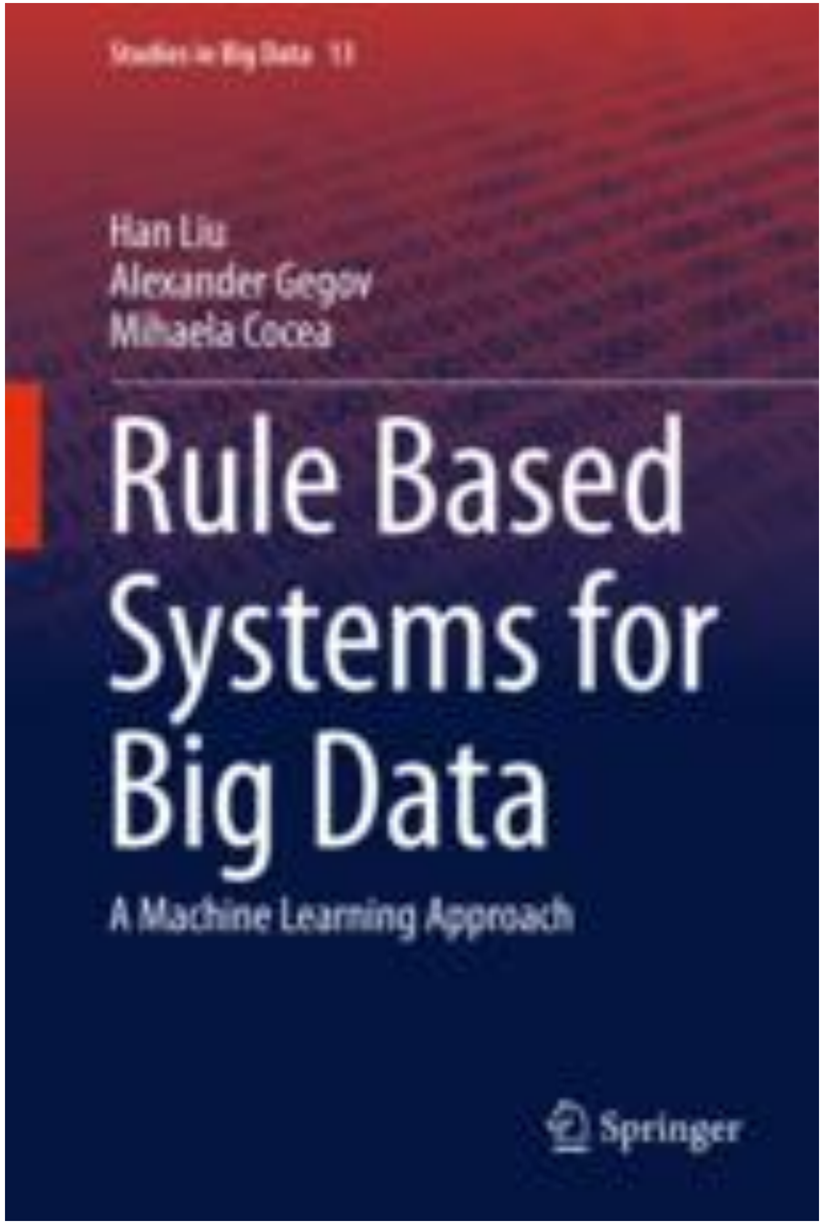
Rule Based Systems and Networks for Knowledge Discovery in Big Data

Alexander Gegov, David Sanders

University of Portsmouth, UK

Contents

1. Introduction
2. Theoretical Preliminaries
3. Rule Generation
4. Rule Simplification
5. Rule Representation
6. Case Studies
7. Conclusion



1. Introduction

- Types

  single set of if-then rules (rule based systems)

  multiple sets of if-then rules (rule based networks)

- Applications
  - ✓Decision support
  - ✓Decision making
  - ✓Correlation analysis
  - ✓Predictive modelling
  - ✓Automatic control

2. Theoretical Preliminaries

2.1 If-then Rules

2.2 Computational Logic

2.3 Machine Learning

2.1 If-Then Rules

- if $x_1 = 0$ and $x_2 = 0$ then $y=0$;
- if $x_1 = 0$ and $x_2 = 1$ then $y=0$;
- if $x_1 = 1$ and $x_2 = 0$ then $y=0$;
- if $x_1 = 1$ and $x_2 = 1$ then $y=1$;

Antecedents: left hand side

Consequents: right hand side

2.2 Computational Logic

- Deterministic rules (based on deterministic logic)

  if x=1 and y=0 then z= 0

- Probabilistic rules (based on probabilistic logic)

  if x=1 and y=0 then z= 0 (70% chance) or z=1(30% chance)

- Fuzzy rules (based on fuzzy logic)

  if x=1 and y=0 then z= 0 (70% truth) or z=1 (30% truth)

2.3 Machine Learning

- Concepts
- Overfitting Problem
- Causes of Prediction Errors

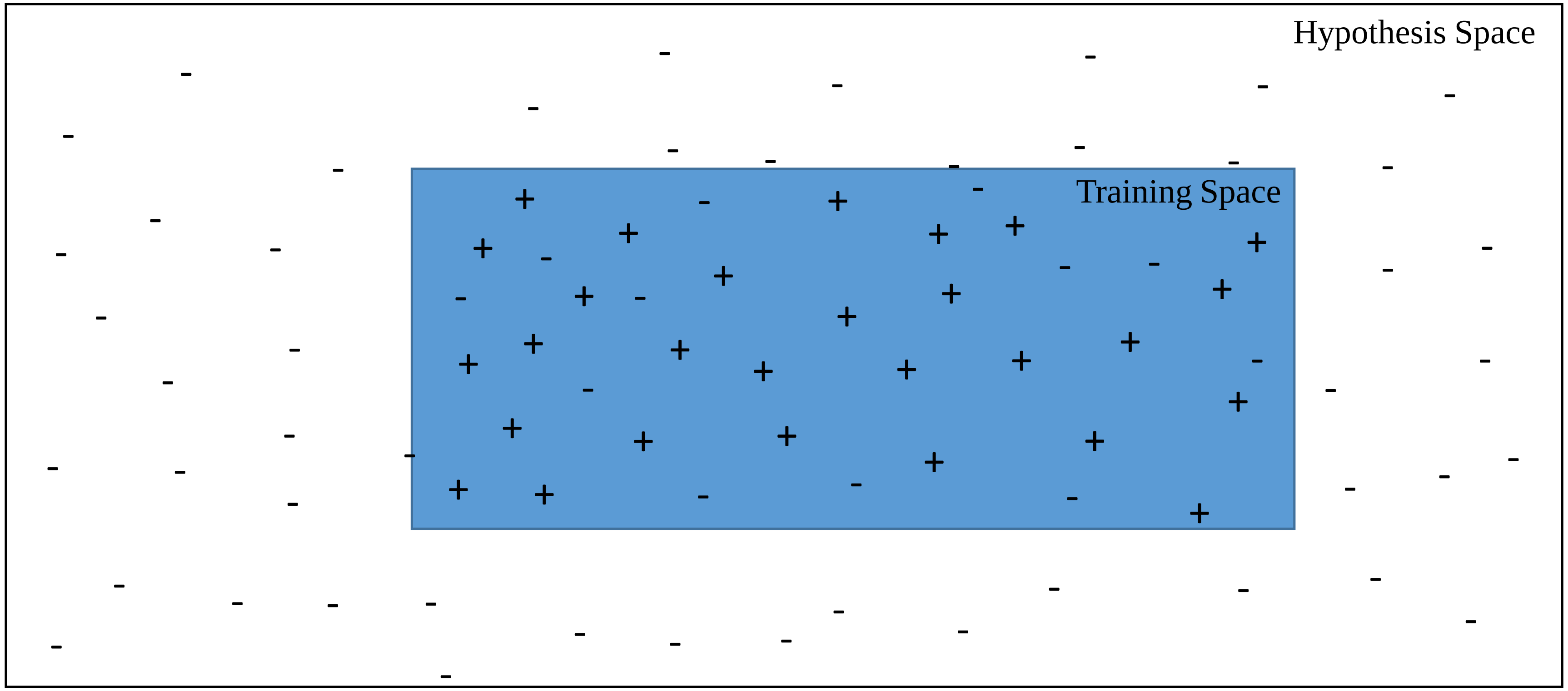
Concepts

- Learning Process
  1. Training: build a model by learning from data
  2. Testing: evaluate the model using different data

- Strategies
  - ✓Learning based on statistical heuristics e.g. ID3, C4.5
  - ✓Learning on a random basis e.g. random decision trees

Overfitting Problem

- Essence: a model performs a high level of accuracy on training data but low level of accuracy on testing data.
- Illustration



NB: “+” indicates training instance and “-” indicates testing instance

Causes of Prediction Errors

- Bias: errors originating from statistical heuristics of algorithms
- Variance: errors originating from random noise in data

3. Rule Generation

- Purpose: to generate rule based models on an inductive basis

- Approaches
  - ✓Divide and conquer: to generate a set of rules recursively in the form of a decision tree, e.g. ID3 and C4.5
  - ✓Separate and conquer: to generate a set of if-then rules sequentially, e.g. Prism

Example for Divide and Conquer

Eye colour	Married	Sex	Hair length	Class
brown	yes	male	long	football
blue	yes	male	short	football
brown	yes	male	long	football
brown	no	female	long	netball
brown	no	female	long	netball
blue	no	male	long	football
brown	no	female	long	netball
brown	no	male	short	football
brown	yes	female	short	netball
brown	no	female	long	netball
blue	no	male	long	football
blue	no	male	short	football

Fig.1 Training Set for Football/Netball Example

Sport Example

Eye colour	Married	Sex	Hair length	Class
brown	yes	male	long	football
blue	yes	male	short	football
brown	yes	male	long	football
blue	no	male	long	football
brown	no	male	short	football
blue	no	male	long	football
blue	no	male	short	football

Eye colour	Married	Sex	Hair length	Class
brown	no	female	long	netball
brown	no	female	long	netball
brown	no	female	long	netball
brown	yes	female	short	netball
brown	no	female	long	netball

Rule Set Generated

- Rule 1: If Sex= male Then Class= football;
- Rule 2: If Sex= female Then Class= netball;

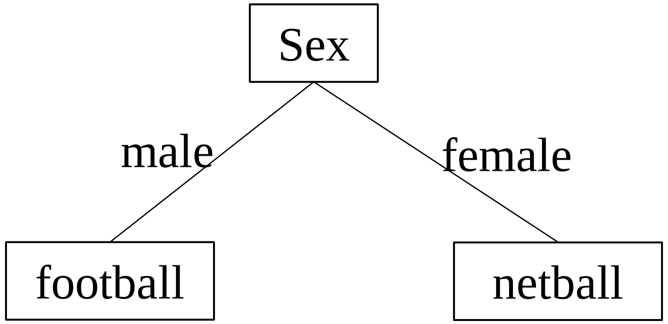


Fig.2 Tree Representation

Example for Separate and Conquer

Outlook	Temp (◦F)	Humidity(%)	Windy	Class
sunny	75	70	true	play
sunny	80	90	true	don’t play
sunny	85	85	false	don’t play
sunny	72	95	false	don’t play
sunny	69	70	false	play
overcast	72	90	true	play
overcast	83	78	false	play
overcast	64	65	true	play
overcast	81	75	false	play
rain	71	80	true	don’t play
rain	65	70	true	don’t play
rain	75	80	false	play
rain	68	80	false	play
rain	70	96	false	play

Fig.3 Weather Data set

Weather Example

Outlook	Temp (◦F)	Humidity(%)	Windy	Class
overcast	72	90	true	play
overcast	83	78	false	play
overcast	64	65	true	play
overcast	81	75	false	play

Fig.4 subset comprising ‘Outlook= overcast’

The first rule generation is complete.
The rule is: If Outlook= overcast Then Class= play;
All instances covered by this rule are deleted from training set.

Weather Example

Outlook	Temp (◦F)	Humidity(%)	Windy	Class
sunny	75	70	true	play
sunny	80	90	true	don’t play
sunny	85	85	false	don’t play
sunny	72	95	false	don’t play
sunny	69	70	false	play
rain	71	80	true	don’t play
rain	65	70	true	don’t play
rain	75	80	false	play
rain	68	80	false	play
rain	70	96	false	play

Fig.5 reduced training set after deleting instances comprising ‘outlook= overcast’

Weather Example

Outlook	Temp (◦F)	Humidity(%)	Windy	Class
rain	71	80	true	don’t play
rain	65	70	true	don’t play
rain	75	80	false	play
rain	68	80	false	play
rain	70	96	false	play

Fig.6 The subset comprising ‘outlook= rain’

Outlook	Temp (◦F)	Humidity(%)	Windy	Class
rain	75	80	false	play
rain	68	80	false	play
rain	70	96	false	play

Fig.7 The subset comprising ‘Windy= false’

The second rule generated is:
If Outlook= rain And Windy= false Then Class= play

4. Rule Simplification

- Purpose: to simplify rules and reduce the complexity of the rule set

- Approaches
  - ✓Pre-pruning: to simplify rules when they are being generated
  - ✓Post-pruning: to simplify rules after they have been generated

Pruning of Decision Trees

- Pre-pruning: to stop a branch growing further

- Post-pruning:
- first, to normally generate a whole tree
- then, to convert the tree into a set of if-then rules
- finally, to simplify each of the rules

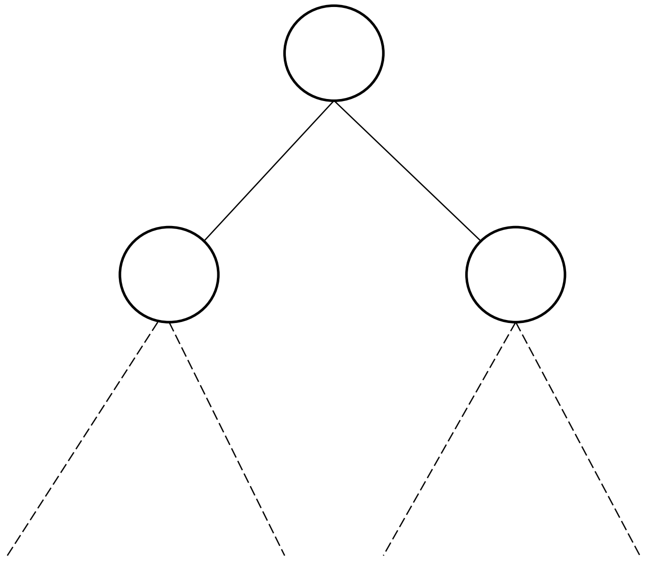


Fig.8 Incomplete Decision Tree

Pruning of If-Then Rules

- Pre-pruning: to prevent a rule being from being too specialised on its left hand side

- Post-pruning:
- first, to normally generate a rule
- then, to simplify the rule by removing some of its rule terms from its left hand side

- Original rule

if a=1 and b=1 and c=1 and d=1 then class=1;

- Simplified rule

if a=1 and b=1 then class= 1;

5. Rule Representation

- Purpose
  - ✓to manage the computational efficiency in predicting unseen instances
  - ✓To manage the interpretability of a rule based model for knowledge discovery

- Techniques
  - ✓ decision tree
  - ✓ linear list
  - ✓ rule based network

Rule Representation Techniques

Treed Rules

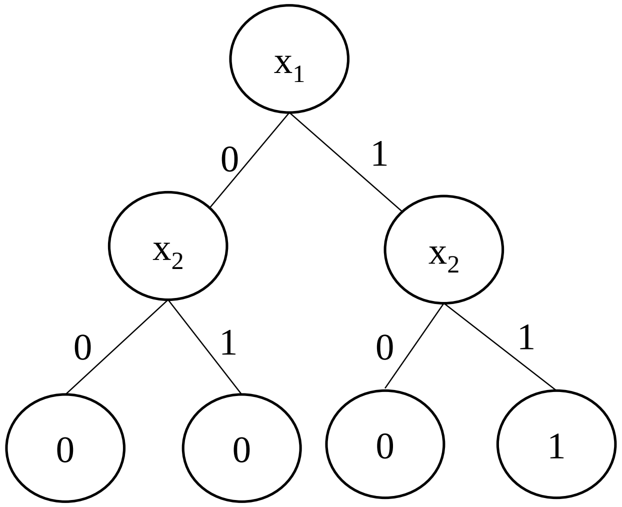


Fig.9 Decision Tree

Listed Rules

if $x_1 = 0$ and $x_2 = 0$ then $y=0$;
if $x_1 = 0$ and $x_2 = 1$ then $y=0$;
if $x_1 = 1$ and $x_2 = 0$ then $y=0$;
if $x_1 = 1$ and $x_2 = 1$ then $y=1$;

Networked Rules

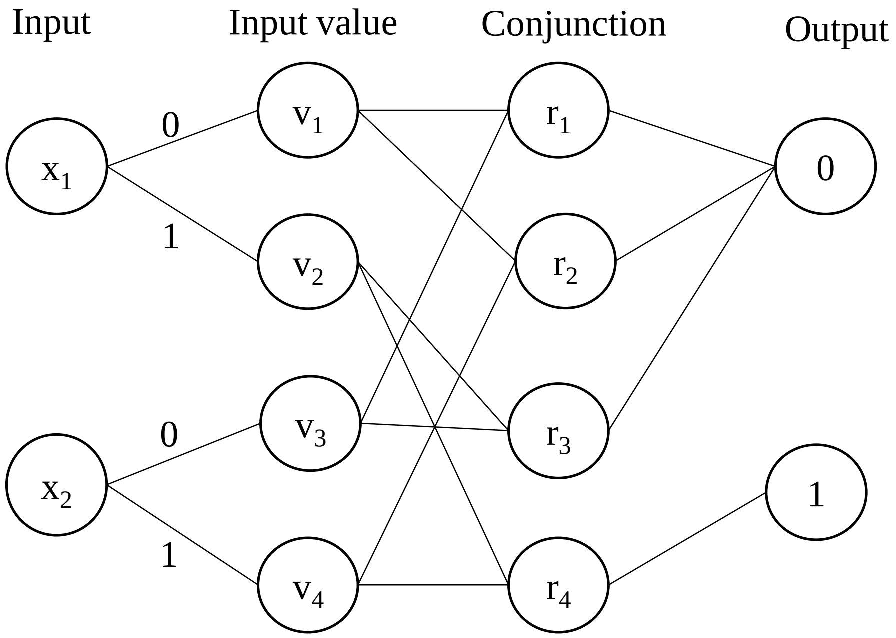


Fig.10 Rule Based Network

Comparison in Efficiency

Decision Tree	Linear List	Rule Based Network
O(log(n))	O(n)	O(log(n

Note: n is the total number of rule terms in a rule set.

Comparison in Interpretability

Criteria	Decision Tree	Linear List	Rule Based Network
correlation between attributes and classes	Poor	Implicit	Explicit
relationship between attributes and rules	Implicit	Implicit	Explicit
ranking of attributes	Poor	Poor	Explicit
ranking of rules	Poor	Explicit	Explicit
attribute relevance	Poor	Poor	Explicit
overall	Low	Medium	High

6. Case Studies

- Overview of big data
- Impact on machine learning
- Findings through cases studies

Overview of Big Data

Four Vs defined by IBM:

- **Volume** - terabytes, petabytes, or more
- **Velocity** - data in motion or streaming data
- **Variety** - structured and unstructured data of all types - text, sensor data, audio, video, click streams, log files and more
- **Veracity** - the degree to which data can be trusted

Impact on Machine Learning

- Advantages
  - ✓Advances in data coverage
  - ✓Advances in overfitting reduction

- Disadvantages
  - ✓Increase of noise in data
  - ✓Increase of computational costs

Findings Through Case Studies

- Case Study I- Rule Generation
  - ✓Individual algorithms generally have their own inductive bias
  - ✓Different algorithms could be complementary to each other
- Case Study II- Rule Simplification
  - ✓Pruning algorithms reduce model overfitting
  - ✓Pruning algorithms reduce model complexity
- Case Study III- Ensemble Learning
  - ✓Bagging reduces variance on data side
  - ✓Collaborative rule learning reduces bias on algorithms side
  - ✓Heuristics based model weighting still causes bias
  - ✓Randomness in data sampling still causes variance

7. Conclusion

- Theoretical Significance
- Practical Importance
- Methodological Impact
- Philosophical Aspects
- Further Directions

Theoretical Significance

- Development of a unified framework for building rule based systems
- Development of novel approaches for rule generation, simplification and representation
- Novel applications of graph theory and BigO notation

Practical Importance

- Knowledge discovery and predictive modelling
- Parallel, distributed and mobile data modelling
- Domain independent in real applications

Methodological Impact

- Complement existing rule learning methods
- Collaboration with existing rule learning methods
- Advances in interpretability of rule based models

Philosophical Aspects

- Novel understanding of data mining and machine learning
- Philosophical inspiration by information theory, system theory and control theory

Further Directions

- Adopt probabilistic or fuzzy logic for uncertainty handling
- Adopt naturally and biologically inspired methods
- Adopt clustering techniques for splitting data into training/testing sets
- Improve representativeness and completeness of data