

Data Science Toolbox

Mike Micatka

April 13, 2016

Contents

1	Data Techniques	1
2	Machine Learning Techniques	2
2.1	Supervised Learning	2
2.1.1	Averaged One-Dependence Estimators (AODE)	2
2.1.2	Bayesian Statistics	3
2.1.3	Case-Based Reasoning	3
2.1.4	Gaussian Process Regression	3
2.1.5	Gene Expression Programming	3
2.1.6	Group Method of Data Handling (GMDH)	3
2.1.7	Inductive Logic Programming	3
2.1.8	Instance-based Learning	3
2.1.9	Lazy Learning	3
2.1.10	Learning Vector Quantization	3
2.2	Unsupervised Learning	3
2.3	Semi-Supervised Learning	3
2.4	Reinforcement Learning	3
2.5	Deep Learning	3
3	Choosing the Right Algorithm	4

What is Data Science?

This is a work in progress. Mainly made for learning purposes. Hopelessly abbreviated, will do my best to provide sources/other resources.

Chapter 1

Data Techniques

Chapter 2

Machine Learning Techniques

2.1 Supervised Learning

2.1.1 Averaged One-Dependence Estimators (AODE)

Averaged One-Dependence Estimators is a probabilistic classification technique. It is an improvement on the Naive Bayes Estimator. This technique produces class probabilities rather than single classes which allows more flexibility by having the end-user set the threshold for selection. The computational complexity for training is $O(ln^2)$ and $O(kn^2)$ for classification where n is the number of features, l is the number of training samples, and k is the number of testing samples. The equation for this classifier is as follows:

$$\hat{P}(y|x_1, \dots, x_n) = \frac{\sum_{i: 1 \leq i \leq n \wedge F(x_i) \geq m} \hat{P}(y, x_i) \prod_{j=1}^n \hat{P}(x_j|y, x_i)}{\sum_{y' \in Y} \sum_{i: 1 \leq i \leq n \wedge F(x_i) \geq m} \hat{P}(y', x_i) \prod_{j=1}^n \hat{P}(x_j|y', x_i)}$$

Where $\hat{P}(\cdot)$ is the estimate of $P(\cdot)$, $F(\cdot)$ is the frequency of the argument in the sample data, and m is the user specified minimum frequency, usually set at 1. The computational complexity makes this technique infeasible for high-dimensional data but is linear with respect to the number of samples so can handle large amounts of training data.

Implementations:

- Weka

- 2.1.2 Bayesian Statistics
- 2.1.3 Case-Based Reasoning
- 2.1.4 Gaussian Process Regression
- 2.1.5 Gene Expression Programming
- 2.1.6 Group Method of Data Handling (GMDH)
- 2.1.7 Inductive Logic Programming
- 2.1.8 Instance-based Learning
- 2.1.9 Lazy Learning
- 2.1.10 Learning Vector Quantization
- 2.1.11 Logistic Model Tree
- 2.1.12 Minimum Message Length
- 2.1.13 Probably Approximately Correct Learning (PAC)
- 2.1.14 Ripple Down Rules
- 2.1.15 Support Vector Machine (SVM)

Add in stuff about different kernels

- 2.1.16 Random Forests
- 2.1.17 Ensemble Learning
- 2.1.18 Ordinal Classification
- 2.1.19 Information Fuzzy Network (IFN)
- 2.1.20 Conditional Random Field
- 2.1.21 ANOVA
- 2.1.22 Linear Classifier
- 2.1.23 Quadratic Classifier
- 2.1.24 Nearest Neighbor
- 2.1.25 Boosting
- 2.1.26 Decision Tree

This will have several parts

2.1.27 Bayesian Network

2.1.28 Hidden Markov Model

2.2 Unsupervised Learning

2.2.1 Expectation-Maximization Algorithm

2.2.2 Vector Quantization

2.2.3 Generative Topographic Map

2.2.4 Information Bottleneck Method

2.2.5 Association Rule Learning

- Apriori Algorithm
- Eclat Algorithm
- FP-Growth Algorithm

2.2.6 Hierarchical Clustering

- Single-Linkage Clustering
- Conceptual Clustering

2.2.7 Cluster Analysis

- K-Means Algorithm
- Fuzzy Clustering
- DBSCAN
- OPTICS Algorithm

2.2.8 Outlier Detection

2.3 Semi-Supervised Learning

2.3.1 Generative Model

2.3.2 Low-Density Separation

2.3.3 Graph-Based Methods

2.3.4 Co-Training

2.4 Reinforcement Learning

2.4.1 Temporal Difference Learning

2.4.2 Q-Learning

2.4.3 Learning Automata

2.4.4 State-Action-Reward-State-Action (SARSA)

2.5 Deep Learning

2.5.1 Deep Belief Network

2.5.2 Deep Boltzman Machine

2.5.3 Deep Convolution Neural Networks (CNN)

2.5.4 Deep Recurrent Neural Networks (RNN)

2.5.5 Hierarchical Temporal Memory

Chapter 3

Choosing the Right Algorithm