Machine learning applications in biotechnology research

UC Davis Biotechnology Program 2012
Davis, CA

Tobias Kind
UC Davis Genome Center
FiehnLab - Metabolomics
Machine learning

- Machine learning commonly used for prediction of future values
- Complex models (black box) do not always provide causal insight
- Predictive power is most important
Why Machine Learning?

People cost money, are slow, don’t have time

Let the machine (computer) do it...

Replace people with computers...

The Terminator (2029)
Machine Learning Personalities

Epicurus:  
Principle of multiple explanations  
“All consistent models should be retained”.

Epicurus  
(341 BC)

Occam's Razor:  
Of two equivalent theories or explanations, all other things being equal, the simpler one is to be preferred.

Ockham  
(1288)

Trust is good, control is better

Trust, but verify

Turing Test  
Can machines think?

Alan Turing  
(1912)

Artificial intelligence, Neural networks

Lenin*

Ronald Reagan*

Marvin Minsky  
(1927)

Sources: Wikipedia; (*) In principle
Machine Learning Algorithms

Unsupervised learning:

Supervised learning:

Transduction:

Clustering methods

Support vector machines
MARS (multivariate adaptive regression splines)
Neural networks
Naive Bayes classifier

Random Forest, Boosting trees, Honest trees,
Decision trees
CART (Classification and regression trees)
Genetic programming

Bayesian Committee Machine
Transductive Support Vector Machine

...thanks to WIKI and Stuart Gansky
Concept of predictive data mining for classification

Data Preparation
- Basic Statistics, Remove extreme outliers, transform or normalize datasets, mark sets with zero variances

Feature Selection
- Predict important features with MARS, PLS, NN, SVM, GDA, GA; apply voting or meta-learning

Model Training + Cross Validation
- Use only important features, apply bootstrapping if only few datasets;
  Use GDA, CART, CHAID, MARS, NN, SVM, Naive Bayes, kNN for prediction

Model Testing
- Calculate Performance with Percent disagreement and Chi-square statistics

Model Deployment
- Deploy model for unknown data;
  use PMML, VB, C++, JAVA
Automated machine learning workflows – tools of the trade

Statistica Dataminer workflow

WEKA KnowledgeFlow workflow

Statistica – academic license (www.onthehub.com/statsoft); WEKA – open source (www.cs.waikato.ac.nz/ml/weka/)
Massive parallel computing
Free lunch is over – concurrency in machine learning

Modern workstation (with 4-64 CPUs)

GPU computing (with 1000 stream processors)

Cloud computing (with 10,000 CPUs/GPUs)

Google Prediction API

Amazon Elastic MapReduce

Picture Source: ThinkMate Workstation; Wikipedia; Office; Google; Amazon
Common ML applications in biotechnology

**Classification** - genotype/wildtype, sick/healthy, cancer grading, toxicity prediction and evaluations (FDA, EPA)

Regression - predicting biological activities, toxicity evaluations, prediction of molecular properties of unknown substances (QSAR and QSPR)
Supervised learning with categorical data

Classification

<table>
<thead>
<tr>
<th>Category y</th>
<th>Value x1</th>
<th>Value x2</th>
<th>Value x3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample1</td>
<td>blue</td>
<td>615.4603</td>
<td>3.363</td>
</tr>
<tr>
<td>Sample2</td>
<td>blue</td>
<td>371.3181</td>
<td>3.491</td>
</tr>
<tr>
<td>Sample3</td>
<td>blue</td>
<td>285.2924</td>
<td>3.636</td>
</tr>
<tr>
<td>Sample4</td>
<td>blue</td>
<td>571.4323</td>
<td>3.785</td>
</tr>
<tr>
<td>Sample5</td>
<td>blue</td>
<td>419.3184</td>
<td>3.933</td>
</tr>
<tr>
<td>Sample6</td>
<td>blue</td>
<td>659.4875</td>
<td>4.091</td>
</tr>
<tr>
<td>Sample7</td>
<td>green</td>
<td>832.6272</td>
<td>4.255</td>
</tr>
<tr>
<td>Sample8</td>
<td>green</td>
<td>681.4981</td>
<td>4.418</td>
</tr>
<tr>
<td>Sample9</td>
<td>green</td>
<td>549.4212</td>
<td>4.575</td>
</tr>
<tr>
<td>Sample10</td>
<td>green</td>
<td>527.4065</td>
<td>4.736</td>
</tr>
<tr>
<td>Sample11</td>
<td>green</td>
<td>458.3863</td>
<td>4.893</td>
</tr>
<tr>
<td>Sample12</td>
<td>green</td>
<td>628.5179</td>
<td>5.501</td>
</tr>
<tr>
<td>Sample13</td>
<td>green</td>
<td>304.3019</td>
<td>5.565</td>
</tr>
<tr>
<td>Sample14</td>
<td>green</td>
<td>796.5588</td>
<td>5.62</td>
</tr>
<tr>
<td>Sample15</td>
<td>green</td>
<td>774.5773</td>
<td>5.686</td>
</tr>
<tr>
<td>Sample16</td>
<td>green</td>
<td>650.4938</td>
<td>5.76</td>
</tr>
</tbody>
</table>

\[ y = \text{function (x values)} \]

where \( y \) are discrete categories such as text
multiple categories (here colors) are possible

\[ \text{Category } y = \sin((40.579 \cdot \text{Value } x3 + \cos(\sin(4.2372 \cdot \text{Value } x1)) - 3.25702)/(1.43018 + \sin(\text{Value } x2))) \]

\[ \text{Category } y = \text{round}(0.1219 \cdot \text{Value } x2) \]
Figures of merit for classifications

A) Calculate prediction and true/false values

<table>
<thead>
<tr>
<th>Category y</th>
<th>Value x1</th>
<th>Value x2</th>
<th>Value x3</th>
<th>predicted</th>
<th>true/false</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample1</td>
<td>blue</td>
<td>615.4603</td>
<td>3.363</td>
<td>0.0561</td>
<td>blue</td>
</tr>
<tr>
<td>Sample2</td>
<td>blue</td>
<td>371.3181</td>
<td>3.491</td>
<td>0.0582</td>
<td>blue</td>
</tr>
<tr>
<td>Sample3</td>
<td>blue</td>
<td>285.2924</td>
<td>3.636</td>
<td>0.0606</td>
<td>blue</td>
</tr>
<tr>
<td>Sample4</td>
<td>blue</td>
<td>571.4323</td>
<td>3.785</td>
<td>0.0631</td>
<td>blue</td>
</tr>
<tr>
<td>Sample5</td>
<td>blue</td>
<td>419.3184</td>
<td>3.933</td>
<td>0.0656</td>
<td>blue</td>
</tr>
<tr>
<td>Sample6</td>
<td>blue</td>
<td>659.4875</td>
<td>4.091</td>
<td>0.0682</td>
<td>blue</td>
</tr>
<tr>
<td>Sample7</td>
<td>green</td>
<td>832.6272</td>
<td>4.255</td>
<td>0.0709</td>
<td>blue</td>
</tr>
<tr>
<td>Sample8</td>
<td>green</td>
<td>681.4981</td>
<td>4.418</td>
<td>0.0736</td>
<td>green</td>
</tr>
<tr>
<td>Sample9</td>
<td>green</td>
<td>549.4212</td>
<td>4.575</td>
<td>0.0763</td>
<td>green</td>
</tr>
<tr>
<td>Sample10</td>
<td>green</td>
<td>527.4065</td>
<td>4.736</td>
<td>0.0789</td>
<td>green</td>
</tr>
<tr>
<td>Sample11</td>
<td>green</td>
<td>458.3863</td>
<td>4.893</td>
<td>0.0816</td>
<td>green</td>
</tr>
<tr>
<td>Sample12</td>
<td>green</td>
<td>628.5179</td>
<td>5.501</td>
<td>0.0917</td>
<td>green</td>
</tr>
<tr>
<td>Sample13</td>
<td>green</td>
<td>304.3019</td>
<td>5.565</td>
<td>0.0928</td>
<td>green</td>
</tr>
<tr>
<td>Sample14</td>
<td>green</td>
<td>796.5588</td>
<td>5.62</td>
<td>0.0937</td>
<td>green</td>
</tr>
<tr>
<td>Sample15</td>
<td>green</td>
<td>774.5773</td>
<td>5.686</td>
<td>0.0948</td>
<td>green</td>
</tr>
<tr>
<td>Sample16</td>
<td>green</td>
<td>650.4938</td>
<td>5.76</td>
<td>0.0960</td>
<td>green</td>
</tr>
</tbody>
</table>

Example is special case of binary classification multiple categories are possible

B) Confusion matrix

<table>
<thead>
<tr>
<th>true positives</th>
<th>true negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>false positives</td>
<td>false negatives</td>
</tr>
</tbody>
</table>

C) Figures of merit

True positive rate or sensitivity or recall = TP/(TP+FN)
False positive rate = FP/(FP+TN)
Accuracy = (TP+TN)/(TP+TN+FP+FN)
Specificity = (TN/(FP+TN)
Precision = TP/(TP+FN)
Negative predictive value = TN/(TN+FN)
False discovery rate = FP/(FP+TP)

D) ROC curves
Supervised learning with continuous data
Regression

\[ y = \text{function (x values)} \]
where \( y \) are continuous values such as numbers

<table>
<thead>
<tr>
<th>Sample</th>
<th>Value x1</th>
<th>Value x2</th>
<th>Value x3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample1</td>
<td>12.10</td>
<td>615.4603</td>
<td>3.363</td>
</tr>
<tr>
<td>Sample2</td>
<td>14.96</td>
<td>371.3181</td>
<td>3.491</td>
</tr>
<tr>
<td>Sample3</td>
<td>13.10</td>
<td>285.2924</td>
<td>3.636</td>
</tr>
<tr>
<td>Sample4</td>
<td>15.51</td>
<td>571.4323</td>
<td>3.785</td>
</tr>
<tr>
<td>Sample5</td>
<td>14.10</td>
<td>419.3184</td>
<td>3.933</td>
</tr>
<tr>
<td>Sample6</td>
<td>14.99</td>
<td>659.4875</td>
<td>4.091</td>
</tr>
<tr>
<td>Sample7</td>
<td>15.10</td>
<td>832.6272</td>
<td>4.255</td>
</tr>
<tr>
<td>Sample8</td>
<td>25.03</td>
<td>681.4981</td>
<td>4.418</td>
</tr>
<tr>
<td>Sample9</td>
<td>16.10</td>
<td>549.4212</td>
<td>4.575</td>
</tr>
<tr>
<td>Sample10</td>
<td>16.43</td>
<td>527.4065</td>
<td>4.736</td>
</tr>
<tr>
<td>Sample11</td>
<td>17.10</td>
<td>458.3863</td>
<td>4.893</td>
</tr>
<tr>
<td>Sample12</td>
<td>24.69</td>
<td>628.5179</td>
<td>5.501</td>
</tr>
<tr>
<td>Sample13</td>
<td>18.10</td>
<td>304.3019</td>
<td>5.565</td>
</tr>
<tr>
<td>Sample14</td>
<td>27.93</td>
<td>796.5588</td>
<td>5.62</td>
</tr>
<tr>
<td>Sample15</td>
<td>19.10</td>
<td>774.5773</td>
<td>5.686</td>
</tr>
<tr>
<td>Sample16</td>
<td>23.39</td>
<td>650.4938</td>
<td>5.76</td>
</tr>
</tbody>
</table>

\[ y = \text{mod}(\text{Value x2}^2, 6.61) \times \tan(0.2688 \times \sin(5.225 \times \text{Value x2}^2)) \times \max(\text{Value x2} + -25.59/(\text{Value x2}^2) \times \cos(42.32 \times \text{Value x2}), \text{floor}(\text{mod}(\text{Value x2}^2, 6.61))) + \text{round}(4.918 \times \text{Value x2} - 3.945 - 3.672 \times \text{ceil}(\sin(5.225 \times \text{Value x2}^2))) \]

\[ R^2 \text{ Goodness of Fit} = 0.99522414 \]
\[ \text{Correlation Coefficient} = 0.99760921 \]
\[ \text{Maximum Error} = 0.60953998 \]
\[ \text{Mean Squared Error} = 0.092117594 \]
\[ \text{Mean Absolute Error} = 0.22821247 \]
Figures of merit for regressions

Figures of merit are also calculated for external test and validation sets such as the **predictive squared correlation coefficient** $Q^2$

<table>
<thead>
<tr>
<th>Figure of Merit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R^2$ Goodness of Fit</td>
<td>0.99522414</td>
</tr>
<tr>
<td>Correlation Coefficient</td>
<td>0.99760921</td>
</tr>
<tr>
<td>Maximum Error</td>
<td>0.60953998</td>
</tr>
<tr>
<td>Mean Squared Error</td>
<td>0.092117594</td>
</tr>
<tr>
<td>Mean Absolute Error</td>
<td>0.22821247</td>
</tr>
</tbody>
</table>
Overfitting – trust but verify

Old model applied to new data
$R^2 = 0.995 \rightarrow Q^2 = 0.7227$

External validation failed
**Prediction power is most important**
Overfitting – avoid the unexpected

Dogs

Cats

New Kid on the block
Avoid overfitting

Internal cross-validation (weak)

70/30 split development/test set (good)

External validation set or blind hold-out (best)

TOP:
Combine all three methods

n-fold CV

Training (70%)
Test (30%)

Training (70%)
Test (30%)
External validation set (+30%)

Training (70%) with CV
Test (30%)
External validation set (+30%)
Sample selection for testing and validation

Sample selection for test and validation set split should be truly randomized

Range of the y-coordinate (activity or response) should be completely covered

Training and test set variables should not overlap
Why do we need feature selection?

• Reduces computational complexity
• Curse of dimensionality is avoided
• Improves accuracy
• The selected features can provide insights about the nature of the problem*

* Margin Based Feature Selection Theory and Algorithms; Amir Navot
Feature selection example

Principal component analysis (PCA) microarray data

NO feature selection $\rightarrow$ no separation

With feature selection $\rightarrow$ separation

Data: Golub, 1999 Science Mag
Approach: Automated substructure detection

**Aim1:** take unknown mass spectrum – predict all substructures
**Aim2:** classification into common compound classes (sugar, amino acid, sterol)

**Pioneers:** Dendral project at Stanford University in the 1970s
Varmuza at University of Vienna
Steve Stein at NIST
MOLGEN-MS team at University Bayreuth
Principle of mass spectral features

Mass Spectral Features
- m/z value
- m/z intensity
- delta (m/z)
- delta (m/z) x intensity
- non linear functions
- intensity series

MS Feature matrix

<table>
<thead>
<tr>
<th></th>
<th>f1</th>
<th>f2</th>
<th>f3</th>
<th>f4</th>
<th>f5</th>
<th>fn</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS1</td>
<td>100</td>
<td>20</td>
<td>50</td>
<td>60</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MS2</td>
<td>100</td>
<td>20</td>
<td>50</td>
<td>60</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>MS3</td>
<td>100</td>
<td>20</td>
<td>60</td>
<td>50</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MS4</td>
<td>0</td>
<td>40</td>
<td>20</td>
<td>50</td>
<td>0</td>
<td>40</td>
</tr>
<tr>
<td>MS5</td>
<td>0</td>
<td>40</td>
<td>20</td>
<td>50</td>
<td>0</td>
<td>40</td>
</tr>
</tbody>
</table>

Substructure matrix

<table>
<thead>
<tr>
<th></th>
<th>s1</th>
<th>s2</th>
<th>s3</th>
<th>s4</th>
<th>s5</th>
<th>sn</th>
</tr>
</thead>
<tbody>
<tr>
<td>Molecule1</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Molecule2</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Molecule3</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Molecule4</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Molecule5</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
Application - Substructure detection and prediction

**Generalized Linear Models (GLM)**
- General Discriminant Analysis
- Binary logit (logistic) regression
- Binary probit regression

**Nonlinear models**
- Multivariate adaptive regression splines (MARS)

**Tree models**
- Standard Classification Trees (CART)
- Standard General Chi-square Automatic Interaction Detector (CHAID)
- Exhaustive CHAID
- Boosting classification trees
- M5 regression trees

**Meta Learning**

**Neural Networks**
- Multilayer Perceptron
- Neural network (MLP)
- Radial Basis Function neural network (RBF)

**Machine Learning**
- Support Vector Machines (SVM)
- Naive Bayes classifier
- k-Nearest Neighbors (KNN)
Strategy - let all machine learning algorithms compete

Lower is better
Application: Retention time prediction for liquid chromatography

Calibration using logP concept for reversed phase liquid chromatography data

- very simplistic and coarse filter for RP only
- problematic with multi ionizable compounds
- logD (includes pKa) better than logP
- possible use as time segment filter
Application: Retention time prediction for liquid chromatography

- Based on logD, pKa, logP and Kier & Hall atomic descriptors;
- 90 compounds; \((n_{\text{dev}} = 48, n_{\text{test}} = 32)\); Std error 3.7 min
- Good models need development set \(n>500\), needs to be highly diverse
- Prediction power is most important

QSRR Model: Tobias Kind (FiehnLab) using ChamAxon Marvin and WEKA
Application: Decision tree supported substructure prediction of metabolites from GC-MS profiles

Hummel J, Strehmel N, Selbig J, Walther D, Kopka J.

Decision tree supported substructure prediction of metabolites from GC-MS profiles.

Spectrum  Decision tree  Compound structure
Toxicity and carcinogenicity predictions with ToxTree

Conclusions – Machine Learning

**Classification** (categorical data) and **regression** (continuous data) for prediction of future values

Let algorithms **compete** for best solution (voting, boosting, bagging)

**Validation** (trust but verify) is the cornerstone of machine learning to avoid false results and wishful thinking

Modern algorithms do not necessarily provide direct causal insight they rather provide the best statistical solution or equation

**Domain knowledge** of the learning problem is important and **helpful for artifact removal** and final interpretation

**Prediction power** is most important
Thank you!