CAP 5510: Introduction to Bioinformatics
CGS 5166: Bioinformatics Tools
Giri NARASIMHAN

www.cis.fiu.edu/~giri/teach/BioinfF18.html
Machine Learning
Machine Learning

- Human Endeavor
  - Data → Information → Knowledge
- Machine Learning
  - Automatically extracting information from data
- Types of Machine Learning
  - Unsupervised
    - Clustering
    - Pattern Discovery
  - Supervised
    - Learning
    - Classification
Support Vector Machines

- **Supervised Statistical Learning Method for:**
  - Classification
  - Regression

- **Simplest Version:**
  - **Training:** Present series of labeled examples (e.g., gene expressions of tumor vs. normal cells)
  - **Validation:** Step to fine-tune hyperparameters
  - **Prediction:** Predict labels of new examples.
Learning Problems
Learning Problems

- Binary Classification
- Multi-class classification
- Regression
SVM – Binary Classification

- Partition feature space with a surface.
- Surface is implied by a subset of the training points (vectors) near it. These vectors are referred to as Support Vectors.
- Efficient with high-dimensional data.
- Solid statistical theory
- Subsume several other methods.
Classification of 2-D (Separable) data
Classification of 2-D (Separable) data
Classification of (Separable) 2-D data

- Margin of a point
- Margin of a point set
Classification using the Separator

\[ w \cdot x_i + b > 0 \]

\[ w \cdot x_j + b < 0 \]

\[ w \cdot x + b = 0 \]
Perceptron Algorithm (Primal)

Rosenblatt, 1956

Given separable training set $S$ and learning rate $\eta > 0$

$w_0 = 0$; // Weight

$b_0 = 0$; // Bias

$R = \max |x_i|

\text{repeat}

k = 0;

\text{for } i = 1 \text{ to } N

\text{if } y_i (w_k \cdot x_i + b_k) \leq 0 \text{ then}

w_{k+1} = w_k + \eta y_i x_i

b_{k+1} = b_k + \eta y_i R^2

k = k + 1

\text{Until no mistakes made within loop}

\text{Return } k, \text{ and } (w_k, b_k) \text{ where } k = \# \text{ of mistakes}
Performance for Separable Data

Theorem:

If margin $m$ of $S$ is positive, then

$$k \leq (2R/m)^2$$

i.e., the algorithm will always converge, and will converge quickly.
Non-linear Separators
Main idea: Map into feature space

Figure 2. The idea of SV machines: map the training data nonlinearly into a higher-dimensional feature space via $\phi$, and construct a separating hyperplane with maximum margin there. This yields a nonlinear decision boundary in input space. By the use of a kernel function, it is possible to compute the separating hyperplane without explicitly carrying out the map into the feature space.
Non-linear Separators
Useful URLs

http://www.support-vector.net
Perceptron Algorithm (Primal)

Rosenblatt, 1956

Given separable training set S and learning rate $\eta > 0$

$w_0 = 0$; // Weight

$b_0 = 0$; // Bias

$R = \max |x_i|$

repeat

\[ k = 0; \]

for $i = 1$ to $N$

if $y_i(w_k \cdot x_i + b_k) \leq 0$ then

\[ w_{k+1} = w_k + \eta y_i x_i \]

\[ b_{k+1} = b_k + \eta y_i R^2 \]

$k = k + 1$

Until no mistakes made within loop

Return $k$, and $(w_k, b_k)$ where $k =$ # of mistakes

$w = \Sigma \eta a_i y_i x_i$
Perceptron Algorithm (Dual)

Given a separable training set $S$

$a = 0; b_0 = 0$

$R = \max |x_i|$

repeat
  for $i = 1$ to $N$
    if $y_i \left( \sum \eta a_j y_j x_i \cdot x_j + b \right) \leq 0$ then
      \[ a_i = a_i + 1 \]
      \[ b = b + y_i R^2 \]
    endif
  Until no mistakes made within inner for-loop
Return $(a, b)$
Perceptron Algorithm (Dual)

Given a separable training set $S$

$a = 0; b_0 = 0; \quad R = \max |x_i|$

repeat

for $i = 1$ to $N$

if $y_i \left( \sum a_j y_j \Phi'(x_i, x_j) + b \right) \leq 0$ then

$a_i = a_i + 1$

$b = b + y_i R^2$

Until no mistakes made within loop

Return $(a, b)$

$\Phi'(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$
Different Kernel Functions

- **Polynomial kernel**
  \[ \kappa(X, Y) = (X \cdot Y)^d \]

- **Radial Basis Kernel**
  \[ \kappa(X, Y) = \exp\left(\frac{-\|X - Y\|^2}{2\sigma^2}\right) \]

- **Sigmoid Kernel**
  \[ \kappa(X, Y) = \tanh(\omega (X \cdot Y) + \theta) \]
SVM Ingredients

- Support Vectors
- Mapping from Input Space to Feature Space
- Dot Product – Kernel function
- Weights
Generalizations

- How to deal with more than 2 classes?
  **Idea:** Associate weight and bias for each class.

- How to deal with non-linear separator?
  **Idea:** Support Vector Machines.

- How to deal with linear regression?

- How to deal with non-separable data?
Applications

- **Text Categorization & Information Filtering**
  - 12,902 Reuters Stories, 118 categories (91% !)

- **Image Recognition**
  - Face Detection, tumor anomalies, defective parts in assembly line, etc.

- **Gene Expression Analysis**

- **Protein Homology Detection**
<table>
<thead>
<tr>
<th>Class</th>
<th>Method</th>
<th>T</th>
<th>F</th>
<th>N</th>
<th>P</th>
<th>T</th>
<th>N</th>
<th>Cost</th>
<th>T</th>
<th>F</th>
<th>N</th>
<th>P</th>
<th>T</th>
<th>N</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>tumor</td>
<td>Tumor SVM</td>
<td>0</td>
<td>2</td>
<td>7</td>
<td>2045</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discounted SVM</td>
<td>6</td>
<td>5</td>
<td>14</td>
<td>2059</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discounted 2 SVM</td>
<td>6</td>
<td>5</td>
<td>14</td>
<td>2059</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discounted 3 SVM</td>
<td>6</td>
<td>5</td>
<td>14</td>
<td>2059</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Parzen</td>
<td>7</td>
<td>3</td>
<td>9</td>
<td>2044</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2044</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FDD</td>
<td>6</td>
<td>5</td>
<td>14</td>
<td>2059</td>
<td>10</td>
<td>1</td>
<td>2</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ck-5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2048</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2048</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MCG</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>2046</td>
<td>12</td>
<td>1</td>
<td>1</td>
<td>2045</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Comparison of error rates for various classification methods (continued). See caption for Table 2.

Table 2: Comparison of error rates for various classification methods. Classes were determined in Table 1. The methods are the unified basis function SVM, the SVMs using the unified decision function, a decision function, and the two decision function, FDD, Ck-5, and MCG. The most frequent errors are false positives, false negatives, true positives, and true negatives and are counted over three cross-validation splits, followed by the most frequent errors over the number of false positives, false negatives, true positives, and true negatives. These frequencies are repeated three times using the training data from the training set, and their using the threshold that maximizes the cost on the test set. The threshold optimization is not possible for the decision tree method, since they do not produce optimal results.
Table 4: Comparison of SVM performance using various kernels. For each of the two class functions, SVMs were trained using five different kernel functions on five different random three-fold splits of the data, training on those data and leaving the new group out. The test error measurements are indicated in Table 1. The second column indicates the kernel function, as described in Table 1. The third column indicates the kernel function, as described in Table 2. The last two columns contain the number of true positives plus the number of false negatives for each of the two random three-fold splits. The last column is the total error percentage.

<table>
<thead>
<tr>
<th>Gene</th>
<th>Local</th>
<th>Error</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCA</td>
<td>YPR007W</td>
<td>CFT</td>
<td>FN</td>
</tr>
<tr>
<td>YFL042W</td>
<td>LBC1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YKL350C</td>
<td>LTH1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YLR374W</td>
<td>HDA1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YLR165W</td>
<td>KOD1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YLR164C</td>
<td>KOD2</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YIL006W</td>
<td>LTH1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YLR150S</td>
<td>ALE1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
<tr>
<td>YLR010S</td>
<td>CGL1</td>
<td>FN</td>
<td>rate-limited respiratory chain</td>
</tr>
</tbody>
</table>

Table 4a: Consistently misclassified genes. The table lists all 55 genes that were consistently misclassified by SVMs trained using the M2GO class functions listed in Table 1. The type of errors is indicated across the row(s) on which the SVM misclassifies the gene in the given class but the M2GO classification does not have a false-negative (FN) row when the SVM does not misclassify the gene in the given class but the M2GO class functions do.
Table 1: Results for the perceptron on all data sets. The results are averaged over 5 shufflings of the data set. This algorithm is sensitive to the order in which it receives the data points. The first column is the dataset used and the second is the number of features in the dataset. For the ovarian and colon datasets, the number of cancerous tissues misclassified (FP) and the number of normal tissues misclassified (FN) is reported. For the AML/ALL training dataset, the number of AML samples misclassified (FP) and the number of ALL patients misclassified (FN) is reported. For the AML treatment dataset, the number of unsuccessfully treated patients misclassified (FP) and the number of successfully treated patients misclassified (FN) is reported. The last two columns report the best score obtained by the SVM on that dataset.

Table 2: Error rates for ovarian cancer tissue experiments.

For each setting of the SVM, consisting of a linear and diagonal linear (DP) set, three were examined. Column 2 lists the number of features (classes) used. The next two columns list the number of cancerous tissues misclassified (FP), normal tissues misclassified (FN), cancerous samples correctly classified (TP), and normal tissues classified correctly (TN).