A Few Useful Things to Know about Machine Learning

Author:
Pedro Domingos
Department of Computer Science and Engineering University of Washington

Published in: Communications of the ACM Volume 55 Issue 10, October 2012

Team of
Kiel Gordon - Matt Pymm - John Tuazon
Spring 2016

Background

Machine learning system can automatically learn programs form data, making them more attractive than manually constructing them.

The article focused on classification algorithms.

With machine learning integrated in many applications, there are many textbooks and resources but it is easy to overlook common “folk knowledge” that is needed to successfully to develop applications.

This article points out pitfalls to avoid, important issues to focus on, and answers to common questions

Key Lessons

1. Learning = Representation + Evaluation + Optimization
2. Its Generalization that counts
3. Data Alone is not enough
4. Overfitting has many faces
5. Intuition fails in high dimensions
6. Theoretical guarantees are not what they seem
7. Feature engineering is the key
8. More data beats a cleverer algorithm
9. Learn many models not just one
10. Simplicity does not imply accuracy
11. Representable does not imply learnable
12. Correlation does not imply Causation

Representation	Evaluation	Optimization
Instances	Accuracy/Error rate	Combinatorial optimization
K-nearest neighbor	Precision and recall	Greedy search
Support vector machines	Squared error	Beam search
Hyperplanes	Likelihood	Branch-and-bound
Naive Bayes	Posterior probability	Continuous optimization
Logistic regression	Information gain	Unconstrained
Decision trees	K-L divergence	Gradient descent
Sets of rules	Cost/Utility	Conjugate gradient
Propositional rules	Margin	Quasi-Newton methods
Logic programs		Constrained
Neural networks		Linear programming
Graphical models		Quadratic programming
Bayesian networks		
Conditional random fields		

Technical Highlights of problem solving (1)
Learning = Representation + Evaluation + Optimization

The paper described machine learning algorithms and being made up of three main components. Representation, Evaluation, and Optimization.

Representation - A classifier must be represented in a formal language that the computer can handle. Creating a set of classifiers the learner can learn is crucial.

Evaluation - An Evaluation function is needed to distinguish good classifiers from bad ones.

Optimization - Finally we need a method to search among the classifiers in the language for the highest scoring one. The choice of optimization technique is key to the efficiency of the algorithm.

Technical Highlights of problem solving (2)
1st Main Problem and Mitigation Strategy

1. Overfitting is the largest problem in Machine Learning.
   Low bias, High Variance

   Solutions:

   1. Strong false assumptions can be better than weak true ones, because a learner with latter needs more data to avoid overfitting.
   2. Cross-Validation: Too many parameter choices itself can cause overfitting.
   3. Regularization can penalize classifiers with more structure, promotes ones with less structure and thus avoids overfitting.
   4. Chi-Squared test before adding new structure. (Determine whether adding the structure actually affects the classification)
   5. Obtain more examples (data) in your training set.

Technical Highlights of problem solving (2)
2nd Main Problem and Mitigation Strategy

Curse of Dimensionality: Volume of the space increases so fast that the available data becomes sparse.

Intuition fails at high dimensions: Difficult to design a strong classifier because lack of understanding beyond 3D.
Fixed number of examples only cover a small fraction of input space.
Similarity-based reason fails, all objects appear sparse and dissimilar.

Solutions:

1. Blessing of non-uniformity: Examples are not spread uniformly.
2. Obtain more examples (data) to counter the large number of dimensions
3. Perform dimension reduction.
4. Feature engineering: Determining which features to use is the most important factor of a successful ML algorithm.

Technical Highlights of problem solving (3)
It’s Generalization that counts

The fundamental goal of machine learning is to generalize beyond the examples in the training set. So just because you can predict something with a very high degree of accuracy from your training set, if you can’t make an accurate prediction in the real world your learner becomes obsolete.

Contamination of your classifier by your test data can occur if you don’t take precautions. You can avoid contamination by randomly dividing your data into subsets and holding out each subset while training on the rest. Such as performing the Ten-Fold Cross-Validation or Leave-One-Out strategies discussed in our class.

Team’s review opinion on the work

Matt- Very informative. It pointed out a lot of problems that can arise out of machine learning that I overlooked. I would recommend to anyone interested in learning more about machine learning.

Kiel - I really enjoyed the white paper. It allowed me to gain a deeper understanding of the hidden facets of machine learning. The paper provided the key aspects in ML and exposed the many problems that can affect one's classifier. I definitely will keep this paper as future reference.

John- Its useful as it made me aware of the common pitfalls for machine learning.

Analysis of Crime Data of Sac & SF

Team of
Kiel Gordon - Matt Pymm - John Tuazon
Spring 2016

Overview

- Motivation
- Objectives
- Methods
- Schedule

Motivation Data Warehousing

Analyze various aspects of crime from Sacramento and San Francisco

1. Produce answers about spatial or temporal trends in crime.
2. To find common trends in crime, and potential correlations between them.
3. Compare the common trends in crime between the two cities.

Motivation Data Mining

1. Produce regions of crime in a map environment to show spatial hot spots.
2. Find potential overlap and correlations in areas of crime.
3. Observe formation of crime clusters during certain times of day or year.
4. A previous study done by The Stanford Center on Poverty and Inequality with the support of The Russell Sage Foundation showed that the percentage of crime in the US actually went down during the 2007-2010 recession. We hope our study shows similar results for the greater San Francisco and Sacramento area.

Objectives Data Warehousing

1. Preprocess the data.
2. Design and model database schema.
3. Load into Data Warehouse.
4. Design front end of website.
5. Design database and queries.

Objectives Data Mining

Cluster the crime data into regions using a spatial clustering algorithm.

Learn how to utilize common tools such as Weka and RapidMiner

1. Train ML algorithm with data.
2. Cross-validate the trained ML algorithm to check accuracy.
3. Repeat 1 & 2 till ML algorithm meets our desired accuracy threshold.
4. Repeat the three steps for all desired crimes in both cities.

Methods Data Warehouse

Queries (As of now):

1. What crime occurs most at certain days? Certain times?
2. What is the most common crimes?
3. Does the most common crime change over time?
    a. Month to Month?
    b. Year to Year?
    c. During a recession?

Tentative Schedule

Week 1: Data Preprocessing. Data transfer into our database

Week 2: Design and model database schema.

Week 3: Build The Data Warehouse

Week 4: Design and build frontend interface

Week 5: Finalize Data Mart

Week 6: Begin designing ML algorithm

Week 7: Finalize Algorithm and produce results

Limitations

1. Based off the assumption the data provide by Sac and SF is statistically accurate.

2. Time remaining in the semester.

Data Resource URL Links

Sacramento Open Data: http://data.cityofsacramento.org/home

http://data.cityofsacramento.org/dataviews/93308/sacramento-crime-data-from-one-year-ago/

http://data.cityofsacramento.org/dataviews/93307/sacramento-crime-data-from-current-year/

San Francisco Open Data: https://data.sfgov.org/

https://data.sfgov.org/Public-Safety/SFPD-Incidents-from-1-January-2003/tmnf-yvry

Crime Study: http://www.soc.umn.edu/~uggen/crime_recession.pdf