Introduction to Machine Learning & Its Application in Healthcare

Lecture 4

Oct 3, 2018

Presentation by: Leila Karimi
What Is Machine Learning?

• A branch of artificial intelligence, concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data.

• Arthur Samuel (1959). Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.

• Tom Mitchell (1998) Well-posed Learning Problem: A computer program is said to learn from experience $E$ with respect to some task $T$ and some performance measure $P$, if its performance on $T$, as measured by $P$, improves with experience $E$. 
What Is Machine Learning? Example

• “A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.”

• Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam.
  • Classifying emails as spam or not spam ---> Task T
  • Watching you label emails as spam or not spam ---> Experience E
  • The number (or fraction) of emails correctly classified as spam/not spam ---> Performance measure P
ML Applications
The Learning Setting

• Imagine learning algorithm is trying to decide which loan applicants are bad credit risks.

• Might represent each person by $n$ features. (e.g., income range, debt load, employment history, etc.)

• Take sample $S$ of data, labeled according to whether they were/weren’t good risks.

• Goal of algorithm is to use data seen so far produce good prediction rule (a “hypothesis”) $h(x)$ for future data.
The learning setting example

<table>
<thead>
<tr>
<th>% down</th>
<th>recent delinq?</th>
<th>other accts</th>
<th>mmp/inc</th>
<th>high debt?</th>
<th>Good risk?</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>N</td>
<td>Y</td>
<td>0.32</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>10</td>
<td>N</td>
<td>N</td>
<td>0.25</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>5</td>
<td>Y</td>
<td>N</td>
<td>0.30</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>20</td>
<td>N</td>
<td>Y</td>
<td>0.31</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>5</td>
<td>N</td>
<td>N</td>
<td>0.42</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>10</td>
<td>Y</td>
<td>N</td>
<td>0.38</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>10</td>
<td>N</td>
<td>N</td>
<td>0.25</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

- Given this data, some reasonable rules might be:
  - Predict YES iff (!recent delinq) AND (%down > 5).
  - Predict YES iff 100*[mmp/inc] – 1*[%down] < 25.
  - ...
Big Questions

(A) How might we automatically generate rules that do well on observed data? --- > Algorithms

(B) What kind of confidence do we have that they will do well in the future? --- > Performance Evaluation
The machine learning framework

\[ y = f(x) \]

**Training:** given a training set of labeled examples \( \{(x_1, y_1), ..., (x_n, y_n)\} \), estimate the prediction function \( f \) by minimizing the prediction error on the training set.

**Testing:** apply \( f \) to a never before seen *test example* \( x \) and output the predicted value \( y = f(x) \)
ML in a Nutshell

• Every machine learning algorithm has three components:
  • Representation
  • Evaluation
  • Optimization
Representation

• Decision trees
• Sets of rules / Logic programs
• Graphical models (Bayes/Markov nets)
• Neural networks
• Support vector machines
• …
Evaluation

• Accuracy
• Precision and recall
• Squared error
• Likelihood
• Posterior probability
• Cost / Utility
• Margin
• Entropy
• K-L divergence
• ...
Optimization

- Combinatorial optimization
  - E.g.: Greedy search

- Convex optimization
  - E.g.: Gradient descent

- Constrained optimization
  - E.g.: Linear programming
Machine Learning Algorithms

• Supervised Learning
  • Training data includes desired outputs

• Unsupervised Learning
  • Training data does not include desired outputs

• Semi-supervised learning
  • Training data includes a few desired outputs

• Others: Reinforcement learning, recommender systems
Supervised Learning
Supervised learning process: two steps

• Learning (training): Learn a model using the training data
• Testing: Test the model using unseen test data to assess the model accuracy

\[ Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}} \]
Unsupervised Learning

• Learning patterns from unlabeled data

• Tasks
  • understanding and visualization
  • anomaly detection
  • information retrieval
  • data compression
Unsupervised Learning (Cont.)

Slide credit: Yi-Fan Chang
Supervised Learning (Cont.)

- Supervised learning categories and techniques
  - **Linear classifier** (numerical functions)
  - **Parametric** (Probabilistic functions)
    - Naïve Bayes, Gaussian discriminant analysis (GDA), Hidden Markov models (HMM), Probabilistic graphical models
  - **Non-parametric** (Instance-based functions)
    - K-nearest neighbors, Kernel regression, Kernel density estimation, Local regression
  - **Non-metric** (Symbolic functions)
    - Classification and regression tree (CART), decision tree
  - **Aggregation**
    - Bagging (bootstrap + aggregation), Adaboost, Random forest
Unsupervised Learning (Cont.)

• Unsupervised learning categories and techniques
  • Clustering
    • K-means clustering
    • Spectral clustering
  • Density Estimation
    • Gaussian mixture model (GMM)
    • Graphical models
  • Dimensionality reduction
    • Principal component analysis (PCA)
    • Factor analysis
Supervised Learning: Linear Classifier

- Find a linear function to separate the classes
- Techniques:
  - Perceptron
  - Logistic regression
  - Support vector machine (SVM)
  - Ada-line
  - Multi-layer perceptron (MLP)

\[ g(x_n) = \text{sign}(w^T x_n) \]

, where \( w \) is an \( d \)-dim vector (learned)
Supervised Learning: Non-Linear Classification

- Techniques:
  - Support vector machine (SVM)
  - Neural Networks
  - ...

\[ x_n = [x_{n1}, x_{n2}] \]

\[ x_n = [x_{n1}, x_{n2}, x_{n1} \cdot x_{n2}, x_{n1}^2, x_{n2}^2] \]

\[ g(x_n) = \text{sign}(w^T x_n) \]
Supervised Learning: Decision Trees

Should I wait at this restaurant?

[Decision Tree Diagram]

Slide credit: SRI International
Decision Tree Induction

(Recursively) partition examples according to the *most important* attribute.

Key Concepts

- **entropy**
  - impurity of a set of examples (entropy = 0 if perfectly homogeneous)
  - (#bits needed to encode class of an arbitrary example)

- **information gain**
  - expected reduction in entropy caused by partitioning
Decision Tree Induction: Decision Boundary
Supervised Learning: Neural Networks

• Motivation: human brain
  • massively parallel ($10^{11}$ neurons, ~20 types)
  • small computational units with simple low-bandwidth communication ($10^{14}$ synapses, 1-10ms cycle time)

• Realization: neural network
  • units ($\approx$ neurons) connected by directed weighted links
  • activation function from inputs to output
Neural Networks (continued)

\[ a_5 = g(W_{3,5} \cdot a_3 + W_{4,5} \cdot a_4) = g(W_{3,5} \cdot g(W_{1,3} \cdot a_1 + W_{2,3} \cdot a_2) + W_{4,5} \cdot g(W_{1,4} \cdot a_1 + W_{2,4} \cdot a_2)) \]

- Neural Network = parameterized family of nonlinear functions types
Neural Network Learning

• *Key Idea*: Adjusting the weights changes the function represented by the neural network (*learning = optimization in weight space*).

• Iteratively *adjust weights* to reduce *error* (difference between network output and target output).

• Weight Update
  • *perceptron training rule*
  • *linear programming*
  • *delta rule*
  • *backpropagation*
Neural Network Learning: Decision Boundary

- Single-layer perceptron
- Multi-layer network
Supervised Learning: Support Vector Machines

• *Kernel Trick*: Map data to *higher-dimensional space* where they will be *linearly separable*.

• Learning a Classifier: optimal linear separator is one that has the *largest margin* between positive examples on one side and negative examples on the other.

\[ \Phi: \mathbf{x} \rightarrow \varphi(\mathbf{x}) \]
Support Vector Machines: Decision Boundary
Supervised Learning: Nearest Neighbor Models

- **Key Idea**: Properties of an input $x$ are likely to be similar to those of points in the neighborhood of $x$.

- **Basic Idea**: Find ($k$) nearest neighbor(s) of $x$ and infer target attribute value(s) of $x$ based on corresponding attribute value(s).
Nearest Neighbor Model: Decision Boundary
Evaluating classification methods

- Predictive accuracy

\[
\text{Accuracy} = \frac{\text{Number of correct classifications}}{\text{Total number of test cases}},
\]

- Efficiency
  - time to construct the model
  - time to use the model

- Robustness: handling noise and missing values

- Scalability: efficiency in disk-resident databases

- Interpretability:
  - understandable and insight provided by the model

- Compactness of the model

Slide credit: Bing Liu
Performance Evaluation

• Randomly split examples into *training set* $U$ and *test set* $V$.
• Use training set to learn a hypothesis $H$.
• Measure % of $V$ correctly classified by $H$.
• Repeat for different random splits and average results.
Generalization

• Components of generalization error
  • **Bias**: how much the average model over all training sets differ from the true model?
    • Error due to inaccurate assumptions/simplifications made by the model
  • **Variance**: how much models estimated from different training sets differ from each other

• **Underfitting**: model is too “simple” to represent all the relevant class characteristics
  • High bias and low variance
  • High training error and high test error

• **Overfitting**: model is too “complex” and fits irrelevant characteristics (noise) in the data
  • Low bias and high variance
  • Low training error and high test error
Bias-Variance Trade-off

• Models with too few parameters are inaccurate because of a large bias (not enough flexibility).

• Models with too many parameters are inaccurate because of a large variance (too much sensitivity to the sample).
Machine Learning for Healthcare
Applying Machine Learning to Healthcare

• Healthcare sector is being transformed by the ability to record massive amounts of information

• Machine learning provides a way to automatically find patterns and reason about data

• It enables healthcare professionals to move to personalized care known as precision medicine.
Why to use ML?

- Adoption of Electronic Health Records (EHR) has increased 9x since 2008

[Henry et al., ONC Data Brief, May 2016]
Why to use ML?

• Large datasets
  • MIT Laboratory for Computational Physiology de-identified health data from ~40K critical care patients
  • Demographics, vital signs, laboratory tests, medications, notes, ...
  • Available data on nearly 230 million unique patients since 1995
Why to use ML?

• Diversity of digital health data
Why to use ML?

- Standardization
  - Diagnosis codes: ICD-9 and ICD-10 (International Classification of Diseases)
  - Laboratory tests: LOINC codes
  - Pharmacy: National Drug Codes (NDCs)
  - Unified Medical Language System (UMLS): millions of medical concepts

[https://blog.curemd.com/the-most-bizarreicd-10-codes-infographic/]
Industry interest in AI & healthcare

- Google DeepMind Health
  - CLINICIAN-LED TECHNOLOGY
  - Deep learning technology can save lives by helping detect curable diseases early

- IBM Watson for Oncology
  - Get oncologists the assistance they need to make more informed treatment decisions. Watson for Oncology analyzes a patient’s medical information against a vast array of data and expertise to provide evidence-based treatment options.

- Lumiata
  - Enabling healthcare to be predictive-first where health + care is proactive, hyper-personalized, and real-time.

Slide credit: David Sontag
What can machine learning do for the healthcare industry?

• Improve accuracy of diagnosis, prognosis, and risk prediction.
• Reduce medication errors and adverse events.
• Model and prevent spread of hospital acquired infections.
• Optimize hospital processes such as resource allocation and patient flow.
• Identify patient subgroups for personalized and precision medicine.
• Discover new medical knowledge (clinical guidelines, best practices).
• Automate detection of relevant findings in pathology, radiology, etc.

Improve quality of care and population health outcomes, while reducing healthcare costs.
Example Application: Improve accuracy of diagnosis and risk prediction

- New methods are developed for chronic disease risk prediction and visualization.
- These methods give clinicians a comprehensive view of their patient population, risk levels, and risk factors, along with the estimated effects of potential interventions.
Example Application: Optimize hospital processes

• By early and accurate prediction of each patient’s Diagnosis Related Group (DRG), demand for scarce hospital resources such as beds and operating rooms can be better predicted.
Example Application: Automate detection of relevant findings

• Pattern detection approaches have been successfully applied to detect regions of interest in digital pathology slides, and work surprisingly well to detect cancers.

• Automatic detection of anomalies and patterns is especially valuable when the key to diagnosis is a tiny piece of the patient’s health data.
Example Application: Breast Cancer Diagnosis

Research by Mangasarian, Street, Wolberg
Breast Cancer Diagnosis Separation

Research by Mangasarian, Street, Wolberg
Example Application: ICU Admission

- An emergency room in a hospital measures 17 variables (e.g., blood pressure, age, etc) of newly admitted patients.
- A decision is needed: whether to put a new patient in an intensive-care unit.
- Due to the high cost of ICU, those patients who may survive less than a month are given higher priority.
- Problem: to predict high-risk patients and discriminate them from low-risk patients.
What is unique about ML in healthcare?

• Life or death decisions
  • Need robust algorithms
  • Checks and balances built into ML deployment
  • (Also arises in other applications of AI such as autonomous driving)
  • Need fair and accountable algorithms

• Many questions are about unsupervised learning
  • Discovering disease subtypes, or answering question such as “characterize the types of people that are highly likely to be readmitted to the hospital”?

• Many of the questions we want to answer are causal
  • Naïve use of supervised machine learning is insufficient
What makes healthcare different?

- Often very little labeled data (e.g., for clinical NLP)
  - Motivates semi-supervised learning algorithms
- Sometimes small numbers of samples (e.g., a rare disease)
  - Learn as much as possible from other data (e.g. healthy patients)
  - Model the problem carefully
- Lots of missing data, varying time intervals, censored labels
What makes healthcare different?

• Difficulty of de-identifying data
  • Need for data sharing agreements and sensitivity

• Difficulty of deploying ML
  • Commercial electronic health record software is difficult to modify
  • Data is often in silos; everyone recognizes need for interoperability, but slow progress
  • Careful testing and iteration is needed