UNIVERSITY OF SURREY

UDRC Summer School

Machine Learning

Introduction

Josef Kittler

email: J.Kittler@surrey.ac.uk

www.ee.surrey.ac.uk/Personal/J.Kittler

UNIVERSITY OF SURREY

Course topics

- Machine learning problem
  - Examples
  - Problem formulation
  - Learning scenarios
- Basic linear machines
  - Nearest neighbour classifier
  - Perceptron
  - Sparse representation based classifier
- Nonlinear extensions
  - Kernel methods
  - Multilayer neural networks
  - Deep neural networks
- Dimensionality reduction
- Classifier design issues

UNIVERSITY OF SURREY

Rest of the course

- Training deep neural networks (DNN)
- Advanced DNN

Muhammad Rana

- Recurrent neural networks
- Deep learning libraries

Fei Yan

- Anomaly detection in graphs

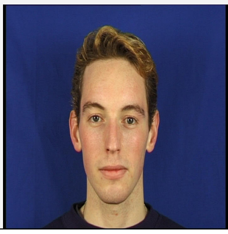
Radek Marik

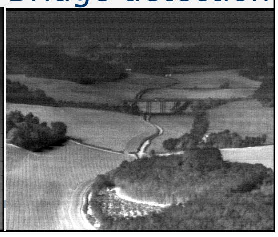
UNIVERSITY OF SURREY

Introduction

- **Machine Learning** is a field of study concerned with the development of algorithms that can learn from and make predictions on data
- **The aim of machine learning** is to give computers the ability to learn (find solutions to problems) without being explicitly programmed
- Applications span a vast range of problems.

Biometrics Object detect Target detect Bridge detection



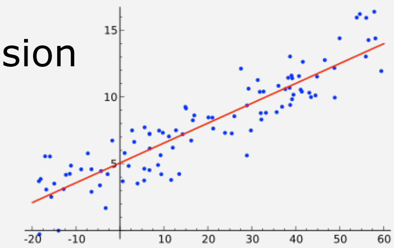



Typical machine learning tasks

- To generate response to input data so as to achieve required functionality
- Examples include
  - Regression (predict output given input)
  - Classification (predict class membership)
  - Cluster assignment (associate input to data structure)
  - Detection (detect a specific object)
  - Anomaly detection (identification of input as an outlier)

Examples of regression

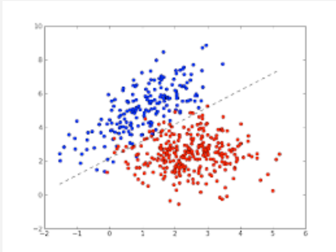
- Simple linear regression



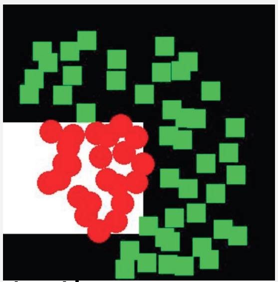
- Multivariate regression

Examples of classification

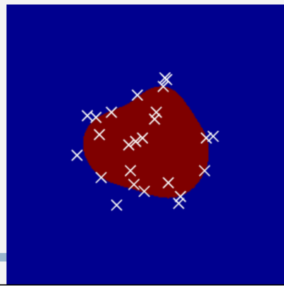
- Linear classification



- Detection



- Anomaly detection



7

General statement of the machine learning problem

- Mathematically, the machine is realising an appropriate function

$$\mathbf{y} = \mathscr{F}(\mathbf{x}, W)$$

  - x…… D dimensional input
  - y…… d dimensional output
  - W….. parameters

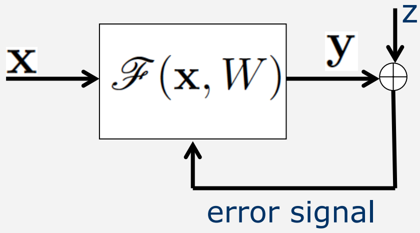
$$\mathbf{x} \longrightarrow \boxed{\mathscr{F}(\mathbf{x}, W)} \longrightarrow \mathbf{y}$$

8

UNIVERSITY OF SURREY

Machine training

- Pre-requisites
  - Training set $X = \{\mathbf{x}_1, \ldots\ldots, \mathbf{x}_N\}$
  - Ground truth target values $Z = \{\mathbf{z}_1, \ldots\ldots, \mathbf{z}_N\}$
  - Form of function $\mathscr{F}$ (architecture)
  - Objective function (error measure)
  - Procedure for updating W



UNIVERSITY OF SURREY

Learning scenarios

- Supervised (target set available)
- Non supervised (target set unavailable)
  - clustering
- Semi-supervised (some training data labelled)
- Transfer learning
  - Supervised learning in the source domain
  - Target domain different from the source domain
  - Only unlabelled data available in the target domain

UNIVERSITY OF SURREY

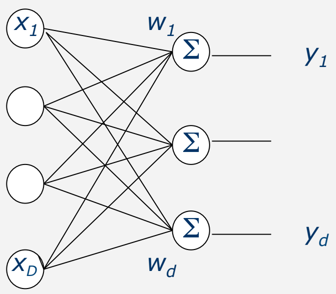
Basic linear machine

- Function $\mathscr{F}$ is linear, namely

$$\mathbf{y} = W\mathbf{x}$$

where W is a d x D matrix of parameters

- $$\mathbf{y} = \begin{bmatrix} w_1^T \\ \cdot \\ \cdot \\ w_d^T \end{bmatrix} \mathbf{x}$$



11

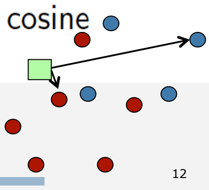
UNIVERSITY OF SURREY

Similarity based classification

- Nearest Neighbour classifier labels patterns based on similarity, gauged in terms of distance
- Squared Euclidean distance between $\mathbf{x}$ and $\mathbf{x}_j$ is given as

$$(\mathbf{x} - \mathbf{x}_j)^T(\mathbf{x} - \mathbf{x}_j) = \mathbf{x}^T\mathbf{x} - 2\mathbf{x}^T\mathbf{x}_j + \mathbf{x}_j^T\mathbf{x}_j \qquad (10)$$

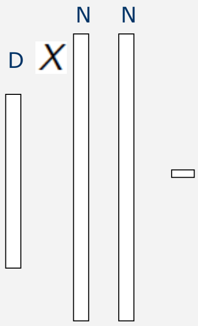
- The distance between $\mathbf{x}$ and $\mathbf{x}_j$ minimal when $\mathbf{x}^T\mathbf{x}_j$ is maximal
- scalar product $\mathbf{x}^T\mathbf{x}_j$ gauges similarity
- other notions of similarity can be defined, e.g. cosine similarity, Gaussian kernel, subjective grading



12

NN classifier machine

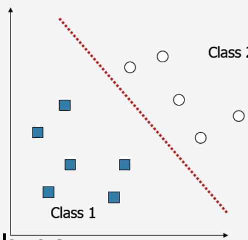
- N… number of training samples
- D… dimensionality of each sample

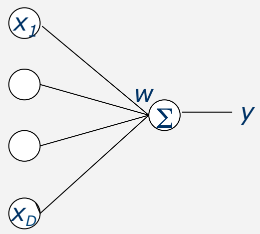


- Augment the result by input layer
- Compute scalar products for all training samples
- Min

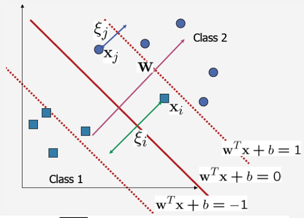
Learning objectives

- Linear classifier

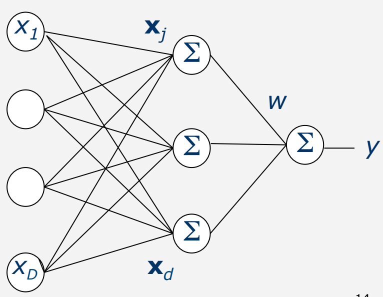




- Support vector machine



$$\mathscr{F}(\mathbf{x}) = \sum_{j\,(\text{support vectors})} w_j z_j (\mathbf{x}_j^T \mathbf{x}) + b$$



UNIVERSITY OF SURREY

Sparse representation classification (SRC)

- Let $X = [\mathbf{x}_1, ...\mathbf{x}_N]$ be training data matrix
- Reconstruct an unknown sample $\mathbf{y}$ as

$$\mathbf{y} = X\mathbf{a}$$

where $\mathbf{a}$ is a vector of coefficients

- Premise: for a sample $\mathbf{y}\epsilon\omega_i$, we would expect the reconstruction to be constituted by training samples from class i, i.e. vector $\mathbf{a}$ should be sparse (all entries for samples from other classes should be zero)
- The solution has to be regularised by imposing a minimum norm on a

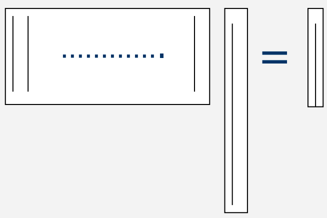
15

UNIVERSITY OF SURREY

SRC

- By imposing sparsity on the reconstruction solution, we should be able to identify the class of $\mathbf{y}$
- This can be achieved by using l_1 and solving

$$argmin||\mathbf{a}||_1 \quad s.t.\mathbf{y} = X\mathbf{a} \tag{7}$$



16

SRC algorithm

UNIVERSITY OF SURREY

1. Solve

$$\underset{}{argmin}||\mathbf{a}||_1 \quad s.t.||\mathbf{y} - X\mathbf{a}|| < \epsilon \tag{8}$$

2. Let $\mathbf{a_i}$ be vector $\mathbf{a}$ with all entries associated with samples from class $j \neq i$ set to zero, and compute the residual

$$r_i(\mathbf{y}) = ||\mathbf{y} - X\mathbf{a}_i||_2 \tag{9}$$

3. assign $\mathbf{y} \rightarrow \omega_i$ if $r_i(\mathbf{y}) = min_j r_j(\mathbf{y})$

- Relationship to the k-NN classifier
  - k-NN classifier minimises the distances to $\mathbf{y}$
  - in addition, SRC involves pairwise interactions of residual error vectors

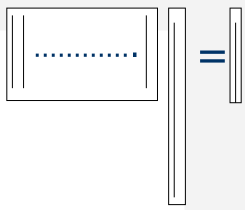
- Solution to be found for every test sample

17

The effect of norm

UNIVERSITY OF SURREY

- ℓ_p norm of $\mathbf{a} = [a_1, ..., a_d]^T$



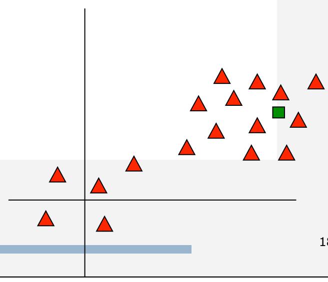
$$\ell_p = [\sum_{j=1}^{d} |a|^p]^{\frac{1}{p}}$$

- ℓ_0 ...... counts the number of nonzero elements

  ℓ_1 ...... induces sparsity

  ℓ_2 ...... length of vector $\mathbf{a}$

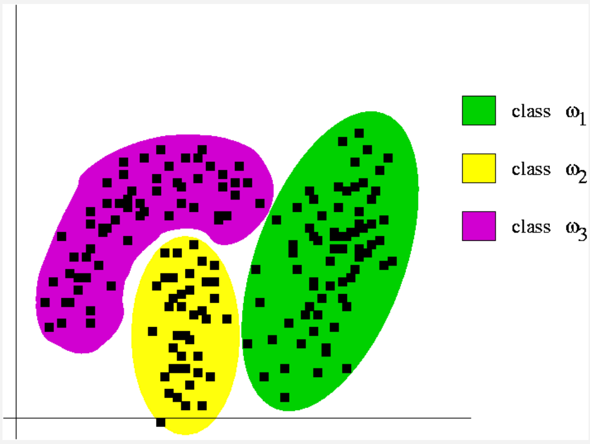
  ℓ_∞ ...... selects $\arg\max_j a_j$



18

Nonlinear separation

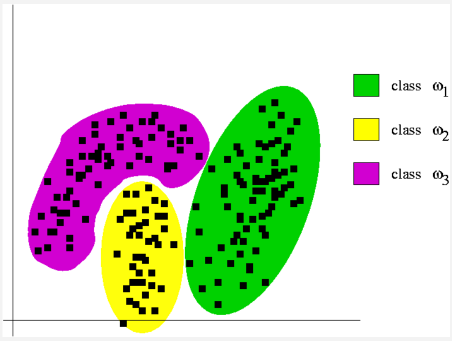
Geometric viewpoint of the pattern recognition problem



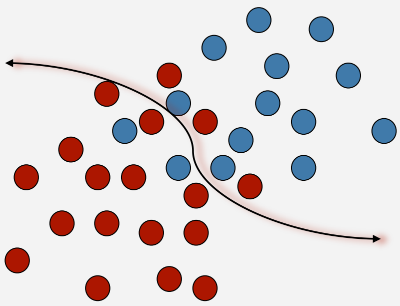
19

Overlapping classes

- Class boundaries may be nonlinear



- Classes may be overlapping



20

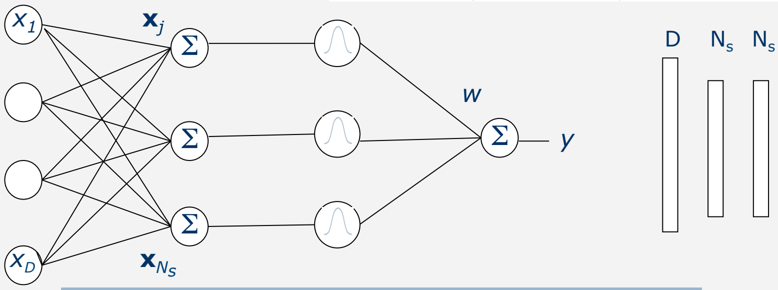
Kernel SVM

- Scalar product replaced by a kernel function

$k(\mathbf{x}_j, \mathbf{x})$

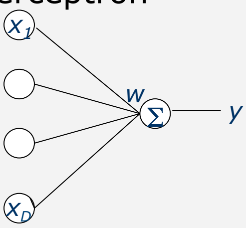
for example, a radial basis function

$$\mathscr{F}(\mathbf{x}) = \sum_{j\,(\text{support vectors})} w_j z_j k(\mathbf{x}_j, \mathbf{x}) + b$$

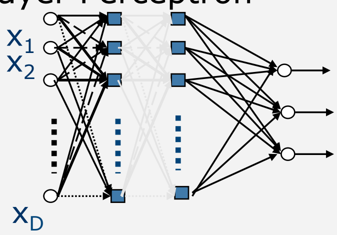


Neural networks

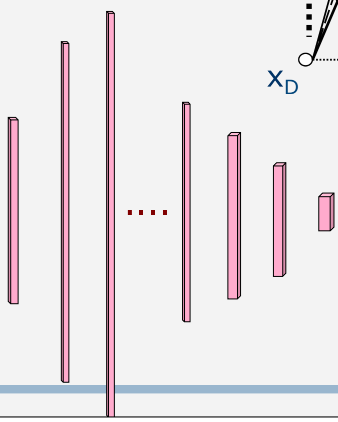
- Perceptron



Multilayer Perceptron

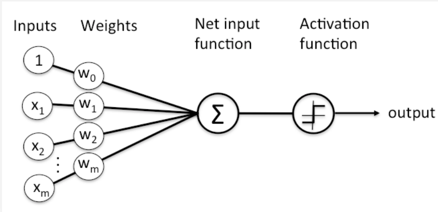


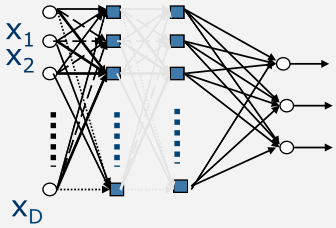
- Deep neural network



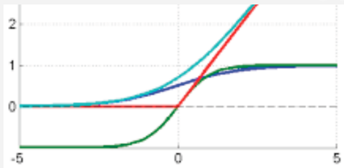
Node expansion motivation
- Ensemble
- Generate different terms

Neural networks





- Activation function
  - sigmoid
  - rectified linear
  - tanh
  - softplus
  - radial basis



23

Essence of machine learning

- Defining the structure of the approximation of function $\mathcal{F}$
- Making the computation of $\mathcal{F}$ robust
- Finding the values of the unknown parameters to achieve the desired objectives (system behaviour)
  - Avoiding local optima
  - Promoting the ability of the solution to generalise to unseen data
  - Key measure
    - Dimensionality reduction
    - Training set management
    - Ensembles (dropouts)

24

Classifier design issues

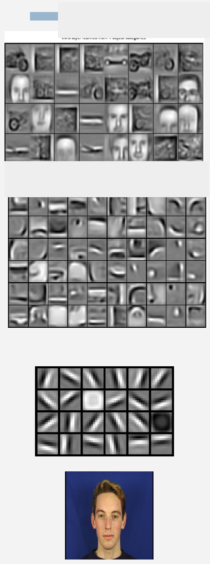
- System architecture
  - # of layers
  - connectivity
  - role of layer
  - dropouts
- Activation function
- Objective function
  - primary objective
  - constraints
  - parameters to be learned
  - regularisation of solution
  - metric
  - fusion
- Training data usage
  - training/testing
  - evalutaion
  - augmentation
    - Mirroring
    - Model based
    - Random sampling
    - Perturbation
- Learning process and its parameters
  - Epoch
  - Learning rate
  - Greedy learning

Primary objective function

- Choice very important, as it induces different learning properties
- Examples
  - Classification error
  - Mean squared error $[\mathbf{y}-\mathbf{z}]^T[\mathbf{y}-\mathbf{z}]$
  - Cross entropy $-\sum_{j=1}^{m} y_j \log z_j$

UNIVERSITY OF SURREY

Dimensionality reduction



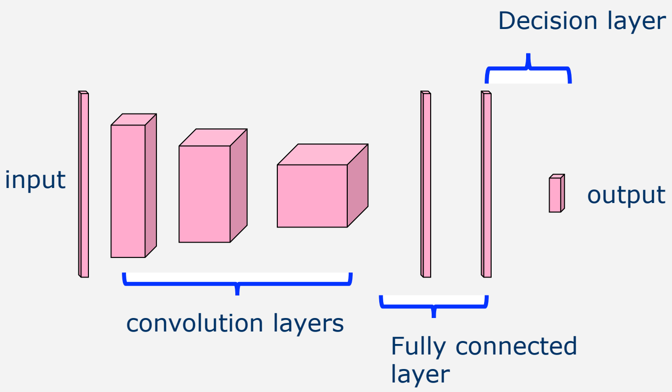
- Motivation – reduce over fitting
- Scope
  - Redundancy
  - Irrelevant content
  - Hierarchical relations – from local to global
- Implications on architecture
  - # of relevant features is low
  - At lower layers only local filters are needed: restricted connectivity
  - translation invariance -

27

UNIVERSITY OF SURREY

Convolutional neural network

- Basic CNN architecture



28

Dimensionality reduction - PCA

- Principal component analysis (PCA): an orthogonal basis transformation
- Transform correlated variables into uncorrelated ones (principal components)
- Can be used for dimensionality reduction
- The underlying aim is to find bases V and transformed variables $\mathbf{y}$ so that in expectation $|\mathbf{x} - V\mathbf{y}|_2$ is minimal subject to $V^T V = I$
- Retains as much variance as possible when reducing dimensionality

How PCA works

- Given m centred vectors: $X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$
  - X: $D \times N$ data matrix
- The squared fitting error averaged over the training set

$$tr\{[X - VV^T X]^T [X - VV^T X]\}$$

- This can be rearranged as

$$tr\{[X^T X - X^T VV^T X]\}$$

- The solution to the constrained optimisation problem is a system of eigenvectors and eigenvalues satisfying

$$XX^T V - V\Lambda = 0$$

- Note $C = XX^T$ is the $D \times D$ covariance matrix
- Diagonal matrix Λ: eigenvalues

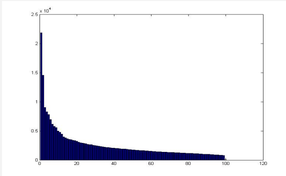
Mapping data to low dimensional feature space

- Given $V=[v_1, v_2, \ldots v_d]$, the d-dimensional feature vector in the PCA space is given by

$$y = V^T x$$

- dimensionality d chosen to retain a certain fraction of variance
- d<D
- d<= N

Distribution of eigenvalues



Example of eigenvectors

1

Kernelising PCA

- Premultiplying eq (1) by data matrix X^T gives

$$X^T X X^T V - X^T V \Lambda = 0 \qquad (2)$$

- an eigenvalue problem with eigenvectors $U = X^T V$ and the same eigenvalues
- matrix $X^T X$ is $N \times N$
- if $N << D$, then solving eq (2) is more efficient
- matrix V can be obtained as $V = XU\Lambda^{-1}$
- matrix $K = X^T X$ is referred to as kernel matrix
- its element $k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ measures "similarity" of vectors $\mathbf{x}_i, \mathbf{x}_j$
- $k_{ij} = \mathbf{x}_i^T \mathbf{x}_j$

32

Kernelising PCA

- the kernel formulation of PCA offers a scope for using different notions of similarity
- for instance, $k_{ij} = \exp\{-||\mathbf{x}_i - \mathbf{x}_j||^2\sigma^{-2}\}$
- other notions of similarity remap $\mathbf{x}$ in a nonlinear way into $\phi(\mathbf{x})$
- $\phi(\mathbf{x})$ may be of infinite dimensionality
- $\phi(\mathbf{x})$ is defined implicitly, is unknown, but satisfies $k_{ij} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$
- working with K is like working with data matrix $X = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \cdots, \phi(\mathbf{x}_N)]$
- We have kernelised PCA

33

Kernelising PCA

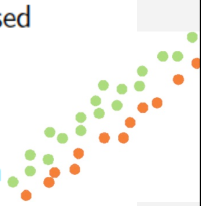
- Note
  - $V = XU\Lambda^{-1}$ is not explicitly available: U and Λ are, but X is not
- However... we are interested in projection onto basis V, not the basis itself
- Projection onto V: $X^TV = X^TXU\Lambda^{-1} = KU\Lambda^{-1}$
- All K and U and Λ are available
- Λ purely rescales the data and can be omitted

34

UNIVERSITY OF SURREY

Kernel Discriminant Analysis

- Kernel Fisher discriminant analysis: another supervised learning technique
- Focusing on discriminantion, rather than faithful representation
- Seeking the projection **w** maximising Fisher criterion



$$\max_{\mathbf{w}} \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T (S_T + \lambda I)\mathbf{w}} \tag{3}$$

- S_B and S_T: between class and total scatters
- λ: regularisation parameter
- The total scatter matrix equals mixture covariance matrix $S_T = XX^T$
- Between class scatter S_B can be expressed as $S_B = X\Delta X^T$
- Block diagonal matrix Δ contains a constant in block i, proportional to the number of samples from class i

35

UNIVERSITY OF SURREY

KDA

- Expressing $\mathbf{w} = X\alpha$, and substituting for S_B, we can kernelise Fisher criterion as

$$\max_{\alpha} \frac{\alpha^T X^T X \Delta X^T X \alpha}{\alpha^T X^T (XX^T + \lambda I) X\alpha} = \max_{\alpha} \frac{\alpha^T K \Delta K \alpha}{\alpha^T (K^2 + \lambda K)\alpha} \tag{4}$$

- The maximisation leads to an eigenvalue problem

$$\Delta K \alpha - \kappa (K + \lambda I)\alpha = 0 \tag{5}$$

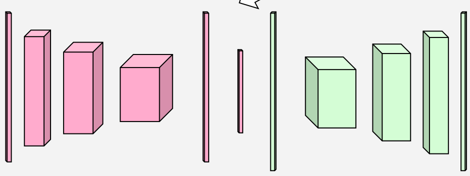
Examples of DNN feature extraction approaches

- Auto-encoder: find features that can regenerate the input

Feature layer

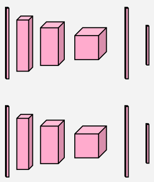
Learning objective
❖ output=input

input

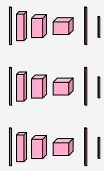


output

- Extracting representation that outputs similar/dissimilar features for similar/dissimilar inputs



siamese network

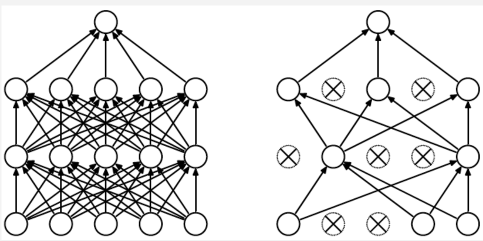


tripple loss

40

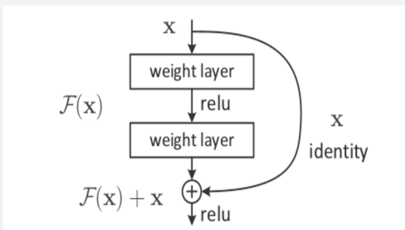
Learning and generalisation enhancements

- Node dropout



N Srivastava et al., Dropout: A simple way to prevent neural networks from overfitting, JMLR 2014

- Residual network – vanishing gradient problem



41

Metric learning

- The underlying assumption in pattern recognition is that similar patterns belong to the same class.
- This is the premise behind Nearest Neighbour methods
- Also Gaussian classifier assign patterns to their classes based on similarity (distance from the class mean)
- However, the choice of metric used for measuring distances is critical

42

Metric learning

- Desired properties of metric

- In general, we would like our metric $d_A(\mathbf{x}_i, \mathbf{x}_j)$ to reflect the notion of similarity, i.e.

$$\begin{array}{llll} d_A(\mathbf{x}_i, \mathbf{x}_j) = \text{small} & \leq t_s & \text{similar pair} \\ d_A(\mathbf{x}_i, \mathbf{x}_j) = \text{large} & \geq t_l & \text{dissimilar pair} \end{array} \tag{13}$$

  where threshold t_s is small, and t_l is large
- Let metric $d_A(\mathbf{x}_i, \mathbf{x}_j)$ be parameterised as

$$d_A(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T A(\mathbf{x}_i - \mathbf{x}_j) \tag{14}$$

43

Problem formulation

- Thus the metric learning problem can be formulated as

$$\begin{aligned} \min_A \quad & [tr(AA_0^{-1}) - \log|AA_0^{-1}|] \\ \text{s.t.} \quad & tr[A(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T] \leq t_s \quad (i,j)\epsilon S \\ & tr[A(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T] \geq t_l \quad (i,j)\epsilon D \end{aligned} \tag{17}$$

- As there may not exist a feasible solution, the optimisation problem can be relaxed using slack variables $\xi_{c(i,j)}$ These replace the constraint t_x for the (i,j) pair indexed by $c(i,j)$, where $1 \leq c \leq n_S$ for the similar pairs, and $n_S + 1 \leq c \leq n$ for the dissimilar pair.