Multivariate Data Modeling and Its Applications to Conditional Outlier Detection

Charmgil Hong
Department of Computer Science

August 9, 2017

Advisor
Milos Hauskrecht, PhD, Department of Computer Science

Dissertation Committee
Rebecca Hwa, PhD, Department of Computer Science
Adriana Kovashka, PhD, Department of Computer Science
Gregory Cooper, MD, PhD, Department of Biomedical Informatics

Prepared in partial fulfillment of the requirements for the degree of Doctor of Philosophy
Introduction

- As the data technology advances, the *data we collect and store becomes more complex*
  - Various kinds of data are being collected from heterogeneous sources
  - Univariate time series data are replaced with multivariate time series
  - Low-dimensional data objects are becoming high-dimensional
  - Input-output data pairs for classification include multiple class labels
- These create new challenges in data analytic and machine learning

Introduction

- Focus on *input-output data objects with high-dimensional multivariate binary output space*

Examples

- Semantic image/video analysis
- Document topic classification

Examples:

- Car
- Road
- Building

- Politics
- Economics
Introduction

• Focus on **input-output data objects with high-dimensional multivariate binary output space**

Examples

Gene functional annotation

- cell growth
- signal transduction
- cellular organization

Medication prescription

- metronidazole
- ciprofloxacin
- oxycodone
Introduction

- Develop new data analytic and machine learning solutions for two problems

<table>
<thead>
<tr>
<th>Problem 1</th>
<th>Problem 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Multi-Label Classification</strong></td>
<td><strong>Conditional Outlier Detection</strong></td>
</tr>
<tr>
<td>How to accurately and efficiently learn and predict the best output (response) from complex input-output data?</td>
<td>How to effectively identify unusual output patterns in input-output data?</td>
</tr>
</tbody>
</table>

**Approaches to Problem 1**
- Utilize decomposed data representations and their ensembles

**Approaches to Problem 2**
- Directly utilize decomposed probabilistic models or Borrow the decomposition idea for a non-probabilistic evaluation schema

Key theme: Decomposition
Part 1

Multi-Label Classification (MLC)
Part I - Multi-Label Classification

• Agenda
  • Motivation
  • Problem definition
  • Existing solutions
  • Our approaches
  • Experimental Evaluation
Motivation

• Traditional classification tasks
  • Each data instance is associated with a *single* class variable
Motivation

• In many real-world applications, data can include *multiple class variables*

Document topic classification

Image classification

Gene functional annotation

Medication prescription
Problem Definition

- **Multi-Label Classification (MLC)**
  - Each data instance is associated with *multiple binary* class labels (outputs)
  - **Goal:** To assign each instance the *most probable assignment* of the class variables

\[
h : \mathbf{X} \in \mathbb{R}^m \rightarrow \mathbf{Y} \in \{0, 1\}^d
\]

Class 1 $\in \{\text{Red, Blue}\}$
Class 2 $\in \{\circ, \triangle\}$
Problem Definition

• Multi-Label Classification (MLC)
  
  • Each data instance is associated with *multiple binary* class labels (outputs)
  
  • *Goal*: To assign each instance the most probable assignment of the class variables

\[ h : \mathbf{X} \in \mathbb{R}^m \rightarrow \mathbf{Y} \in \{0, 1\}^d \]

• Probabilistically, this goal is equivalent to maximize the joint distribution of \( \mathbf{Y} \) given observation \( \mathbf{X} = \mathbf{x} \);
  i.e., the *maximum a posteriori* or MAP assignment of \( \mathbf{Y} = (Y_1, \ldots, Y_d) \)

\[ h^*(\mathbf{x}) = \arg \max_y P(\mathbf{Y} = y|\mathbf{X} = \mathbf{x}) \]
Existing Solutions (1/2)

- **Binary Relevance (BR)** [Clare and King, 2001; Boutell et al, 2004]
  - Transform a multi-label classification problem to **multiple single-label classification problems**
  - Learn an independent classifier for each class variable

- **Illustration**

<table>
<thead>
<tr>
<th>$D_{train}$</th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$Y_1$</th>
<th>$Y_2$</th>
<th>$Y_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n=1$</td>
<td>0.7</td>
<td>0.4</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$n=2$</td>
<td>0.6</td>
<td>0.2</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$n=3$</td>
<td>0.1</td>
<td>0.9</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$n=4$</td>
<td>0.3</td>
<td>0.1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$n=5$</td>
<td>0.8</td>
<td>0.9</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Existing Solutions (1/2)

• **Binary Relevance (BR)** [Clare and King, 2001; Boutell et al, 2004]
  
  • Transform a multi-label classification problem to **multiple single-label classification problems**
  
  • Learn **an independent classifier for each class variable**

• **BR maximizes the marginal probability of each class variable**
  
  • It does **not capture the dependence relations among the class variables**
  
  • It does **not find the MAP assignment of the class variables**
Existing Solutions (2/2)

- **Two-Layer classification approach** [Godbole and Sarawagi, 2004; Cheng and Hüllermeier, 2009]
- **Multi-label extension of $k$-nearest neighbor** [Zhang and Zhou, 2007]
- **Error-correcting output coding approach** [Hsu et al., 2009; Zhang and Schneider, 2012]
- **Classifier Chains methods** [Read et al., 2009; Dembczynski et al., 2010]
- **Multi-dimensional Bayesian networks** [van der Gaag and de Waal, 2006; Bielza et al., 2011]
Our Approaches

1. Conditional Tree-structured Bayesian Network (CTBN) [Batal et al., 2013]
2. Mixture of the tree-structured classifiers (MC) [Hong et al., 2014]
3. Multi-label mixtures-of-experts framework (ML-ME) [Hong et al., 2015]
Conditional Tree-structured Bayesian Network (CTBN)

• Motivation: The MLC problem is a hard problem, in that
  • For learning, the dependence relations among multiple class variables should be discovered and considered that have combinatorially large search space
  • For prediction, exponentially many label combinations should be evaluated
Conditional Tree-structured Bayesian Network (CTBN)

- Key idea: Restrict the class dependence structure to follow a directed tree
  - A class variable can have at most one other class variable as a parent without creating a cycle (dependencies among classes form a tree)
  - The feature vector $X$ is the common parent for all class variables

An example ($d=4$)
CTBN - Representation

- CTBN represents the conditional class distribution:

\[ P(y_1, ..., y_d|x) = \prod_{i=1}^{d} P(y_i|x, \pi(y_i, T)) \]

where \( T \) denotes a CTBN model, and \( \pi(y_i, T) \) is the parent of \( y_i \) in \( T \)

- CTBN decomposes the multivariate conditional class distribution as a product of the dependences in the network
CTBN - Representation

• CTBN represents the conditional class distribution:

\[ P(y_1, \ldots, y_d | x) = \prod_{i=1}^{d} P(y_i | x, \pi(y_i, T)) \]

where \( T \) denotes a CTBN model, and \( \pi(y_i, T) \) is the parent of \( y_i \) in \( T \)

• CTBN decomposes the multivariate conditional class distribution as a product of the dependences in the network

\[ P(y_1, y_2, y_3, y_4 | x) = P(y_3 | x) \cdot P(y_2 | x, y_3) \cdot P(y_1 | x, y_2) \cdot P(y_4 | x, y_2) \]
CTBN - Parameter Learning

- To parameterize CTBN

\[ P(y_1, \ldots, y_d | x) = \prod_{i=1}^{d} P(y_i | x, \pi(y_i, T)) \]

each \( P(y_i | x, \pi(y_i, T)) \) is represented by a **probabilistic classifier** function; e.g., logistic regression, naive Bayes, or maximum entropy classifiers.
CTBN - Discussion

• By restricting the dependence structure to a tree,
  
  1. The **optimal** dependence structure can be learned efficiently
     • Cast structure learning into the maximum branching tree [Tarjan, 1977] problem that optimizes the conditional log-likelihood of data
  
  2. **Exact MAP inference** can be done in $O(d)$ time
     • Present a variant of the max-product [Koller and Friedman, 2009] algorithm that performs exact MAP inference on CTBNs in a linear time
     • The tree-structure assumption lets us avoid evaluating all possible configurations of $Y$

• **Detailed description of the algorithms and relevant examples are provided in the dissertation document**
CTBN - Discussion

• Limitation of CTBN
  • The underlying dependence structure in data may not follow or be more complex than a tree
  • In such cases, a single CTBN cannot model the data properly
Multi-Label Mixtures-of-Experts (ML-ME)

• Motivation

• Dependence relations in a dataset could be more complex
  • What if a tree-structured model fails in learning a dataset?
  • What if there exist multiple dependence relations that may change across a dataset?
    • E.g., different class dependences may be found if a dataset contains instances like (relations between ‘cat’ and other labels change):

![Examples of different class dependences](image)
Multi-Label Mixtures-of-Experts (ML-ME)

- Key idea: Improve the MLC framework by
  1. Generalizing the structural assumptions in MLC model learning
  2. Incorporating multiple MLC models using a mixture framework
    - Adopt the mixtures-of-experts [Jacobs et al., 1991] framework that can represent variable dependence relations across a dataset

An example ML-ME ($d=4$)
ML-ME - Representation

• Generalizing CTBN
  • Let the structure assumption be specified by the user:
    \[
P(Y|X; M) = \prod_{i=1}^{d} P(Y_i|X, \pi(Y_i, M))\]
    where \( \pi(Y_i, M) \) denotes the parent class variables of \( Y_i \) defined by model \( M \)
  • This generalizes a number of existing MLC models (BR, CTBN, CC, …)
  • Reformulate the existing MLC models as a decomposition of \( P(Y|X) \) using a product of the univariate class posteriors \( P(Y_i|X, \pi(Y_i, M)) \)
ML-ME - Representation

• By specifying different structural assumptions, one can instantiate different structured probabilistic MLC models:

\[
P(Y|X; M) = \prod_{i=1}^{d} P(Y_i|X, \pi(Y_i, M))
\]

<table>
<thead>
<tr>
<th>Model</th>
<th>BR [Boutell et al., 2004]</th>
<th>CTBN</th>
<th>CC [Read et al., 2009]</th>
<th>DBR [Montañes et al., 2014]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Independent</td>
<td>Tree</td>
<td>Chain</td>
<td>Circular chain</td>
</tr>
<tr>
<td>(\pi(Y_i, M))</td>
<td>{}</td>
<td>at most one other class variable</td>
<td>all preceding class variables</td>
<td>all other class variables</td>
</tr>
<tr>
<td>Example</td>
<td>((d=4))</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• We refer to these structured probabilistic graphical MLC models as **Classifier Chains Family (CCF)**
ML-ME - Representation

- ML-ME defines the multivariate posterior distribution \( P(Y|X) \) by combining multiple CCF models

\[
P(y|x) = \sum_{k=1}^{K} g_k(x) P(y|x, M_k)
\]

\[
= \sum_{k=1}^{K} g_k(x) \prod_{i=1}^{d} P(y_i|x, \pi_{M_k}(Y_i))
\]

- \( P(y|x, M_k) = \prod_{i=1}^{d} P(y_i|x, \pi_{M_k}(Y_i)) \) is the conditional joint distribution defined by the \( k \)-th CCF model;

- \( g_k(x) = P(M_k|x) \) is the softmax gating function reflecting how much \( M_k \) influences the prediction for \( x \)

\[
g_k(x) = \frac{\exp(\theta_{G_k} x)}{\sum_{k'=1}^{K} \exp(\theta_{G_{k'}} x)}
\]
ML-ME - Learning and Prediction Algorithms

- **Parameter learning algorithm**: Optimizes the parameters of ML-ME using the expectation-maximization (EM) approach

- **Structure learning algorithm**: Learns multiple CC or CTBN structures representing different dependence relations from data, using a boosting-style approach

- **Prediction algorithm**: Finds the maximum a posteriori (MAP) assignment of class variables using the *annealed MAP* [Yuan et al. 2004] approach
ML-ME - Experimental Evaluation

• Compared methods
  • Binary Relevance (BR) [Boutell et al., 2004, Clare et al., 2001]
  • Classifier chains (CC) [Read et al., 2009]
  • Probabilistic Classifier chains (PCC) [Demczynski et al., 2010]
  • Single CTBN (CTBN)
  • Ensemble of Classifier chains (ECC) [Read et al., 2009]
  • Ensemble of Probabilistic Classifier chains (EPCC) [Demczynski et al., 2010]
  • Mixtures-of-CTBN (ME-CTBN)
  • Mixtures-of-CC (ME-CC)
ML-ME - Experimental Evaluation

• Evaluation Metric

  • **Exact Match Accuracy (EMA)**

    • The probability of all class labels are being correctly predicted (higher is better)

      $$ EMA = \frac{1}{N} \sum_{n=1}^{N} \delta(y^{(n)}, h(x^{(n)})) $$

    • EMA is the most appropriate metric for evaluating MLC performances (MAP prediction)
ML-ME - Experimental Evaluation

- Datasets: 5 publicly available datasets from different domains

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Instances</th>
<th># Features</th>
<th># Classes</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>2,407</td>
<td>294</td>
<td>6</td>
<td>Image</td>
</tr>
<tr>
<td>Emotions</td>
<td>593</td>
<td>72</td>
<td>6</td>
<td>Music</td>
</tr>
<tr>
<td>Yeast</td>
<td>2,417</td>
<td>103</td>
<td>14</td>
<td>Biology</td>
</tr>
<tr>
<td>Medical</td>
<td>998</td>
<td>1,449</td>
<td>45</td>
<td>Clinical</td>
</tr>
<tr>
<td>Enron</td>
<td>1,702</td>
<td>1,001</td>
<td>53</td>
<td>Text</td>
</tr>
</tbody>
</table>
ML-ME - Experimental Evaluation

- **Metric: EMA** *higher is better*

Numbers in boldface indicate the best results (averages over 10 fold cross validation; by paired t-test at $\alpha=0.05$) on each dataset; The last row shows the average ranking (lower is better)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>BR</th>
<th>CC</th>
<th>PCC</th>
<th>CTBN Our method</th>
<th>ECC</th>
<th>EPCC</th>
<th>ME-CTBN Our method</th>
<th>ME-CC Our method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>0.54</td>
<td>0.63</td>
<td>0.67</td>
<td>0.63</td>
<td>0.66</td>
<td>0.67</td>
<td>0.65</td>
<td>0.71</td>
</tr>
<tr>
<td>Emotions</td>
<td>0.27</td>
<td>0.27</td>
<td>0.32</td>
<td>0.32</td>
<td>0.29</td>
<td>0.34</td>
<td>0.37</td>
<td>0.36</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.15</td>
<td>0.19</td>
<td>0.23</td>
<td>0.19</td>
<td>0.20</td>
<td>0.22</td>
<td>0.22</td>
<td>0.26</td>
</tr>
<tr>
<td>Medical</td>
<td>0.64</td>
<td>0.69</td>
<td>-</td>
<td>0.67</td>
<td>0.70</td>
<td>-</td>
<td>0.71</td>
<td>0.71</td>
</tr>
<tr>
<td>Enron</td>
<td>0.17</td>
<td>0.20</td>
<td>-</td>
<td>0.18</td>
<td>0.18</td>
<td>-</td>
<td>0.20</td>
<td>0.19</td>
</tr>
<tr>
<td>Average Rank* lower is better</td>
<td>8.0</td>
<td>5.4</td>
<td>4.8</td>
<td>5.6</td>
<td>3.3</td>
<td>4.6</td>
<td>2.7</td>
<td>1.8</td>
</tr>
</tbody>
</table>

- PCC and EPCC did not finish on Medical and Enron within 24 hours.
**ML-ME - Experimental Evaluation**

- **Metric:** EMA *higher is better*

Numbers in boldface indicate the best results (averages over 10 fold cross validation; by paired t-test at $\alpha=0.05$) on each dataset; The last row shows the average ranking (lower is better)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>BR</th>
<th>CC</th>
<th>PCC</th>
<th>CTBN</th>
<th>ECC</th>
<th>EPCC</th>
<th>ME-CTBN</th>
<th>ME-CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>0.54</td>
<td>0.63</td>
<td>0.67</td>
<td>0.63</td>
<td>0.66</td>
<td>0.67</td>
<td>0.65</td>
<td>0.71</td>
</tr>
<tr>
<td>Emotions</td>
<td>0.27</td>
<td>0.27</td>
<td>0.32</td>
<td>0.32</td>
<td>0.29</td>
<td>0.34</td>
<td>0.37</td>
<td>0.36</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.15</td>
<td>0.19</td>
<td>0.23</td>
<td>0.19</td>
<td>0.20</td>
<td>0.22</td>
<td>0.22</td>
<td>0.26</td>
</tr>
<tr>
<td>Medical</td>
<td>0.64</td>
<td>0.69</td>
<td>-</td>
<td>0.67</td>
<td>0.70</td>
<td>-</td>
<td>0.71</td>
<td>0.71</td>
</tr>
<tr>
<td>Enron</td>
<td>0.17</td>
<td>0.20</td>
<td>-</td>
<td>0.18</td>
<td>0.18</td>
<td>-</td>
<td>0.20</td>
<td>0.19</td>
</tr>
<tr>
<td>Average Rank</td>
<td><strong>8.0</strong></td>
<td><strong>5.4</strong></td>
<td><strong>4.8</strong></td>
<td><strong>5.6</strong></td>
<td><strong>3.3</strong></td>
<td><strong>4.9</strong></td>
<td><strong>2.7</strong></td>
<td><strong>1.8</strong></td>
</tr>
</tbody>
</table>

- PCC and EPCC did not finish on Medical and Enron within 24 hours.
**ML-ME - Experimental Evaluation**

- **Metric: EMA** *higher is better*

Numbers in boldface indicate the best results (averages over 10 fold cross validation; by paired t-test at $\alpha=0.05$) on each dataset; The last row shows the average ranking (lower is better)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>BR</th>
<th>CC</th>
<th>PCC</th>
<th>CTBN</th>
<th>ECC</th>
<th>EPCC</th>
<th>ME-CTBN</th>
<th>ME-CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>0.54</td>
<td>0.63</td>
<td>0.67</td>
<td>0.63</td>
<td>0.66</td>
<td>0.67</td>
<td>0.65</td>
<td>0.71</td>
</tr>
<tr>
<td>Emotions</td>
<td>0.27</td>
<td>0.27</td>
<td>0.32</td>
<td>0.32</td>
<td>0.29</td>
<td>0.34</td>
<td>0.37</td>
<td>0.36</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.15</td>
<td>0.19</td>
<td>0.23</td>
<td>0.19</td>
<td>0.20</td>
<td>0.22</td>
<td>0.22</td>
<td>0.26</td>
</tr>
<tr>
<td>Medical</td>
<td>0.64</td>
<td>0.69</td>
<td>-</td>
<td>0.67</td>
<td>0.70</td>
<td>-</td>
<td>0.71</td>
<td>0.71</td>
</tr>
<tr>
<td>Enron</td>
<td>0.17</td>
<td>0.20</td>
<td>-</td>
<td>0.18</td>
<td>0.18</td>
<td>-</td>
<td>0.20</td>
<td>0.19</td>
</tr>
<tr>
<td><strong>Average Rank</strong> (<em>lower is better</em>)</td>
<td>0.0</td>
<td>3.3</td>
<td>4.6</td>
<td>5.8</td>
<td>3.3</td>
<td>4.6</td>
<td>2.7</td>
<td>1.8</td>
</tr>
</tbody>
</table>

- PCC and EPCC did not finish on Medical and Enron within 24 hours.
ML-ME - Experimental Evaluation

- Compare how much ML-ME improves the performance of CTBN
- Numbers in boldface indicate that ML-ME is significantly better than CTBN

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CTBN</th>
<th>ME-CTBN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>0.63</td>
<td>0.65</td>
</tr>
<tr>
<td>Emotions</td>
<td>0.32</td>
<td>0.37</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.19</td>
<td>0.22</td>
</tr>
<tr>
<td>Medical</td>
<td>0.67</td>
<td>0.71</td>
</tr>
<tr>
<td>Enron</td>
<td>0.18</td>
<td>0.20</td>
</tr>
</tbody>
</table>
ML-ME - Experimental Evaluation

- Compare how much ML-ME improves the performance of CC
  - Numbers in boldface indicate that ML-ME is significantly better than CC

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CC</th>
<th>ME-CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>0.63</td>
<td>0.71</td>
</tr>
<tr>
<td>Emotions</td>
<td>0.27</td>
<td>0.36</td>
</tr>
<tr>
<td>Yeast</td>
<td>0.19</td>
<td>0.26</td>
</tr>
<tr>
<td>Medical</td>
<td>0.69</td>
<td>0.71</td>
</tr>
<tr>
<td>Enron</td>
<td>0.20</td>
<td>0.19</td>
</tr>
</tbody>
</table>
CTBN & ML-ME - Summary

• **CTBN**: We presented the *Conditional Tree-structured Bayesian Networks* framework that has the following advantages:
  
  • The **optimal** tree-structured dependences can be learned efficiently
  
  • **Exact MAP inference** is performed in a **linear** time

• **ML-ME**: We presented the *Multi-Label Mixtures-of-Experts* framework
  
  • Introduced a **generalized representation** of structured MLC models
  
  • Developed a **probabilistic ensemble framework** for MLC
  
  • Proposed efficient supporting algorithms for **parameter and structure learning**, and **MAP prediction**

• Demonstrated through experiments that our mixture framework outperforms several state-of-the-art multi-label classification methods
MLC - Summary

• We studied and addressed the multi-label classification (MLC) problem
  • Motivated and defined the MLC problem
  • Overviewed some of existing MLC approaches and their limits
  • Presented three of our MLC solutions
    • CTBN is a tree-structured probabilistic graphical model for MLC
      • CTBN offers both efficiency and optimality in performing the MLC tasks
    • MC constructs a probabilistic ensemble of multiple CTBN models
      • MC leverages the efficiency of CTBN and the ability of a mixture ensemble
    • ML-ME builds an ensemble mixture that incorporates with multiple CCF models, that CCF is a class of structured probabilistic graphical MLC models
      • ML-ME recovers a rich set of dependence relations in multi-label data
Part II

Conditional Outlier Detection (COD)
Part II - Conditional Outlier Detection

• Agenda
  • Motivation
    • Existing outlier detection approaches
  • Problem definition
  • Our approaches
    • Probabilistic model-based approach
    • Ratio-based meta-analytic approach
Motivation

• Outlier Detection

• A data analytic task that finds atypical behaviors, unusual outcomes, or erroneous readings and annotations
Motivation

• Outlier Detection
  • A data analysis technique that finds atypical behaviors, unusual outcomes, or erroneous readings and annotations

• Applied to various areas:
  • A primary data preprocessing step that helps to remove noisy or irrelevant signals in data [Hodge and Austin, 2004; Liu et al., 2004]

  • A method to discover rare or interesting patterns in data [Fawcett and Provost, 1997; Tan et al., 2002; Wong et al., 2003; Bay and Schwabacher, 2003; Hauskrecht et al., 2007]

  • E.g., Fraud detection, Network intrusion surveillance, Patient monitoring and alerting
Motivation

• Most existing works focus on \textit{unconditional} outliers
  
  • Unconditional outliers are assumed to manifest in the \textit{joint space of all data attributes}
  
• If outliers occur \textit{conditionally}, the existing solutions may not work!
  
  • Conditional outliers are assumed to be present in the \textit{response (output)} space, \textit{conditioned on the context (input)} of data instances
Motivation

- Example unconditional and multivariate conditional outliers (marked in red)

**Unconditional:**
Find rare images
Corresponds to seeking low $P(x)$ where $x$ is an image instance

**Conditional:**
Find incorrect/rare image annotations
Corresponds to seeking low $P(y|x)$ where $y$ denotes image tags; $x$ is an image instance
Problem Definition

• Conditional outlier detection (COD)

  • Each data instance consists of
    \( m \)-dimensional continuous input (context) \( x = (x_1, ..., x_m) \) and
    its \( d \)-dimensional binary output (response) \( y = (y_1, ..., y_d) \)

  • **Goal**: Given input-output data pairs, to identify the instances that are
    with unusual association of input and output

    • Equiv., to find the instances with an unusual output \( y \) given their input \( x \)

  • To our knowledge, this is the first work in the literature that formally defines and
    tackles the COD problem with multi-dimensional binary output
Problem Definition

- Conditional outlier detection (COD) can be categorized by the output dimensionality
  - When $d = 1$ (output dimensionality): univariate COD (UCOD)
  - Otherwise: multivariate COD (MCOD)
Our Approaches

• We present *two* approaches

1. **Probabilistic model-based** approach
   • Develop a new probabilistic COD framework, by extending one of the successful COD approaches

2. **Ratio-based meta-analysis** approach
   • Define a new conditional outlier score that can incorporate with existing unconditional outlier scoring methods
• Key idea

  • Use a **probabilistic model** to represent the **conditional dependence relations** in data and to examine instances for conditional outliers

    • Instances with a **low probability estimate** \( \tilde{P}(Y=y|X=x; \mathcal{M}) \) are considered as conditional outliers
Probabilistic Approach to UCOD

- A two-phase approach

1. Build a data model \( \mathcal{M} \) of conditional probability \( P(Y|X) \)

2. Compute outlier scores by estimating \( \tilde{P}(Y=y|X=x; \mathcal{M}) \)

**Phase 1: Data Modeling**

\[ f(X) = P(Y|X;\mathcal{M}) \]

**Phase 2: Outlier Scoring**

\[ \text{diff}(f(x^{(n)}), y^{(n)}) \]
**Phase 1: Model Building**

- **Objective:** To learn the dependence relation from input to output by obtaining an accurate data model $\mathcal{M}$ of univariate conditional $P(Y|X)$

- The $L_2$-regularized logistic regression model is used to represent $P(Y|X)$
  - A simple linear model that produces probabilistic output
  - Can handle the input attributes defined by a mixture of continuous and discrete values
Objective: To compute outlier scores using the obtained data model $\mathcal{M}$; such that the higher the score is, the more likely the instance is an outlier.

- Outliers are associated with low probabilities.
- We use the negative logarithm to compute outlier scores (for connections to the following MCOD extension):

$$Score_{\text{PROB}}(y^{(n)}|x^{(n)}) = - \log \tilde{P}(y^{(n)}|x^{(n)}; \mathcal{M})$$

- This preserves the original ordering of the probability estimates.

\[f(X) = P(Y|X; \mathcal{M}) \quad \text{diff}(f(x^{(n)}), y^{(n)})\]
**PROB MCOD** - Probabilistic Approach to MCOD

- **Key idea:** *how to extend PROB to MCOD?*
  - Tackle the MCOD problem by building a probabilistic model of conditional joint $P(Y|X)$
  - Apply the structured (decomposed) data models studied in MLC to support probabilistic MCOD

---

**Phase 1: Data Modeling**

- Data Instances
- $f(X) = P(Y_1, \ldots, Y_d | X; M)$
- Learning a Statistical Representation from Data
- A data model of conditional joint

**Phase 2: Outlier Scoring**

- $\text{diff}(f(x^{(n)}), y^{(n)})$
- Multivariate Outlier Score
• Key idea: how to extend PROB to MCOD?

• Tackle the MCOD problem by building a probabilistic model of conditional joint $P(Y|X)$

• Apply the structured (decomposed) data models studied in MLC to support probabilistic MCOD

• Our extension of PROB includes three variations of probabilistic MCOD scores:

1. MPROB-RELAX: Multivariate-PROB (MPROB) based on a RELAXed multi-dimensional output model

2. MPROB-RW: MPROB with Reliability Weights

3. MPROB-LRW: MPROB with Local Reliability Weights
**Phase 1: Data Modeling**

- **Objective:** To learn **usual response patterns** in data by obtaining an accurate **probabilistic model of multivariate binary output**

- **Decompose** the conditional joint into a product of conditional univariate distributions (chain rule of probability) \[ \text{[Read et al., 2009]} \]

\[
P(Y_1, \ldots, Y_d|X) = \prod_{i=1}^{d} P(Y_i|X, \pi(Y_i))
\]

where \( \pi(Y_i) \) denotes the parents of \( Y_i \)

- Again, the logistic regression model is used to **handle** input spaces defined by a mixture of continuous and discrete variables \( (X, \pi(Y_i)) \)
Consider two chains with different orders:

- $P(y|x) = P(y_1|x) \ P(y_2|x, y_1) \ P(y_3|x, y_1, y_2)$
- $P(y|x) = P(y_3|x) \ P(y_2|x, y_3) \ P(y_1|x, y_2, y_3)$
Phase 1: Data Modeling

- Practical issues: Chain orders
  - Different chain orders produce different conditional joint distribution models [Dembczynski et al., 2010]

- Resolution
  - Relax the chain rule by permitting circular dependences among the output variables:
    \[
    \Psi(Y_1, \ldots, Y_d|x) = \prod_{i=1}^{d} P(Y_i|X, Y_{-i})
    \]
    where \(Y_{-i}\) denotes all output variables other than \(Y_i\)
  - By regularizing the base statistical models (logistic regression), the dependence relations could be recovered
- Phase 2: Outlier Scoring

- Objective: To compute **conditional outlier scores** with the help of the obtained probabilistic model

- The scores are computed along with the decomposition:

\[
Score_{\text{MPROB}}(y^{(n)}|x^{(n)}) = - \log \tilde{P}(y^{(n)}|x^{(n)}; \mathcal{M})
\]

\[
= \sum_{i=1}^{d} - \log \tilde{P}(y_{i}^{(n)}|x^{(n)}, \pi(y_{i}^{(n)}); \mathcal{M})
\]

\[
= \sum_{i=1}^{d} Score_{\text{PROB}(i)}
\]

\[
\text{:: Decomposition}
\]

\[
\text{:: Negative logarithm}
\]

\[
\text{:: Reduction to UCOD}
\]

- Remember that we have relaxed the chain rule, the conditional outlier scores are computed as:

\[
Score_{\text{MPROB-RELAX}}(y^{(n)}|x^{(n)}) = \sum_{i=1}^{d} - \log \tilde{P}(y_{i}^{(n)}|x^{(n)}, y_{-i}^{(n)}; \mathcal{M})
\]
- Phase 2: Outlier Scoring

- Practical issues

  - The quality of individual models trained on finite size data may be inconsistent
    - Some dimensions of $Y_i | X, \pi(Y_i)$ may not fit well the base statistical assumption (a logistic curve)
    - As a result, our data model may result in miscalibrated (inaccurate) probability estimations

→ If we treat all dimensions of $P(Y_i | X, \pi(Y_i))$ equally reliable, the resulting scores may degenerate to a noisy vector
**Phase 2: Outlier Scoring**

- **Resolution**
  - We introduce a weight term $w_i$ in the score:
    \[
    \text{Score}_{\text{MPROB-RW}}(y^{(n)}|x^{(n)}) = \sum_{i=1}^{d} -w_i \log \tilde{P}(y_i^{(n)}|x^{(n)}, y_{-i}^{(n)}; \mathcal{M})
    \]
  - $w_i$ controls the influence of individual models towards the outlier score

- **Reliability weights ($w_i$)**
  - Motivated by the Brier score [Brier, 1950], we define the reliability weights as the mean estimation error of the probabilistic output:
    \[
    w_i = \frac{N}{\sum_{n=1}^{N} \epsilon_i^{(n)}}
    \]
    where $\epsilon_i^{(n)} = 1 - P(Y_i=y_i^{(n)}|X=x^{(n)}, \pi(Y_i)=\pi(y_i^{(n)}))$ denotes the error in the probability estimate.
Experiments

• Compared methods
  • **LOF**: Local Outlier Factor [Breunig et al., 2000] applied to the joint space of all data attributes (both input and output)
  • **M-PROB**: Our base MPROB approach (MPROB-RELAX)
  • **M-RW**: Our MPROB approach with the reliability weights (MPROB-RW)

• Parameter settings
  • [LOF] distance metric: Mahalanobis distance; \#neighbors \( k = 50 \)
  • [M-PROB, M-RW] data models: \( L_2 \)-regularized logistic regression
    (regularization parameters are chosen by internal cross validation)


- **Experiments**

  - **Real-world Datasets**

    | Dataset   | $N / m / d$ | Domain | Description |
    |-----------|-------------|--------|-------------|
    | Birds     | 645 / 276 / 19 | Sound | Bird songs | Species |
    | Yahoo-arts | 7,484 / 1,836 / 26 | Text | News articles | Topics |
    | Mediamill | 43,907 / 120 / 101 | Video | Video frames | Concepts |

  - **Experiments with simulated outliers**

    - For our comparative evaluation, we simulate multivariate conditional outliers by perturbing the output space of data
- Experiments

  - Experimental (simulation) setup

    - In each simulation, we simulate conditional outliers by:

      1. Randomly select 1% of the instances in the dataset (outlier ratio = 1%)

      2. For each \((x^{(k)}, y^{(k)})\) of the selected instances, randomly pick \(\{5, 20\}\%\) of output dimension \(l\) (outlier dimensionality = \(\{5, 20\}\%\))

      3. Flip the value of the selected dimension; i.e., \(y_l^{(k)} := |y_l^{(k)} - 1|\)

  - Illustration:

<table>
<thead>
<tr>
<th></th>
<th>(X_1)</th>
<th>(X_2)</th>
<th>(Y_1)</th>
<th>(Y_2)</th>
<th>(Y_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(n=1)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=2)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=3)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(n=4)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(n=5)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=6)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
• Experimental (simulation) setup

  • In each simulation, we simulate conditional outliers by:

  ✔ 1. Randomly select 1% of the instances in the dataset (outlier ratio = 1%)

  2. For each \((x^{(k)}, y^{(k)})\) of the selected instances, randomly pick \{5, 20\}% of output dimension \(l\) (outlier dimensionality = \{5, 20\}%)

  3. Flip the value of the selected dimension; i.e., \(y_l^{(k)} := |y_l^{(k)} - 1|\)

• Illustration:

<table>
<thead>
<tr>
<th>(n=1)</th>
<th>(X_1)</th>
<th>(X_2)</th>
<th>(Y_1)</th>
<th>(Y_2)</th>
<th>(Y_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>(n=2)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=3)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(n=4)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(n=5)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=6)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Experiments

- Experimental (simulation) setup

- In each simulation, we simulate conditional outliers by:
  1. Randomly select 1% of the instances in the dataset (outlier ratio = 1%)
  2. For each \((x^{(k)}, y^{(k)})\) of the selected instances, randomly pick \{5, 20\}% of output dimension \(l\) (outlier dimensionality = \{5, 20\}%)
  3. Flip the value of the selected dimension; i.e., \(y_l^{(k)} := |y_l^{(k)} - 1|\)

- Illustration:

<table>
<thead>
<tr>
<th></th>
<th>(X_1)</th>
<th>(X_2)</th>
<th>(Y_1)</th>
<th>(Y_2)</th>
<th>(Y_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(n=1)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=2)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=3)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(n=4)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(n=5)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(n=6)</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
- Experiments

- Experimental (simulation) setup

  - In each simulation, we simulate conditional outliers by:

    ✔ 1. Randomly select 1% of the instances in the dataset (outlier ratio = 1%)
    ✔ 2. For each \((x^{(k)}, y^{(k)})\) of the selected instances, randomly pick \(\{5, 20\}\%\) of output dimension \(l\) (outlier dimensionality = \(\{5, 20\}\%\))
    ✔ 3. Flip the value of the selected dimension; i.e., \(y_l^{(k)} := |y_l^{(k)} - 1|\)

- Illustration:

<table>
<thead>
<tr>
<th>(n)</th>
<th>(X_1)</th>
<th>(X_2)</th>
<th>(Y_1)</th>
<th>(Y_2)</th>
<th>(Y_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.xx</td>
<td>0.xx</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.xx</td>
<td>0.xx</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\(\leftarrow y_2^{(2)} := 0\)

\(\leftarrow y_6^{(1)} := 1\)
Experiments

- Evaluation metrics
  - Precision-alert rate (PAR) curves
  - Average precision-alert rate in $[0.00, 0.01]$ range ($\text{APAR}_{[0.00, 0.01]}$)
**PROB MCOD - Results**

- **Metric: PAR Curves** *Higher is better*

![Graphs showing PAR curves for different datasets and outlier dimensions.](image-url)
**Results**

- **Metric:** $\text{APAR}_{[0.00, 0.01]}$ *Higher is better*

  *Numbers in boldface indicate the best results (averages over 5 repeats; by paired t-test at $\alpha=0.05$)*

<table>
<thead>
<tr>
<th></th>
<th>Outlier Dimensionality = 5%</th>
<th></th>
<th>Outlier Dimensionality = 20%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LOF</td>
<td>M-PROB</td>
<td>M-RW</td>
</tr>
<tr>
<td></td>
<td>Birds</td>
<td>0.04 ± 0.08</td>
<td>0.39 ± 0.25</td>
</tr>
<tr>
<td></td>
<td>Yahoo-arts</td>
<td>0.00 ± 0.01</td>
<td>0.04 ± 0.04</td>
</tr>
<tr>
<td></td>
<td>Mediamill</td>
<td>0.20 ± 0.17</td>
<td>0.57 ± 0.14</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Birds</td>
<td>0.32 ± 0.22</td>
<td>0.78 ± 0.19</td>
</tr>
<tr>
<td></td>
<td>Yahoo-arts</td>
<td>0.00 ± 0.00</td>
<td>0.25 ± 0.09</td>
</tr>
<tr>
<td></td>
<td>Mediamill</td>
<td>0.30 ± 0.12</td>
<td>0.99 ± 0.01</td>
</tr>
</tbody>
</table>
Our Approaches

• We present \textit{two} approaches

1. Probabilistic model-based approach
   • Develop a new probabilistic COD framework, by extending one of the successful COD approaches

2. \textit{Ratio-based meta-analysis} approach
   • Define a new conditional outlier score that can incorporate with existing unconditional outlier scoring methods
Ratio-based Meta-Approach

• Motivation

  • The probabilistic approach may fail when the underlying data models do not produce well calibrated probability estimates
    → Consider a non-probabilistic approach

  • The current state-of-the-art conditional outlier detection does not take much advantages of the progress and solutions of unconditional approaches
    • Bridging the gap in development between unconditional and conditional outlier detection approaches would be an important research contribution
**ROS - Ratio-based Meta-Approach**

- **Key idea**
  
  - Examine instances for conditional outliers by comparing (via ratio) two unconditional outlier scores defined over the input space:

    \[
    \frac{\text{Score against the instances with the same output value}}{\text{Score against the instances with the opposite output value}}
    \]

  - We refer to this approach as *Ratio of Outlier Scores (ROS)*
ROS - Ratio-based Meta-Approach

• Key idea
  
  • **Meta-approach**: Users can choose which unconditional outlier scoring method to use

• Many existing unconditional outlier scores can serve ROS
  
  • e.g., density-based outlier scores [Breunig et al., 2000; Papadimitriou et al., 2003],
    distance-based outlier scores [Knorr and Ng, 1997], etc.

• In the work, we use and validate ROS with in combination with **Local Outlier Factor (LOF)** [Breunig et al., 2000]
  
  • One of the most widely used nonparametric unconditional outlier score

  • LOF computes its outlier score by comparing the local density of the instance and the average local density of its $k$ nearest neighbors
ROS-MCOD - Ratio-based Meta-Approach to MCOD

- Key idea: *how to apply ROS to MCOD?*
  - To define a new MCOD score using ROS, apply the decomposition schema as used in the probabilistic model-based approach.
  - Also, a discriminative dimensionality reduction is considered for a reliable outlier detection.

- To this end, we present *two* variations of the ratio-based MCOD scores:
  1. **ROS-M**: *ROS on Multi-dimensional Output*
  2. **ROS-MDP**: *ROS on Multi-dimensional Discriminative Projections*
COD - Summary

• We explored the conditional outlier detection (COD) problem
  • Motivated and formalized the COD problem
  • Compared the COD and unconditional outlier detection problems
  • Presented two of our COD approaches
    • Probabilistic model-based approach
    • Ratio-based meta-analysis approach
Conclusions
Conclusions

• We focused on data objects with multivariate binary output and two problems related to them:
  • *Multi-Label Classification* that studies modeling and prediction of multivariate output from complex input-output data
  • *Conditional Outlier Detection* that is concerned about how to identify contextually unusual output patterns in multivariate input-output data
Conclusions

• Our contribution to *Multi-Label Classification* (MLC) includes:
  
  • Conditional tree-structured probabilistic MLC framework

  • A generalization of a number of relevant MLC models (representing multivariate posterior distribution) that has BR, CC, CTBN as special cases

  • Two mixture ensemble MLC frameworks:
    • One works with the tree-structured models
    • Another works with any of BR, CC, or CTBN
Conclusions

• Our contribution to *Conditional Outlier Detection* (COD) includes:
  • Formalization of the multivariate COD problem
  • Application of decomposed multivariate (joint) posterior models to COD
  • Identification of the development gap between conditional and unconditional outlier detection approaches; which motivated us to propose a new ratio-base COD score (ROS)
  • Further generalizations of decomposition, which result in:
    • MCOD with relaxed (circular) dependence
    • MCOD with reliability weights
    • MCOD using the ratio-based COD scores
Future Directions

• Multi-label Classification

1. Considering nonlinear decision boundaries
   • Extend our MLC solutions with nonlinear probabilistic base models e.g., kernel SVMs [Shawe-Taylor and Cristianini, 2004] with a post-hoc calibration [Platt, 1999; DeGroot and Fienberg, 1983]

2. Dealing with class imbalance [He and Garcia, 2009]

3. Improving the prediction algorithm for the mixtures
   • Instead of the stochastic approximation process [Yuan et al., 2004] applied for the mixture models, a proper non-approximating prediction algorithm would be much preferred; e.g., dual decomposition [Sontag, 2010]
Future Directions

• Conditional Outlier Detection

  1. A theoretical justification or performance guarantees of the circular relaxation

  2. Effects of outliers to the base data representation

  3. The ROS approaches in combination with different unconditional outlier scores
Thank you!

Presentation title: Multivariate Data Modeling and Its Applications to Conditional Outlier Detection
Presenter: Charmgil Hong, Department of Computer Science
Presented date: August 9, 2017

Contact: charmgil@cs.pitt.edu
Bibliography

Bibliography


Bibliography


Bibliography


