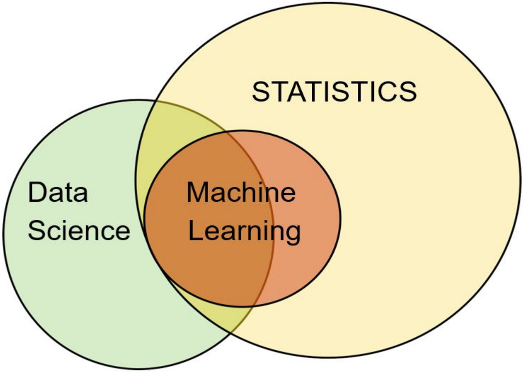
Hybrid Prediction Models: Statistical Perspectives and Applications

Tanujit Chakraborty

Senior Research Fellow,
Statistical Quality Control & Operations Research Unit,
Indian Statistical Institute, Kolkata, India.

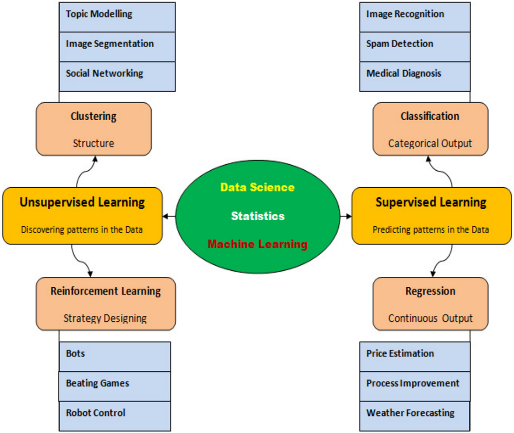
Introduction: An overview of Data Science Problems



"**Statistics** is the universal tool of inductive inference, research in natural and social sciences, and technological applications. Statistics, therefore, must always have purpose, either in the pursuit of knowledge or in the promotion of human welfare" - **Prasanta Chandra Mahalanobis**, Father of Statistics in India, 1956

Introduction: An overview of Data Science Problems

"**Machine learning** is the field of study that gives computers the ability to learn without being explicitly programmed" - **Arthur L. Samuel**, AI pioneer, 1959.



"**Prediction** is very difficult, especially if it's about the future"- **Niels Bohr**, Father of Quantum Mechanics.

Introduction: Developments of Supervised Learning Models

- Linear Regression (Galton, 1875).
- Linear Discriminant Analysis (R.A. Fisher, 1936).
- Logistic Regression (Berkson, JASA, 1944).
- k-Nearest Neighbor (Fix Hodges, 1951).
- Parzens Density Estimation (E Parzen, AMS, 1962)
- Classification and Regression Tree (Breiman et al., 1984).
- Artificial Neural Network (Rumelhart et al., 1985).
- Perceptron Trees (Paul Utgoff, 1989, Connection Science).
- MARS (Friedman, 1991, Annals of Statistics).
- SVM (Cortes Vapnik, Machine learning, 1995)
- Random forest (Breiman, 2001).
- Deep convolutional neural nets (Krizhevsky, Sutskever, Hinton, NIPS 2012).
- Generative Adversial Nets (Ian Goodfellow et al., NIPS 2014).
- Deep Learning (LeCun, Bengio, Hinton, Nature 2015).
- Bayesian deep neural network (Yarin Gal, Islam, Zoubin Ghahramani, ICML 2017).

Introduction: k-Nearest Neighbor

- kNN does not build model from the training data.
- To classify a test instance d, define k-neighborhood P as k nearest neighbors of d.
- Count number n of training instances in P that belong to class c_j.
- Estimate $\Pr(c_j|d)$ as n/k.
- No training is needed. Classification time is linear in training set size for each test case. (Cover & Hart, IEEE IT 1967).

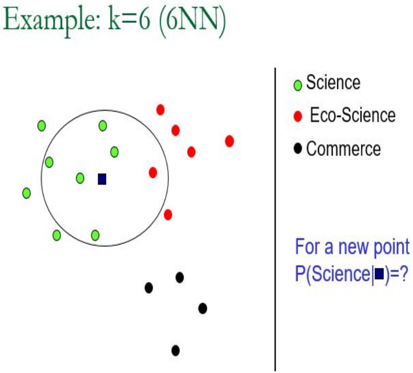


Fig: An example of k-NN

Introduction: k-Nearest Neighbor

Definition (the kNN classier)

The kNN classier: decide f(X) by majority vote among the labels of the training points in the k-nearest neighborhood of X.

- k is usually chosen empirically via a validation set or cross-validation by trying a range of k values.
- Distance function play a crucial role, but depends on applications.
- kNN can deal with complex and arbitrary decision boundaries.
- kNN is slow at the classification time.
- kNN does not produce an understandable model.
- To avoid ties: chooses k odd.

Introduction: Classification Problem

- Statistical learning theory (SLT) studies mathematical foundations for machine learning models, originated in late 1960s.
- Input space (object space): X ; Output space (label space): Y
- The task: to classify objects in X into categories in Y
- binary classification: to classify objects in X into 2 classes in label space Y $= \{0, 1\}$.
- Given (object, label), The goal: to find a classifier $f : X \to Y$ to predict the label of new object X
- A learning algorithm L: inputs training data, outputs a classifier f
- No assumption is made on the joint probability distribution of data μ.
- The goal is to learn a classifier $f : X \to Y$: "how good" a function f is when used as a classifier?

Introduction: Loss Function and Risk

- Introduce a loss function: Given $(X, Y) \in \mathbb{R}^d \times \{0,1\}$, an unknown μ and a classifier $f : \mathbb{R}^d \rightarrow \{0,1\}$, the loss function is defined by: $l_\mu(X, Y, f(X)) = \mu\{f(X) \neq Y\}$.
- The risk or mis-classification error is the average loss over all $X \in \mathbb{R}^d$ $R(f) := E(l(X, Y, f(X)))$
- The risk counts how many elements of the instance space X are misclassified by the classifier f. Smaller the risk, better the classifier.
- The Bayes error is the smallest possible risk over all possible classifiers: $R^* = R^*(\mu) = inf_f\{R_\mu(f)\}$.

  Given μ, the optimal classifier - Bayes classifier is defined as:

$$f_{Bayes}(x) = \begin{cases} 0, & \text{if } \psi(x) \geq 1/2. \\ 1, & \text{otherwise.} \end{cases} \tag{1.1}$$

- It is impossible to compute the Bayes classifier: μ is unknown, but we need to evaluate $\psi(x) = \mu(Y = 1|X = x)$.

Introduction: Consistency and Universal Consistency

Consistency: A learning rule, when presented more and more training examples, → the optimal solution.

Definition (Consistent)

Given an infinite sequence of training points $(X_i, Y_i)_{i \in N}$ *with* μ. *For each* $n \in N$, *let* f_n *be a classifier for the first* n *training points. The learning algorithm is called consistent with respect to* μ *if the risk* $l(f_n)$ *converges to the risk* $l(f_{Bayes})$, *that is for all* $\epsilon > 0$,

$$\mu(R(f_n) - R(f_{Bayes}) > \epsilon) \to 0 \text{ as } n \to \infty.$$

Definition (Universally Consistent)

The learning algorithm is called universally consistent if it is consistent for all probability distributions μ.

Introduction: Consistency result for kNN

Remind: the classifier is called universally consistent for all μ if the risk $R(f_n)$ converges to the risk $R(f_{Bayes})$.

Theorem (Stone, 1977, The annals of statistics)

Let $k \to \infty$, $n \to \infty$, and $k/n \to 0$. Then the k-NN classifier in R^n with Euclidean distance is universally consistent.

- We can choose k such that it grows slowly with n. For example. if one chooses $k = log(n)$, the kNN classification rule is universally consistent.
- However, this theorem can not be generalized to infinite dimension.

Introduction: Decision Trees

- Decision tree is defined by a hierarchy of rules (in form of a tree).
- Rules from the internal nodes of the tree are called root nodes
- Each rule (internal node) tests the value of some feature.
- Labeled training data is used to construct the Decision tree. The tree need not to be always a binary tree.
- CART (Breiman et al., 1984), RF (Breiman, 2001), BART (Chipman et al., 2010).

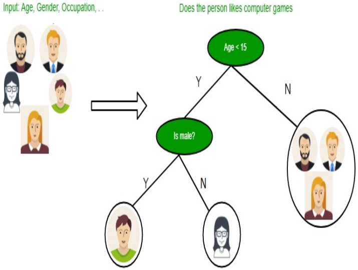

Input: Age, Gender, Occupation, . .
Does the person likes computer games
Age < 15
Y
N
Is male?
Y
N

Fig: An example of a Classification Tree

Introduction: Decision Trees

- CART is a greedy divide-and-conquer algorithm. Trees are constructed in a top-down recursive manner based on selected attributes.
- Attributes are selected on the basis of an impurity function (e.g., IG for Classification MSE for Regression).
- **Pros:** Built-in feature selection mechanism, Comprehensible, easy to design, easy to implement, good for structural learning.
- **Cons:** But may become large for complex problems, too many rules loose interpretability, risk of over-fitting, sticking to local minima.

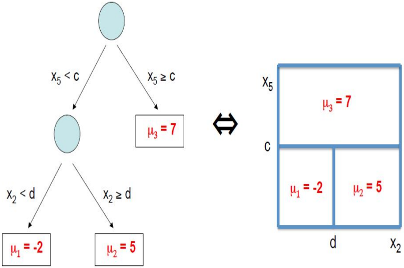


Fig: An example of a Regression Tree

Introduction: Statistical Theory to Decision Trees

- Consistency of data driven histogram methods (Nobel, 1996, Annals of Statistics).
- A Fast, Bottom-Up Decision Tree Pruning Algorithm with Near-Optimal Generalization (Kearns, Mansour, ICML, 1998)
- Generalization Bounds for Decision Trees (Mansour et al., 2000, COLT).
- Analysis of a Complexity-Based Pruning Scheme for CT (Nobel, 2002, IEEE Information Theory).
- Consistency of Online Random Forest (Denil et al., 2013, ICML).
- Consistency of Random Forest (Scornet et al., 2015, Ann. Stat.).

Introduction: Consistency Results for Tree Based Model

Theorem (Lugosi, Nobel, 1996, Annals of Statistics)

Let $(\underline{X}, \underline{Y})$ be a random vector taking values in $\mathbb{R}^p \times C$ and L be the set of first n outcomes of $(\underline{X}, \underline{Y})$. Suppose that Φ is a partition and classification scheme such that $\Phi(L) = (\psi_{pl} \circ \phi)(L)$, where ψ_{pl} is the plurality rule and $\phi(L) = (L)_{\tilde{\Omega}_n}$ for some $\tilde{\Omega}_n \in \mathcal{T}_n$, where $\mathcal{T}_n = \{\phi(\ell_n) : P(L = \ell_n) > 0\}$. Also suppose that all the binary split functions in the question set associated with Φ are hyperplane splits. As $n \to \infty$, if the following regularity conditions hold:

$$\frac{\lambda(\mathcal{T}_n)}{n} \to 0 \tag{1.2}$$

$$\frac{log(\triangle_n(\mathcal{T}_n))}{n} \to 0 \tag{1.3}$$

and for every $\gamma > 0$ and $\delta \in (0, 1)$,

$$\inf_{S \subseteq \mathbb{R}^p : \eta_x(S) \geq 1-\delta} \eta_x(x : diam(\tilde{\Omega}_n[x] \cap S) > \gamma) \to 0 \tag{1.4}$$

with probability 1. then Φ is risk consistent.

- Equation (1.1) is the sub-linear growth of the number of cells, Equation (1.2) is the sub-exponential growth of a combinatorial complexity measure, and Equation (1.3) is the shrinking cell condition.

- The process defined above is binary in the sense that each application of the function ϕ splits each node in a partition into two or fewer child nodes. (See Figure below).

Introduction: Graphical Interpretation

- Let $\underline{X}$ be the space of all possible values of p features and C be the set of all possible binary outcomes. We are given a training sample with n observations $L = \{(X_1, C_1), (X_2, C_2), \ldots, (X_n, C_n)\}$, where $X_i = (X_{i1}, X_{i2}, \ldots, X_{ip}) \in \underline{X}$ and $C_i \in C$.
- Also let $\Omega = \{\omega_1, \omega_2, \ldots, \omega_k\}$ be a partition of the feature space $\underline{X}$. We denote $\widetilde{\Omega}$ as one such partition of Ω. Define $L_{\omega_i} = \{(X_i, C_i) \in L : X_i \in \omega_i, C_i \in C\}$ as the subset of L induced by ω_i and let $L_{\widetilde{\Omega}}$ denote the partition of L induced by $\widetilde{\Omega}$.
- Now, let us define $\widehat{L}$ to be the space of all learning samples and $\mathbb{D}$ be the space of all partitioning classification function, then $\Phi : \widehat{L} \to \mathbb{D}$ such that $\Phi(L) = (\psi \circ \phi)(L)$, where ϕ maps L to some induced partition $(L)_{\widetilde{\Omega}}$ and ψ is an assigning rule which maps $(L)_{\widetilde{\Omega}}$ to d on the partition $\widetilde{\Omega}$.
- The most basic reasonable assigning rule ψ is the plurality rule $\psi_{pl}(L_{\widetilde{\Omega}}) = d$ such that if $x \in \omega_i$, then $d(\underline{x}) = \arg\max_{c \in C} |L_{c,\omega_i}|$.
- For any random variable X and set A, let $\eta_{n,X}(A) = \frac{1}{n} \sum_{i=1}^{n} I(X_i \in A)$ be the empirical probability that $X \in A$ based on n observations and I denotes the indicator function.

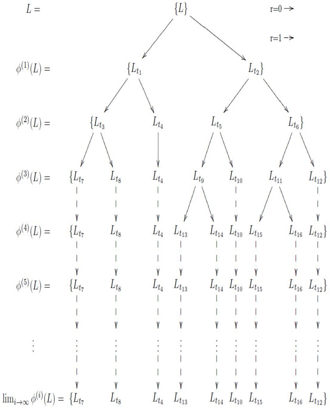

$L =$
$\{L\}$
r=0 →
r=1 →
$\phi^{(1)}(L) =$ $\{L_{t_1}$ $L_{t_2}\}$
$\phi^{(2)}(L) =$ $\{L_{t_3}$ L_{t_4} L_{t_5} $L_{t_6}\}$
$\phi^{(3)}(L) =$ $\{L_{t_7}$ L_{t_8} L_{t_4} L_{t_9} $L_{t_{10}}$ $L_{t_{11}}$ $L_{t_{12}}\}$
$\phi^{(4)}(L) =$ $\{L_{t_7}$ L_{t_8} L_{t_4} $L_{t_{13}}$ $L_{t_{14}}$ $L_{t_{10}}$ $L_{t_{15}}$ $L_{t_{16}}$ $L_{t_{12}}\}$
$\phi^{(5)}(L) =$ $\{L_{t_7}$ L_{t_8} L_{t_4} $L_{t_{13}}$ $L_{t_{14}}$ $L_{t_{10}}$ $L_{t_{15}}$ $L_{t_{16}}$ $L_{t_{12}}\}$
$\lim_{i\to\infty} \phi^{(i)}(L) =$ $\{L_{t_7}$ L_{t_8} L_{t_4} $L_{t_{13}}$ $L_{t_{14}}$ $L_{t_{10}}$ $L_{t_{15}}$ $L_{t_{16}}$ $L_{t_{12}}\}$

Fig: Graphical interpretation of tree structured model.

Introduction: Developments of Neural Nets

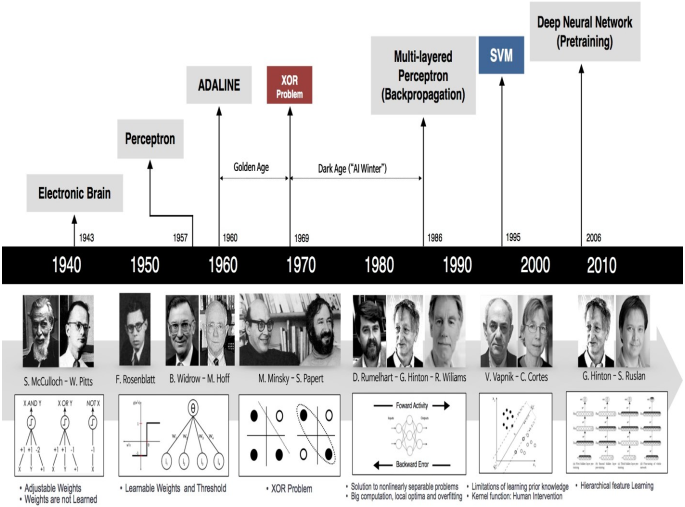

Deep Neural Network (Pretraining)
Multi-layered Perceptron (Backpropagation)
SVM
ADALINE
XOR Problem
Perceptron
Electronic Brain
Golden Age
Dark Age ("AI Winter")
1943
1957
1960
1969
1986
1995
2006
1940
1950
1960
1970
1980
1990
2000
2010
S. McCulloch - W. Pitts
F. Rosenblatt
B. Widrow - M. Hoff
M. Minsky - S. Papert
D. Rumelhart - G. Hinton - R. Wiliams
V. Vapnik - C. Cortes
G. Hinton - S. Ruslan
X AND Y
X OR Y
NOT X
Foward Activity
Backward Error
• Adjustable Weights
• Weights are not Learned
• Learnable Weights and Threshold
• XOR Problem
• Solution to nonlinearly separable problems
• Big computation, local optima and overfitting
• Limitations of learning prior knowledge
• Kernel function: Human Intervention
• Hierarchical feature Learning

Fig: Developments of Neural Network Models

Introduction: Artificial Neural Networks

- ANN is composed of several perceptron-like units arranged in multiple layers.
- Consists of an input layer, one or more hidden layer, and an output layer.
- Nodes in the hidden layers compute a nonlinear transform of the inputs.
- Also called a Feedforward Neural Network (since there is no backward connections between layers, viz., no loops).
- Note: All nodes between layers are assumed to be connected with each other.

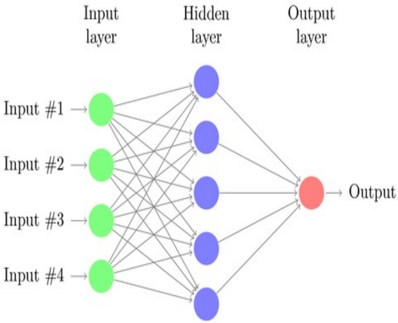


Fig: FFNN Model with one hidden layer with 5 hidden units

Introduction: Artificial Neural Networks

- **Universal Approximation Theorem (Hornik, 1989)**: A one hidden layer FFNN with sufficiently large number of hidden nodes can approximate any function.
- **Caution**: This result is only in terms of theoretical feasibility. Learning the model can be very difficult in practice (e.g., due to optimization difficulties).
- In Deep Neural Networks, hidden layer can automatically extract features from data. (Hinton 2006, Nature)
- Stochastic gradient descent backpropagation is most popular method for weight optimization.

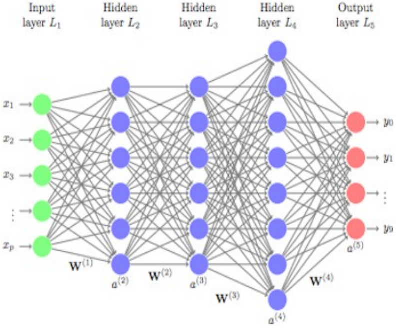


Fig: An example of ANN model with 3 hidden layers

Introduction: Pros & Cons of Neural Nets

- Able to learn any complex nonlinear mapping or approximate any continuous function.
- As a nonparametric model, it doesn't make any prior assumption about the data distribution or input-output mapping function.
- ANN are very flexible with respect to incomplete, missing and noisy data. ANN are "fault tolerant".
- Neural network models can be easily updated / are suitable for dynamic environment.

- Neural network when applied to limited data can overfit the training data and lose generalization capability.
- Training process of ANN is very time-consuming due to having huge number of weights and hyperparameters.
- The selection of the network topology and its parameter lack theoretical background, it is often a "trial and error" matter.
- Advanced ANNs lack theoretical background concerning explanatory capabilities and results in "black-box" model.

Introduction: Statistical Theory to Neural Networks

- Strong Universal Consistency of ANN Classifier (Farago, Lugosi, IEEE IT 1993).
- Approximation properties of ANN (Mhaskar, Advances in Computational Mathematics, 1993).
- Prediction Intervals for Artificial Neural Networks (Hwang, Ding, 1997, JASA)
- Nonasymptotic bounds on the L_2 error of ANN regression estimates (Hamers, Kohler, AISM, 2006).
- Provable approximation properties for DNN (Shaham et al., Applied & Computational Harmonic Analysis, 2018).
- On Deep Learning as a remedy for the curse of dimensionality (Bauer, Kohler, Annals of Statistics, 2019).

Introduction: Consistency Results for Neural Network Classifier

Definition (L_1error)

Define the L_1 error of a function $f : \mathbb{R}^d \to \mathbb{R}$ by

$$J(f) = \mathbb{E}\{|f(X) - Y| \big| Data\}$$

Theorem (Lugosi, Zeger, 1995, IEEE Information Theory)

Consider a neural network with one hidden layer with bounded output weight having k hidden neurons and let σ be a logistic squasher. Let $F_{n,k}$ be the class of neural networks with logistic squasher defined as

$$F_{n,k} = \left\{ \sum_{i=1}^{k} c_i \sigma(a_i^T z + b_i) + c_0 : k \in \mathbb{N}, a_i \in \mathbb{R}^{d_m}, b_i, c_i \in \mathbb{R}, \sum_{i=0}^{k} |c_i| \leq \beta_n \right\}$$

and let ψ_n be the function that minimizes the empirical L_1 error over $\psi_n \in F_{n,k}$. It can be shown that if k and β_n satisfy

$$k \to \infty, \quad \beta_n \to \infty, \quad \frac{k\beta_n^2 log(k\beta_n)}{n} \to 0$$

then the classification rule

$$g_n(z) = \begin{cases} 0, & if\ \psi_n(z) \leq 1/2. \\ 1, & otherwise. \end{cases} \tag{1.5}$$

is universally consistent.

Introduction: Ensemble Models [Murphy Book, 2012]

- Problem: Single classifiers have the drawbacks of sticking to local minimum or over-fitting the data set, etc.
- Ensemble models are such where predictions of multiple models are combined together to build the final model.
- Examples: Bagging, Boosting, Stacking and Voting Method
- Caution: But ensembles dont always improve accuracy of the model but tends to increase the error of each individual base classifier.

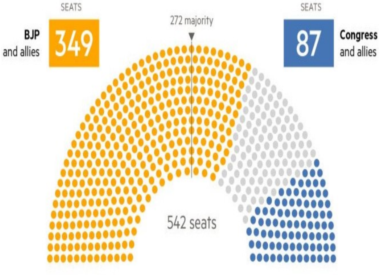


Fig: 2019 Election Result in India which shows the failure of 'Mahajot' in the 'Lok Sabha' Election

Introduction: Hybrid Models [Kuncheva Book, 2013]

- Hybrid models are such where more than one models are combined together.
- It overcomes the limitations of single models and reduce individual variance bias, thus improve the performance of the model.
- Caution: To build a good ensemble classifier the base classifier needs to be simple, as accurate as possible, and distinct from the other classifier used.
- Desired: Interpretability, Less Complexity, Less Tuning Parameters, **high accuracy**.

Fig: "Alone we can do so little; together we can do so much". - Helen Keller

Introduction: Popular Hybrid Prediction Model

- Perceptron Trees (Utgoff, AAAI, 1988).
- Entropy Nets (Sethi, Proceeding of IEEE,1990).
- Neural trees (Sirat, Nadal, Network, 1990).
- Sparse Perceptron Trees (Jackson, Craven, NIPS, 1996).
- SVM Tree Model (Bennett et al., NIPS, 1998)
- Hybrid DT-ANN Model (Jerez-Aragones et al., 2003, AI in Medicine)
- Flexible Neural Tree (Chen et al., Neurocomputing, 2006)
- Hybrid DT-SVM Model (Sugumaran et al,, Mechanical Systems and Signal Processing, 2007).
- Hybrid CNNSVM Classifier (Niu et al., PR, 2012).
- Convolutional Neural Support Vector Machines (Nagi et al., IEEE ICMLA, 2012).
- Hybrid DT model utilizing local SVM (Dejan et al., IJPR, 2013).
- Neural Decision Forests (Bulo, Kontschieder, CVPR, 2014).
- Deep Neural Decision Forests (Kontschieder, ICCV, 2015).
- Soft Decision Tree (Frosst, Hinton, Google AI, 2017).
- Deep Neural Decision Trees (Yang et al., ICML, 2018).

Introduction: Perceptron Trees (AAAI, 1988)

- Perceptron trees are composed of three basic steps:

  (a) Converting a DT into rules.
  (b) Constructing a two hidden layered NN from the rules.
  (c) Training the MLP using gradient descent backpropagation (Rumelhart, Hinton (1988).

- In decision trees, the overfitting occurs when the size of the tree is too large compared to the number of training data.

- Instead of using pruning methods (removing child nodes), PT employs a backpropagation NN to give weights to nodes according to their significance.

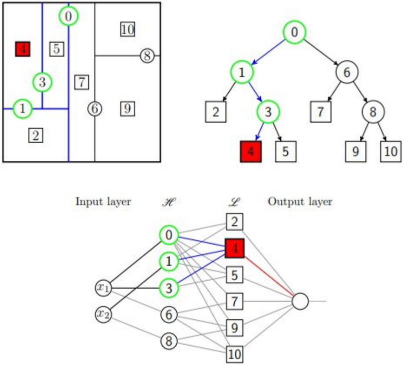


Fig: Graphical Representation of Perceptron Trees Model [Paul Utgoff, 1988, AAAI]

Introduction: SVM Tree Model (NIPS, 1998)

- SVM are generalized to decision trees. SVM is used for each decision in the tree.
- The "optimal" decision tree is characterized, and both a primal and dual space formulation for constructing the tree are introduced.
- The model results in a simple decision trees with multivariate linear or nonlinear decisions.
- Consistency results are yet to be proved and can be extended for different problems (Interesting Problem!).

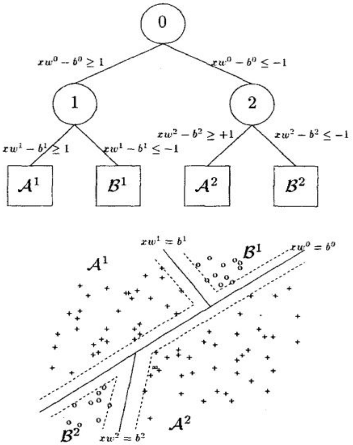


Fig: SVM Formulation for Decision Trees: A logical and geometric depiction of a decision tree with optimal margins [Bennett ET AL., 1998, NIPS]

Introduction: Hybrid DT-ANN Model (AI in Medicine, 2003)

- The DT unit leads to the selection of the most significant prognostic factors from the patients' database for every time interval.
- The NN system computes an attributes set from the prognostic factors selector giving a value corresponding to the a posteriori probability of relapse for the patient under study.
- Useful when (a) data present an important number of attributes with missing values, (b) the prognostic factors' significance is not the same over the time of patient follow-up, and the utilisation of survival estimate techniques is not very advisable.
- Promising area for Biostatisticians.

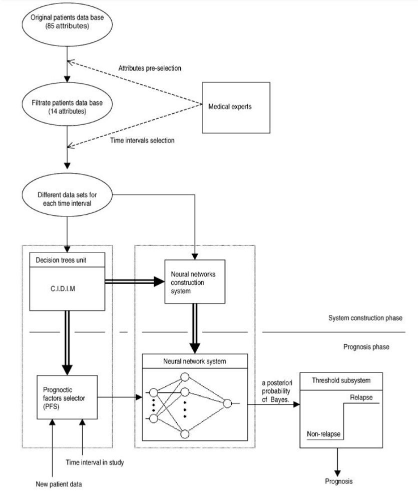

Original patients data base (85 attributes)
Attributes pre-selection
Medical experts
Filtrate patients data base (14 attributes)
Time intervals selection
Different data sets for each time interval
Decision trees unit
C.I.D.I.M
Neural networks construction system
System construction phase
Prognosis phase
Neural network system
Prognoctic factors selector (PFS)
a posteriori probability of Bayes.
Threshold subsystem
Relapse
Non-relapse
Time interval in study
New patient data
Prognosis

Fig: A combined ANN and DT model for prognosis of breast cancer [Jerez-Aragones et al., 2003, AI in Medicine]

Introduction: Hybrid DT-SVM Model (MSSP, 2007)

- DT is used to identify the best features from a given set of samples for the purpose of classification.
- Proximal Support Vector Machine (PSVM) which has the capability to efficiently classify the faults are used for classification task using the DT identified features.
- In general, the approach can be used for feature selection in any domain.
- Simple, interpretable, but lacks accuracy in some typical problems.

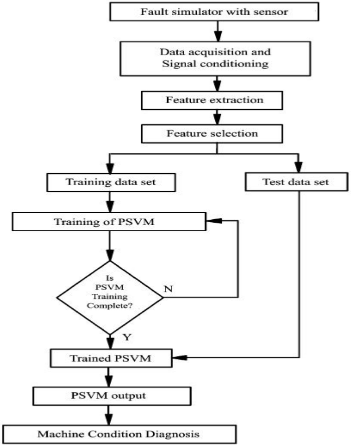

Fault simulator with sensor
Data acquisition and Signal conditioning
Feature extraction
Feature selection
Training data set
Test data set
Training of PSVM
Is PSVM Training Complete?
N
Y
Trained PSVM
PSVM output
Machine Condition Diagnosis

Fig: Feature selection using Decision Tree and classification through Proximal Support Vector Machine for fault diagnostics of roller bearing [Sugumaran et al., 2007, Mechanical Systems & Signal Processing]

Introduction: DNDF for Image Data Sets (ICCV, 2015)

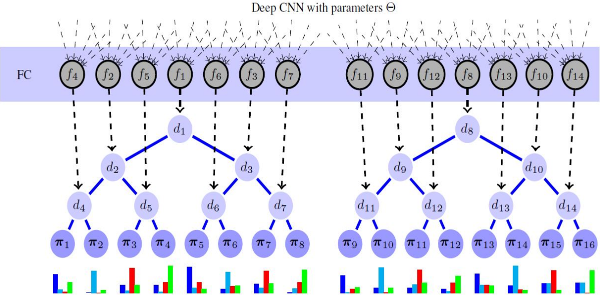

Deep CNN with parameters Θ
FC
f_4 f_2 f_5 f_1 f_6 f_3 f_7 f_{11} f_9 f_{12} f_8 f_{13} f_{10} f_{14}
d_1 d_2 d_3 d_4 d_5 d_6 d_7 d_8 d_9 d_{10} d_{11} d_{12} d_{13} d_{14}
π_1 π_2 π_3 π_4 π_5 π_6 π_7 π_8 π_9 π_{10} π_{11} π_{12} π_{13} π_{14} π_{15} π_{16}

Fig: Deep neural decision forests [Kontschieder et al., 2015, ICCV]

Description: Deep CNN with variable number of layers, subsumed via parameters θ. FC block: Fully Connected layer used to provide functions $f_n(;\theta)$. Each output of f_n is brought in correspondence with a split node in a tree, eventually producing the routing (split) decisions $d_n(x) = \sigma(f_n(x))$. The order of the assignments of output units to decision nodes can be arbitrary (the one we show allows a simple visualization). The circles at bottom correspond to leaf nodes, holding probability distributions π.

Questions?

Thank You