Autonomous Learning Systems
From Data Streams to Knowledge in Real-time

Plamen Angelov
Autonomous Learning Systems
Autonomous Learning Systems
From Data Streams to Knowledge in Real-time

Plamen Angelov
Lancaster University, UK
Contents

Forewords xi
Preface xix
About the Author xxiii

1 Introduction 1
   1.1 Autonomous Systems 3
   1.2 The Role of Machine Learning in Autonomous Systems 4
   1.3 System Identification – an Abstract Model of the Real World 6
      1.3.1 System Structure Identification 6
      1.3.2 Parameter Identification 8
      1.3.3 Novelty Detection, Outliers and the Link to Structure Innovation 9
   1.4 Online versus Offline Identification 9
   1.5 Adaptive and Evolving Systems 10
   1.6 Evolving or Evolutionary Systems 11
   1.7 Supervised versus Unsupervised Learning 13
   1.8 Structure of the Book 14

PART I FUNDAMENTALS

2 Fundamentals of Probability Theory 19
   2.1 Randomness and Determinism 20
   2.2 Frequentistic versus Belief-Based Approach 22
   2.3 Probability Densities and Moments 23
   2.4 Density Estimation – Kernel-Based Approach 26
   2.5 Recursive Density Estimation (RDE) 28
   2.6 Detecting Novelties/Anomalies/Outliers using RDE 32
   2.7 Conclusions 36
# Fundamentals of Machine Learning and Pattern Recognition

## 3.1 Preprocessing
- **3.1.1 Normalisation and Standardisation**
- **3.1.2 Orthogonalisation of Inputs/Features – rPCA Method**
  - **3.1.2.1 The Basics of the PCA Method**
  - **3.1.2.2 Offline PCA**
  - **3.1.2.3 Online (Recursive) Version of PCA, rPCA**

## 3.2 Clustering
- **3.2.1 Proximity Measures and Clusters Shape**
- **3.2.2 Offline Methods**
  - **3.2.2.1 A Brief Introduction to the Mountain Clustering Method**
  - **3.2.2.2 Subtractive Clustering Method Outline**
  - **3.2.2.3 Gustafson–Kessel Clustering Algorithm**
  - **3.2.2.4 Mean Shift Clustering Algorithm**
- **3.2.3 Evolving Clustering Methods**
  - **3.2.3.1 Incremental VQ Clustering Method**
  - **3.2.3.2 Evolving Clustering Algorithm eClustering**
  - **3.2.3.3 Evolving Local Means (ELM) Clustering Algorithm**
  - **3.2.3.4 Evolving GK-like Algorithm**

## 3.3 Classification
- **3.3.1 Recursive LDA, rLDA**

## 3.4 Conclusions

# Fundamentals of Fuzzy Systems Theory

## 4.1 Fuzzy Sets

## 4.2 Fuzzy Systems, Fuzzy Rules
- **4.2.1 Fuzzy Systems of Zadeh–Mamdani Type**
  - **4.2.1.1 Linguistic Terms and Variables**
  - **4.2.1.2 Inference and Defuzzification**
- **4.2.2 Takagi–Sugeno Fuzzy Systems**
  - **4.2.2.1 Architecture of Takagi–Sugeno Fuzzy Systems**
  - **4.2.2.2 Multi-Input–Multi-Output (MIMO) Takagi–Sugeno Fuzzy Systems**
  - **4.2.2.3 Analysis of the Inference in Takagi–Sugeno Fuzzy Systems**

## 4.3 Fuzzy Systems with Nonparametric Antecedents (AnYa)
- **4.3.1 Architecture**
- **4.3.2 Analysis of AnYa**

## 4.4 FRB (Offline) Classifiers

## 4.5 Neurofuzzy Systems
- **4.5.1 Neurofuzzy System Architecture**
  - **4.5.1.1 TS Type NFS**
  - **4.5.1.2 AnYa Type NFS**
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5.2 Evolving NFS</td>
<td>78</td>
</tr>
<tr>
<td>4.5.3 Linguistic Interpretation of the NFS</td>
<td>79</td>
</tr>
<tr>
<td>4.6 State Space Perspective</td>
<td>79</td>
</tr>
<tr>
<td>4.7 Conclusions</td>
<td>81</td>
</tr>
</tbody>
</table>

**PART II METHODOLOGY OF AUTONOMOUS LEARNING SYSTEMS**

5 Evolving System Structure from Streaming Data 85
---
5.1 Defining System Structure Based on *Prior* Knowledge 85
5.2 Data Space Partitioning 86
---
5.2.1 Regular Partitioning of the Data Space 87
5.2.2 Data Space Partitioning through Clustering 87
5.2.3 Data Space Partitioning Based on Data Clouds 88
5.2.4 Importance of Partitioning the Joint Input–Output Data Space 89
5.2.5 Principles of Data Space Partitioning for Autonomous Machine Learning 91
5.2.6 Dynamic Data Space Partitioning – Evolving System Structure Autonomously – Example of Fish Classification 91
---
5.2.6.1 Data Space Partitioning Based on Prior Knowledge 92
5.2.6.2 Regular Data Space Partitioning 95
5.2.6.3 Data Space Partitioning through Clustering 95
5.2.6.4 Data Space Partitioning through Clouds 95
5.3 Normalisation and Standardisation of Streaming Data in an Evolving Environment 96
---
5.3.1 Standardisation in an Evolving Environment 97
5.3.2 Normalisation in an Evolving Environment 98
5.4 Autonomous Monitoring of the Structure Quality 98
---
5.4.1 Autonomous Input Variables Selection 98
5.4.2 Autonomous Monitoring of the Age of the Local Submodels 101
5.4.3 Autonomous Monitoring of the Utility of the Local Submodels 102
5.4.4 Update of the Cluster Radii 103
5.5 Short- and Long-Term Focal Points and Submodels 104
5.6 Simplification and Interpretability Issues 105
5.7 Conclusions 107

6 Autonomous Learning Parameters of the Local Submodels 109
---
6.1 Learning Parameters of Local Submodels 110
6.2 Global versus Local Learning 111
6.3 Evolving Systems Structure Recursively 113
6.4 Learning Modes 116
6.5 Robustness to Outliers in Autonomous Learning 118
6.6 Conclusions 118
7 Autonomous Predictors, Estimators, Filters, Inferential Sensors 121
  7.1 Predictors, Estimators, Filters – Problem Formulation 121
  7.2 Nonlinear Regression 123
  7.3 Time Series 124
  7.4 Autonomous Learning Sensors 125
    7.4.1 Autonomous Sensors – Problem Definition 125
    7.4.2 A Brief Overview of Soft/Intelligent/Inferential Sensors 126
    7.4.3 Autonomous Intelligent Sensors (AutoSense) 127
    7.4.4 AutoSense Architecture 128
    7.4.5 Modes of Operation of AutoSense 129
    7.4.6 Autonomous Input Variable Selection 130
  7.5 Conclusions 131

8 Autonomous Learning Classifiers 133
  8.1 Classifying Data Streams 133
  8.2 Why Adapt the Classifier Structure? 134
  8.3 Architecture of Autonomous Classifiers of the Family AutoClassify 135
    8.3.1 AutoClassify0 136
    8.3.2 AutoClassify1 137
      8.3.2.1 Multiple (m) Two-Class Classification Problems 138
      8.3.2.2 AutoClass1 MIMO 138
    8.4 Learning AutoClassify from Streaming Data 139
      8.4.1 Learning AutoClassify0 139
      8.4.2 Learning AutoClassify1 139
    8.5 Analysis of AutoClassify 140
    8.6 Conclusions 140

9 Autonomous Learning Controllers 143
  9.1 Indirect Adaptive Control Scheme 144
  9.2 Evolving Inverse Plant Model from Online Streaming Data 145
  9.3 Evolving Fuzzy Controller Structure from Online Streaming Data 147
  9.4 Examples of Using AutoControl 148
  9.5 Conclusions 153

10 Collaborative Autonomous Learning Systems 155
  10.1 Distributed Intelligence Scenarios 155
  10.2 Autonomous Collaborative Learning 157
  10.3 Collaborative Autonomous Clustering, AutoCluster by a Team of ALSs 158
  10.4 Collaborative Autonomous Predictors, Estimators, Filters and AutoSense by a Team of ALSs 159
  10.5 Collaborative Autonomous Classifiers AutoClassify by a Team of ALSs 160
  10.6 Superposition of Local Submodels 161
  10.7 Conclusions 161
PART III  APPLICATIONS OF ALS

11  Autonomous Learning Sensors for Chemical and Petrochemical Industries

11.1 Case Study 1: Quality of the Products in an Oil Refinery
11.1.1 Introduction
11.1.2 The Current State-of-the-Art
11.1.3 Problem Description
11.1.4 The Dataset
11.1.5 AutoSense for Kerosene Quality Prediction
11.1.6 AutoSense for Abel Inflammability Test
11.2 Case Study 2: Polypropylene Manufacturing
11.2.1 Problem Description
11.2.2 Drift and Shift Detection by Cluster Age Derivatives
11.2.3 Input Variables Selection
11.3 Conclusions

12  Autonomous Learning Systems in Mobile Robotics

12.1 The Mobile Robot Pioneer 3DX
12.2 Autonomous Classifier for Landmark Recognition
12.2.1 Corner Detection and Simple Mapping of an Indoor Environment through Wall Following
12.2.2 Outdoor Landmark Detection Based on Visual Input Information
12.2.3 VideoDiaries
12.2.4 Collaborative Scenario
12.3 Autonomous Leader Follower
12.4 Results Analysis

13  Autonomous Novelty Detection and Object Tracking in Video Streams

13.1 Problem Definition
13.2 Background Subtraction and KDE for Detecting Visual Novelties
13.2.1 Background Subtraction Method
13.2.2 Challenges
13.2.2.1 Illumination Changes
13.2.2.2 Shadows and Reflection
13.2.2.3 Occlusions and Camouflage
13.2.2.4 Nonstatic Background and Camera Oscillations
13.2.3 Parametric versus Nonparametric Approaches
13.2.4 Kernel Density Estimation Method
13.3 Detecting Visual Novelties with the RDE Method
13.4 Object Identification in Image Frames Using RDE
13.5 Real-time Tracking in Video Streams Using ALS
13.6 Conclusions
Efficient and robust performance in imperfectly known, nonstationary, environments – and this characterizes the vast majority of real-world applications – requires systems that can improve themselves, transcending their initial design, continuously optimizing their parameters, models, and methods. These improvements come predominantly from learning – about the environment, about the ageing self, about the interactions with, and within, the environment, and from the ability to put this learning to use. Batch learning – or at least repeated updating learning from most recent batches, is sufficient only for a limited number of applications. For other applications learning needs to be incremental, to sample level, a learn-or-perish, or at least learn-or-pay (a hefty price) situation. In particular, real-time learning is most critical for bots, virtual or real, agents of the cyberphysical systems that need the agility to swiftly react to virus attacks, or physical robots exposed to hazards while performing search and rescue in disaster areas, or dealing with what is, for now, a largely unpredictable partner: the human. The fast advancement in autonomous systems makes the subject of real-time autonomous learning critically important, and yet the literature addressing this topic is extremely scarce.

Dr Angelov’s pioneering book addresses this problem at its core, focusing on real-time, online learning from streaming data on a sample-by-sample basis. It offers a basic framework for understanding and for designing such systems. It importantly contributes to a more powerful learning, not only of the parameters but of a better structure as well. Conventional approaches are characterized by the fact that the system structure (model) is determined in the beginning, by human designers of the system, and only parameters are learned from the interaction. The entire model identification–learning process can, however, be posed as an optimization problem, as the author points out, and this includes automatically determining the optimal structure in conjunction with the optimal parameters for it. This is done automatically in the methods described in the book, and constitutes a significant and valuable contribution. The system is continuously evolving, not in the evolutionary (genetic) sense of improvement over generations, but continuously perfecting its development.
A valuable contribution of the book is that it offers a high-level, holistic perspective of the field, which helps both students and expert practitioners better comprehend the interplay of various disciplines involved in learning autonomous systems, as diverse as adaptive control and evolutionary algorithms, offering analogies between different disciplines and referring to the equivalency of the concepts characterized by different terminology in different disciplines. It is not meant to be a comprehensive reference of concepts and methods in the field, the author instead paints the landscape with selected brush strokes that allows the viewer to see the forest without getting lost in seeing the trees. It is a work that charters a new field, innovates in methods to advance into it, and outlines new challenges to be addressed by future explorers.

The selected concepts and methods, a good number of which come from the author’s own prior work, are used in the second part of the book to illustrate the implementation of learning in autonomous systems. Concepts such as that of evolving clusters, ‘age’ of an (evolving) local submodel, and methods such as recursive density estimation (RDE) introduced by the author, showing significant improvement over the state-of-the art, are important additions to the arsenal of tools for real-time learning. In particular, I believe that adaptive, self-learning (evolving) classifiers will play a fundamental role in future autonomous learning systems.

The book’s last part is a review of three applications: autonomous learning sensors for chemical and petrochemical applications, autonomous learning in mobile robots, and autonomous novelty detection and object tracking in video systems. These diverse domains illustrate the general applicability of the set of methodology presented in the book and focused on the main theme of this work: the real-time autonomous online learning from data streams.

The field of autonomous learning systems is destined to play an increasingly important role in most systems that will surround us in cyberphysical space. Converting information in data streams, larger and larger, to actionable knowledge, in real time: this is the great challenge ahead, and this book is an important step towards addressing it.

Adrian Stoica
Jet Propulsion Laboratory
California Institute of Technology
June 2012

Vladik Kreinovich

In many practical situations, we have experts who are skilled in doing certain tasks: expert medical doctors are skilled in diagnosing and curing diseases, professional drivers are skilled in driving cars – in particular, driving them in difficult traffic and/or weather conditions, etc. It is desirable to incorporate the knowledge of these top experts in an automatic system that would help other users perform the corresponding tasks – and, ideally, perform these tasks automatically.
Experts are usually willing to share their knowledge, but the difficulty is that in many situations, experts describe their knowledge by using imprecise (“fuzzy”) words from natural language, like “small”. For example, an expert driver rarely describes his or her experience in precise terms like “if the car in front slows down by 10 km/h and it is at a distance of 10 meters, you should press the brake for 0.6 seconds with a force of 2.7 Newtons”; most probably, the rule described by an expert driver is “if the car in front of you is close, and it suddenly slows down some, then you should brake right away”. In this rule, “close” and “some” are imprecise terms: while everyone would agree that, say 100 meters is not close while 5 meters is close, there will not be a precise threshold so that before this threshold the distance is close, and a 1 cm larger distance is not close.

To describe such imprecise (fuzzy) knowledge in computer-understandable precise terms, Professor Lotfi A. Zadeh invented, in the 1960s, a new approach called fuzzy logic. Zadeh’s ideas led to revolutionary changes in many control situations: from the first successful control applications in the 1970s through the fuzzy control boom in the 1990s – when fuzzy-controlled washing machines, camcorders, elevators, trains were heavily promoted and advertised – to the current ubiquity of fuzzy controllers. Just like nowadays computers are ubiquitous – companies no longer brag about computer chips in their cars, since all the cars have such chips – similarly, fuzzy control is ubiquitous: for example, in many cars, automatic transmission systems use fuzzy control.

The existing fuzzy controllers are very successful, but they have a serious limitation: they do not learn. Once the original expert rules are implemented, these same rules are used again and again, even when it becomes clear that the rules need to be updated. We still need an expert to update these rules.

There are, of course, numerous intelligent systems that can learn, such as artificial neural networks, but from the viewpoint of the user, these systems are “black boxes”: we may trust them, but we cannot easily understand the recommendations. In contrast, fuzzy rules, by definition, are formulated in terms of understandable rules. If we could make fuzzy systems themselves learn, make them automatically update the rules – this would combine the clarity of fuzzy rules with the autonomous learning ability of neural networks. This would make learning fuzzy controllers even more efficient – and therefore, even more widely used. This would lead to a second revolution in intelligent control.

And this revolution is starting. This book, by Dr. Plamen Angelov, one of the world’s leading specialists in learning fuzzy systems, is the first book that summarizes the current techniques and successes of autonomously learning fuzzy (and other) systems – techniques mostly developed by Dr. Angelov himself, often in collaboration with other renowned fuzzy researchers (like Dr. Ronald Yager). Some of these techniques have previously appeared in technical journals and proceedings of international conferences, some appear here for the first time.

Ideas are many, it is difficult to describe them all in a short preface, so let us just give a few examples. The first example is an interesting AnYa algorithm invented by Angelov and Yager (Anya is also a Russian short form of Anna (Anne)). In fuzzy logic,