Machine Learning Approaches to Biological Research: Bioimage Informatics and Beyond

Robert F. Murphy
External Senior Fellow, Freiburg Institute for Advanced Studies
Ray and Stephanie Lane Professor of Computational Biology, Carnegie Mellon University
September 29–October 1, 2009

Outline

• Basic principles and paradigms of supervised and unsupervised machine learning
• Concepts of automated image analysis
• Approaches for creating predictive models from images
• Active learning paradigms for closed loop systems of cycles of experimentation, model refinement and model testing

What is Machine Learning?

• Fundamental Question of Computer Science: How can we build machines that solve problems, and which problems are inherently tractable/intractable?
• Fundamental Question of Statistics: What can be inferred from data plus a set of modeling assumptions, with what reliability?

Why Machine Learning?

• Learn relationships from large sets of complex data: Data mining
  – Predict clinical outcome from tests
  – Decide whether someone is a good credit risk
• Do tasks too complex to program by hand
  – Autonomous driving
• Customize programs to user needs
  – Recommend book/movie based on previous likes

Fundamental Question of Machine Learning

• How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?
  – Tom Mitchell

Tom Mitchell white paper

### Why Machine Learning?

- Economically efficient
- Can consider larger data spaces and hypothesis spaces than people can
- Can formalize learning problem to explicitly identify/describe goals and criteria

### Successful Machine Learning Applications

- Speech recognition
  - Telephone menu navigation
- Computer vision
  - Mail sorting
- Bio-surveillance
  - Identifying disease outbreaks
- Robot control
  - Autonomous driving
- Empirical science

Tom Mitchell white paper

### Machine Learning Paradigms

- **Supervised Learning**
  - Classification
  - Regression
- **Unsupervised Learning**
  - Clustering
- **Semi-supervised Learning**
  - Cotraining
  - Active learning

### Supervised Learning

- **Approaches**
  - Classification (discrete predictions)
  - Regression (continuous predictions)
- **Common considerations**
  - Representation (Features)
  - Feature Selection
  - Functional form
  - Evaluation of predictive power

### Classification vs. Regression

- If I want to predict whether a patient will die from a disease within six months, that is classification
- If I want to predict how long the patient will live, that is regression

### Representation

- Definition of thing or things to be predicted
  - Classification: classes
  - Regression: regression variable
- Definition of things (instances) to make predictions for
  - Individuals
  - Families
  - Neighborhoods, etc.
- Choice of descriptors (features) to describe different aspects of instances
Formal description

- Defining $X$ as a set of instances $x$ described by features
- Given training examples $D$ from $X$
- Given a target function $c$ that maps $X \rightarrow \{0, 1\}$
- Given a hypothesis space $H$
- Determine an hypothesis $h$ in $H$ such that $h(x) = c(x)$ for all $x$ in $D$

Inductive learning hypothesis

- Any hypothesis found to approximate the target function well over a sufficiently large set of training example will also approximate the target function over other unobserved example

Hypothesis space

- The hypothesis space determines the functional form
- It defines what are allowable rules/functions for classification
- Each classification method uses a different hypothesis space

Simple two class problem

Describe each image by features
Train classifier

k-Nearest Neighbor (kNN)

- In feature space, training examples are

k-Nearest Neighbor (kNN)

- We want to label '?'

Feature #1 (e.g., 'area')

Feature #2 (e.g., 'roundness')

Feature #3 (e.g., 'area')

Feature #4 (e.g., 'roundness')
k-Nearest Neighbor (kNN)
• Find k nearest neighbors and vote

Feature #1 (e.g., 'area')

Feature #2 (e.g., roundness)

for k=3, nearest neighbors are

So we label it +

Linear Discriminants
• Fit multivariate Gaussian to each class
• Measure distance from ? to each Gaussian

Decision trees
• Again we want to label ‘?’

Feature #1 (e.g., 'area')

Feature #2 (e.g., roundness)

Decision trees
• so we build a decision tree:

Decision trees
• so we build a decision tree:

Decision trees
• Goal: split address space in (almost) homogeneous regions

Decision trees
Support vector machines

- Again we want to label ‘?’

Support Vector Machines (SVMs)

- Use single linear separator??

Support Vector Machines (SVMs)

- Use single linear separator??

Support Vector Machines (SVMs)

- Use single linear separator??

Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

• we want to label ‘?’ - linear separator??
• A: the one with the widest corridor!

Support Vector Machines (SVMs)

• What if the points for each class are not readily separated by a straight line?
• Use the “kernel trick” – project the points into a higher dimensional space in which we hope that straight lines will separate the classes
• “kernel” refers to the function used for this projection

Support Vector Machines (SVMs)

• Definition of SVMs explicitly considers only two classes
• What if we have more than two classes?
• Train multiple SVMs
• Two basic approaches
  – One against all (one SVM for each class)
  – Pairwise SVMs (one for each pair of classes)
    – Various ways of implementing this

Questions

• What are the hypothesis spaces for
  – kNN classifier
  – Linear discriminants
  – Decision trees
  – Support Vector Machines

Cross-Validation

• If we train a classifier to minimize error on a set of data, have no ability to estimate (generalize) error that will be seen on new dataset
• To calculate generalizable accuracy, we use n-fold cross-validation
• Divide images into n sets, train using n-1 of them and test on the remaining set
• Repeat until each set is used as test set and average results across all trials
• Variation on this is called leave-one-out

Describing classifier errors

• For binary classifiers (positive or negative), define
  – TP = true positives, FP = false positives
  – TN = true negatives, FN = false negatives
  – Recall = TP / (TP + FN)
  – Precision = TP / (TP + FP)
  – F-measure = 2*Recall*Precision/(Recall + Precision)
Confusion matrix - binary

<table>
<thead>
<tr>
<th>True</th>
<th>Predicted</th>
<th>Positive</th>
<th>Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive</td>
<td>True Positive</td>
<td>False Negative</td>
<td></td>
</tr>
<tr>
<td>Negative</td>
<td>False Positive</td>
<td>True Negative</td>
<td></td>
</tr>
</tbody>
</table>

Precision-recall analysis

Describing classifier errors

- For multi-class classifiers, typically report
  - Accuracy = # test images correctly classified
  - Confusion matrix = table showing all possible combinations of true class and predicted class

Confusion matrix – multi-class

<table>
<thead>
<tr>
<th>True Class</th>
<th>Output of the Classifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNA</td>
<td>ER</td>
</tr>
<tr>
<td>DNA</td>
<td>0</td>
</tr>
<tr>
<td>ER</td>
<td>0</td>
</tr>
<tr>
<td>Gia</td>
<td>0</td>
</tr>
<tr>
<td>Gpp</td>
<td>0</td>
</tr>
<tr>
<td>Lam</td>
<td>0</td>
</tr>
<tr>
<td>Mit</td>
<td>0</td>
</tr>
<tr>
<td>Nuc</td>
<td>0</td>
</tr>
<tr>
<td>Act</td>
<td>0</td>
</tr>
<tr>
<td>TfR</td>
<td>0</td>
</tr>
<tr>
<td>Tub</td>
<td>0</td>
</tr>
</tbody>
</table>

Overall accuracy = 98%

Ground truth

- What is the source and confidence of a class label?
  - Most common: Human assignment, unknown confidence
  - Preferred: Assignment by experimental design, confidence ~100%

Feature selection

- Having too many features can confuse a classifier
- Can use comparison of feature distributions between classes to choose a subset of features that gets rid of uninformative or redundant features
Basic principle of feature selection

![Feature 1](feature1.png)  ![Feature 2](feature2.png)

![Feature 3](feature3.png)  ![Feature 4](feature4.png)

red=class 1, blue=class 2

Need to consider multivariate distance

![Figure from Guyon & Elisseeff](figure1.png)

Bad and Good Covariance

![Figure from Guyon & Elisseeff](figure2.png)

Feature Selection Methods

- Principal Components Analysis
- Non-Linear Principal Components Analysis
- Independent Components Analysis
- Information Gain
- Stepwise Discriminant Analysis
- Genetic Algorithms

Regression
Linear regression

Given examples $(x_i, y_i)_{i=1}^n$
Predict $y_{n+1}$ given a new point $x_{n+1}$

Ordinary Least Squares (OLS)

$y_i = w_0 + w_1 x_i$
$\hat{y}_i = X_i^T w$

Sum squared error

$\sum (X_i^T w - y_i)^2$

Beyond lines and planes

$\hat{y}_i = w_0 + w_1 x_i + w_2 x_i^2$

still linear in $w$

Geometric interpretation

$\hat{y} = w_1 x + w_2 x^2$

Assumptions vs. Reality

Intel sensor network data
Overfitting

Sensitivity to outliers

\[ E = \sum_i (x_i^T w - y_i)^2 = \sum_i E_i \]

Degree 15 polynomial

High weight given to outliers

Temperature at noon

Influence function

Kernel Regression

\[ \tilde{y}(x) = \frac{\sum_i y_i k(x_i - x)}{\sum_i k(x_i - x)} \]

Kernel regression (sigma=1)

Spline Regression

Regression on each interval

Spline Regression

With equality constraints

Cluster analysis

- Supervised learning (Classification) assumes classes are known
- Unsupervised learning (Cluster analysis) seeks to discover the classes
Formal description

- Given $X$ as a set of instances described by features
- Given an objective function $g$
- Given a partition space $H$
- Determine a partition $h$ in $H$ such that $h(X)$ maximizes/minimizes $g(h(X))$

**Hierarchical vs. $k$-means clustering**

- Two most popular clustering algorithms
- Hierarchical builds tree sequentially from the closest pair of points (wells/cells/probes/conditions)
- $k$-means starts with $k$ randomly chosen seed points, assigns each remaining point to the nearest seed, and repeats this until no point moves

**Hierarchical Clustering**

**K-means**

- Slide courtesy of Elvira Garcia Osuna
Choosing the number of Clusters

- A difficult problem
- Most common approach is to try to find the solution that minimizes the Bayesian Information Criterion

\[ BIC = -2 \ln L + k \ln(n) \]

Microarray raw data

- Label mRNA from one sample with a red fluorescence probe (Cy5) and mRNA from another sample with a green fluorescence probe (Cy3)
- Hybridize to a chip with specific DNAs fixed to each well
- Measure amounts of green and red fluorescence

Example microarray image

Data extraction

- Adjust fluorescent intensities using standards (as necessary)
- Calculate ratio of red to green fluorescence
- Convert to log₂ and round to integer
- Display saturated green = -3 to black = 0 to saturated red = +3

Distances

- High dimensionality
- Based on vector geometry – how close are two data points?
Distances

• High dimensionality
• Based on vector geometry – how close are two data points?

Array 1 | Array 2
---|---
Gene 1 | 1 | 4
Gene 2 | 1 | 3
...

Distance(Gene 1, Gene 2) = 1

Distances

• High dimensionality
• Based on vector geometry – how close are two data points?
• Use distances to determine clusters

Array 1 | Array 2
---|---
Gene 1 | 1 | 4
Gene 2 | 1 | 3
...

Distance(Gene 1, Gene 2) = 1

General Multivariate Dataset

• We are given values of \( p \) variables for \( n \) independent observations
• Construct an \( n \times p \) matrix \( M \) consisting of vectors \( X_1 \) through \( X_n \) each of length \( p \)

Multivariate Sample Mean

• Define mean vector \( \bar{I} \) of length \( p \)

\[
I(j) = \frac{\sum_{i=1}^{n} M(i,j)}{n} \quad \text{or} \quad \bar{I} = \frac{\sum_{i=1}^{n} X_i}{n}
\]

Multivariate Variance

• Define variance vector \( \sigma^2 \) of length \( p \)

\[
\sigma^2(j) = \frac{\sum_{i=1}^{n} (M(i,j) - I(j))^2}{n - 1}
\]

or

\[
\sigma^2 = \frac{\sum_{i=1}^{n} (X_i - \bar{I})^2}{n - 1}
\]
Covariance Matrix

• Define a $p \times p$ matrix $\text{cov}$ (called the covariance matrix) analogous to $\sigma^2$

\[
\text{cov}(j,k) = \frac{\sum_{i=1}^{n} (M(i,j) - I(j))(M(i,k) - I(k))}{n - 1}
\]

Covariance Matrix

• Note that the covariance of a variable with itself is simply the variance of that variable

\[
\text{cov}(j,j) = \sigma^2(j)
\]

Univariate Distance

• The simple distance between the values of a single variable $j$ for two observations $i$ and $l$ is

\[
M(i,j) - M(l,j)
\]

Univariate $z$-score Distance

• To measure distance in units of standard deviation between the values of a single variable $j$ for two observations $i$ and $l$ we define the $z$-score distance

\[
\frac{M(i,j) - M(l,j)}{\sigma(j)}
\]

Bivariate Euclidean Distance

• The most commonly used measure of distance between two observations $i$ and $l$ on two variables $j$ and $k$ is the Euclidean distance

\[
\sqrt{(M(i,j) - M(l,j))^2 + (M(i,k) - M(l,k))^2}
\]

Multivariate Euclidean Distance

• This can be extended to more than two variables

\[
\sqrt{\sum_{j=1}^{p} (M(i,j) - M(l,j))^2}
\]
Effects of variance and covariance on Euclidean distance

Points A and B have similar Euclidean distances from the mean, but point B is clearly "more different" from the population than point A.

Mahalanobis Distance

- To account for differences in variance between the variables, and to account for correlations between variables, we use the Mahalanobis distance

\[ D^2 = (X_i - \mu)\Sigma^{-1}(X_i - \mu)^T \]

Other distance functions

- We can use other distance functions, including ones in which the weights on each variable are learned
- Cluster analysis tools for microarray data most commonly use Pearson correlation coefficient

Input data for clustering

- Genes in rows, conditions in columns

<table>
<thead>
<tr>
<th>YORF</th>
<th>NAME</th>
<th>gWeight</th>
<th>Cell-cycle Alpha</th>
<th>Cell-cycle Beta</th>
<th>Alph</th>
</tr>
</thead>
<tbody>
<tr>
<td>YMR051W</td>
<td>YMR051W</td>
<td>1</td>
<td>0.03</td>
<td>0.3</td>
<td>0.37</td>
</tr>
<tr>
<td>YKL181W</td>
<td>YKL181W</td>
<td>1</td>
<td>0.33</td>
<td>-0.2</td>
<td>-0.12</td>
</tr>
<tr>
<td>YMR124W</td>
<td>YMR124W</td>
<td>1</td>
<td>-0.36</td>
<td>-0.08</td>
<td>-0.06</td>
</tr>
<tr>
<td>YHL020C</td>
<td>YHL020C</td>
<td>1</td>
<td>-0.01</td>
<td>-0.03</td>
<td>0.21</td>
</tr>
<tr>
<td>YGR072W</td>
<td>YGR072W</td>
<td>1</td>
<td>0.2</td>
<td>-0.43</td>
<td>-0.22</td>
</tr>
<tr>
<td>YGR149W</td>
<td>YGR149W</td>
<td>1</td>
<td>0.11</td>
<td>-1.15</td>
<td>-1.03</td>
</tr>
<tr>
<td>YGR185W</td>
<td>YGR185W</td>
<td>1</td>
<td>0.24</td>
<td>-0.23</td>
<td>0.12</td>
</tr>
<tr>
<td>YGL041C</td>
<td>YGL041C</td>
<td>1</td>
<td>0.06</td>
<td>0.23</td>
<td>0.2</td>
</tr>
<tr>
<td>YOR202W</td>
<td>YOR202W</td>
<td>1</td>
<td>0.1</td>
<td>0.48</td>
<td>0.86</td>
</tr>
<tr>
<td>YCR005C</td>
<td>YCR005C</td>
<td>1</td>
<td>0.34</td>
<td>1.64</td>
<td>1.23</td>
</tr>
<tr>
<td>YKR179W</td>
<td>YKR179W</td>
<td>1</td>
<td>0.71</td>
<td>0.03</td>
<td>0.11</td>
</tr>
<tr>
<td>YBR206C</td>
<td>YBR206C</td>
<td>1</td>
<td>-0.22</td>
<td>0.14</td>
<td>0.14</td>
</tr>
<tr>
<td>YMR244W</td>
<td>YMR244W</td>
<td>1</td>
<td>-0.06</td>
<td>-0.18</td>
<td>-0.38</td>
</tr>
<tr>
<td>YAR047C</td>
<td>YAR047C</td>
<td>1</td>
<td>-0.43</td>
<td>-0.56</td>
<td>-0.14</td>
</tr>
<tr>
<td>YMR317W</td>
<td>YMR317W</td>
<td>1</td>
<td>-0.43</td>
<td>-0.03</td>
<td>0.21</td>
</tr>
</tbody>
</table>

Clustering genes and conditions

- Rows and columns can be clustered independently - hierarchical is preferred for visualizing this
Stating Goals vs. Approaches

• Temptation when first considering using a machine learning approach to a biological problem is to describe the problem as automating the approach that you would solve the problem

• “I need a program to predict how much a gene is expressed by measuring how well its promoter matches a template”

Stating Goals vs. Approaches

• “I need a program that given a gene sequence predicts how much that gene is expressed by measuring how well its promoter matches a template”

• “I need a program that given a gene sequence predicts how much that gene is expressed by learning from sequences of genes whose expression is known”

Resources

• Association for the Advancement of Artificial Intelligence
  – http://www.aaai.org/AtTopics/pmwiki/pmwiki.php/AtTopics/MachineLearning

• Machine Learning – Mitchell, Carnegie Mellon
  – http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/mlbook.html

• Practical Machine Learning – Jordan, UC Berkeley
  – http://www.cs.berkeley.edu/~asimma/294-fall06/

• Learning and Empirical Inference – Rish, Tesauro, Jebara, Vadnik – Columbia
  – http://www1.cs.columbia.edu/~jebara/6998/