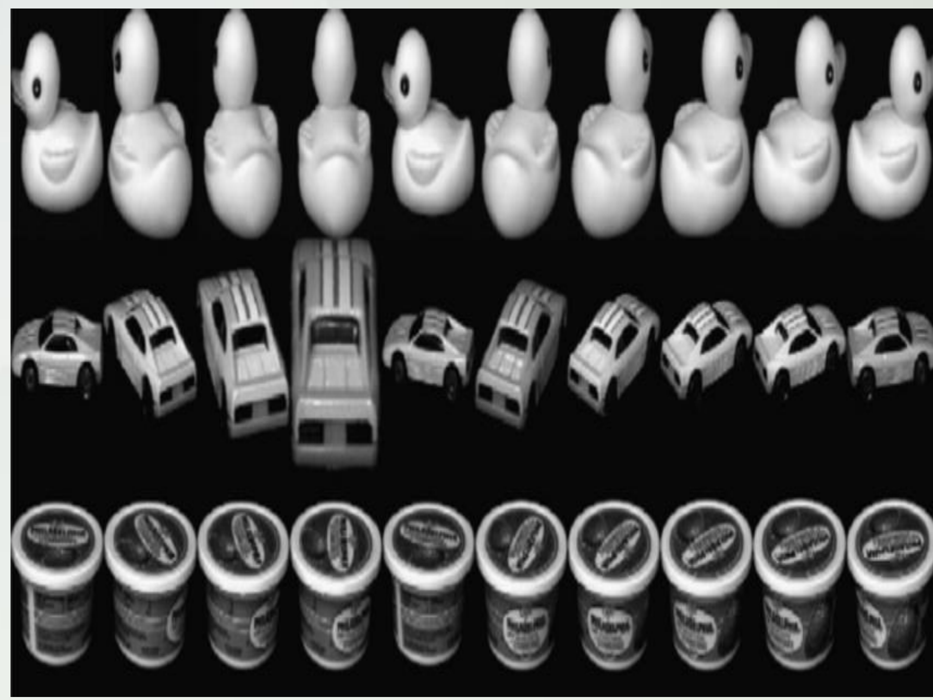


(Nguyen, Xiao, Wang, Wei, JCIM 2017)

Datasets

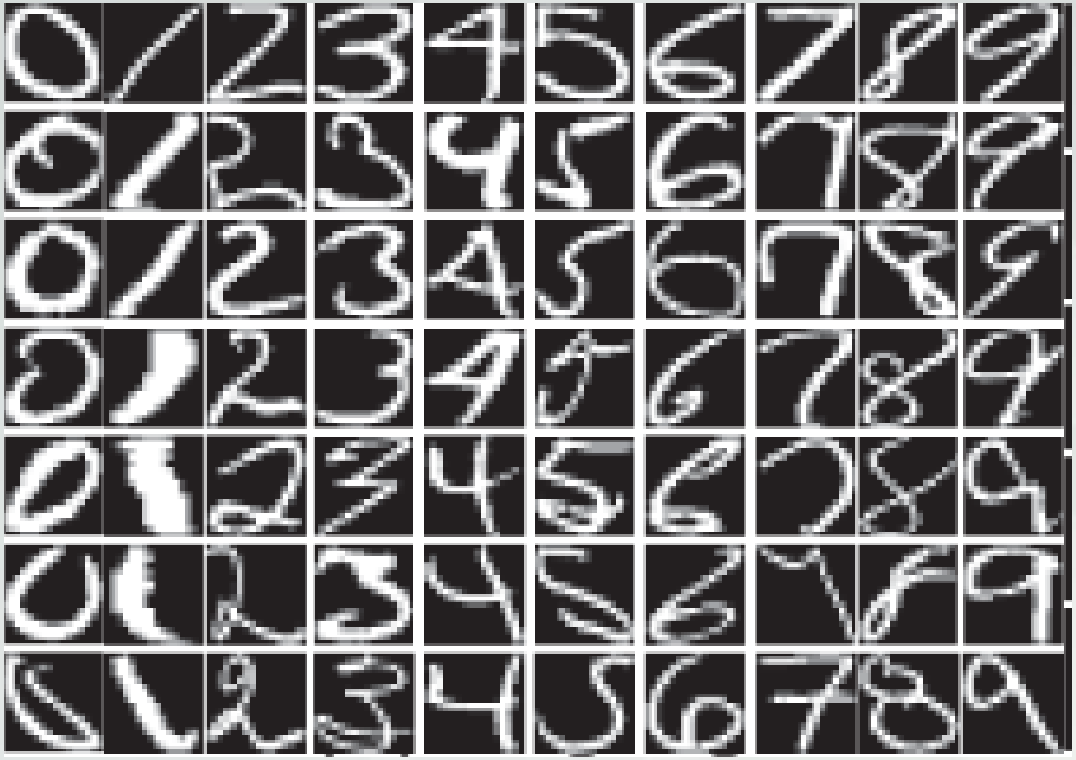
a. **g50c:** an artificial dataset with two classes of point cloud data points generated from two unit covariance normal distributions with equal probability (*Bengio, Grandvalet, NIPS 2004*)

b. **Coil20:** consists of 32x32 gray scale images of 20 objects captured at different angles. (Nene et. al., TR 1996)



Datasets

c. **USPSt:** includes handwritten ten digit images taken from USPS (test) dataset



(Image courtesy of Wang et. al., 2013)

d. **Mac-Windows:** is taken from the 20-newsgroup dataset and is classified into two topics: mac or windows (*Szummer et. al., NIPS 2001*)

Datasets

e. **WebKB:** contains web documents obtained from computer science departments of four universities

It has two categories, namely, course and non-course. There are two ways to describe each web documents: the textual content of web page (called *page* representation), and the anchortext on hyperlinks pointing from other webpages to the current one (link representation). *(Nigam, TR 2001; Joachims, ICML 2003)*

Datasets

Dataset	No. of classes	Sample dim.	No. of data	No. of labeled data
g50c	2	50	550	50
Coil20	20	1024	1440	40
Uspst	10	256	2007	50
Mac-Windows	2	7511	1946	50
WebKB (page)	2	3000	1051	12
WebKB (link)	2	1840	1051	12
WebKB (page+link)	2	4840	1051	12

Results (Error %)

(*Chappele, Zien, AI & Stat. 2005*; *Sindhwani et. al., ICM 2005*)

Dataset → Algorithm ↓	g50c	Coil20	Uspst	Mac-Win	WebKB (link)	WebKB (page)	WebKB (page+link)
Graph-Trans	17.3	6.2	21.3	11.7	22.0	10.7	6.6
TSVM	6.9	26.3	26.5	7.4	14.5	8.6	7.8
Graph-density	8.3	6.4	16.9	10.5	-	-	-
∇TSVM	5.8	17.6	17.6	5.7	-	-	-
LDS	5.6	4.9	15.8	**5.1**	-	-	-
LapSVM	5.4	4.0	12.7	10.4	5.7*	6.6*	5.1*
LapRLS	**5.2**	4.3	12.7	10.0	6.7*	8.9*	5.9*
M-LapSVM (1 ker)	**5.24**	3.62	13.89	10.02	4.51*	4.51*	4.51*
M-LapRLS (1 ker)	5.36	3.62	13.89	10.02	4.51*	4.51*	4.51*
M-LapSVM (2 kers)	5.44	1.48	**9.43**	9.99	4.34*	4.46*	4.32*
M-LapRLS (2 kers)	5.46	1.48	**9.43**	9.96	4.34*	4.46*	4.32*
M-LapSVM (3 kers)	5.44	**1.46**	9.52	9.19	**4.25***	**4.19***	**4.16***
M-LapRLS (3 kers)	5.46	**1.46**	9.52	9.22	**4.25***	4.20*	**4.16***

* : use a sum of Laplacian graphs in each WebKB representation

thank you
danke
рахмат
謝謝
ngiyabonga
spas
teşekkür ederim
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
gracias
хвала
kiitos
dankie
nanni
nandri
dhanyavad
mauruuru
köszönöm
ačiū
dankon
akun
asante
manana
obrigada
tenki
bedankt
bayarlalaa
gràçie
hvala
enkosi
murakoze
djiere dieuf
tau
mochchakkeram
дякую
mamnun
chokrane
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
go raibh maith agat
sobodi
děkuji
obrigado
sagolun
sukriya
kop khun krap
arigatô
takk
dakujem
trugarez
mèsi
didi madloba
ありがとう
tanemirt
grazie
najis tuke
terima kasih
rahmet
kam sah hamnida
rahmat
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерси
তোমাকে ধন্যবাদ
감사합니다
xièxie
merci