Instructor: Sael Lee
CS549 Spring – Computational Biology

Lecture 14: Biomarker Discovery with feature selection methods

Resources:
Robust biomarker identification for cancer diagnosis with ensemble feature selection methods

Thomas Abeel¹,², Thibault Helleputte³,⁴, Yves Van de Peer¹,², Pierre Dupont³,⁴ and Yvan Saeys¹,²,*

¹Department of Plant Systems Biology, VIB, Technologieweg 927, 9052 Gent, ²Department of Molecular Genetics, Ghent University, Gent, ³Department of Computing Science and Engineering INGI and ⁴Machine Learning Group, Université catholique de Louvain, Louvain, Belgium

Received on May 5, 2009; revised on October 26, 2009; accepted on November 2, 2009
Advance Access publication November 25, 2009
Associate Editor: Thomas Lengauer
Abstract

Motivation:
“Biomarker discovery is an important topic in biomedical applications of computational biology, including applications such as gene and SNP selection from high-dimensional data. Surprisingly, the stability with respect to sampling variation or robustness of such selection processes has received attention only recently. ...”
Results:
“We show that the robustness of SVMs for biomarker discovery can be substantially increased by using ensemble feature selection techniques, while at the same time improving upon classification performances. The proposed methodology is evaluated on four microarray datasets showing increases of up to almost 30% in robustness of the selected biomarkers, along with an improvement of ~15% in classification performance. The stability improvement with ensemble methods is particularly noticeable for small signature sizes (a few tens of genes), which is most relevant for the design of a diagnosis or prognosis model from a gene signature. ...”
### Microarray Datasets

#### Types of cancer

<table>
<thead>
<tr>
<th>Name</th>
<th>References</th>
<th># samples (+/-)</th>
<th># dim.</th>
<th>SDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leukemia</td>
<td>Golub et al. (1999)</td>
<td>72 (47/25)</td>
<td>7,129</td>
<td>0.010</td>
</tr>
<tr>
<td>Colon</td>
<td>Alon et al. (1999)</td>
<td>62 (40/22)</td>
<td>2,000</td>
<td>0.031</td>
</tr>
<tr>
<td>Lymphoma</td>
<td>Alizadeh et al. (2000)</td>
<td>45 (22/23)</td>
<td>4,026</td>
<td>0.011</td>
</tr>
<tr>
<td>Prostate</td>
<td>Singh et al. (2002)</td>
<td>102 (52/50)</td>
<td>6,033</td>
<td>0.017</td>
</tr>
</tbody>
</table>

SDR refers to the ratio between the number of samples and the number of dimensions (or features).

**Colon Cancer dataset:** is made of samples from 40 tumor and 22 normal colon tissues measuring more than 6500 genes.

**Leukemia dataset:** model to discriminate between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) tissues.
Microarray Datasets

Types of cancer

<table>
<thead>
<tr>
<th>Name</th>
<th>References</th>
<th># samples (+/−)</th>
<th># dim.</th>
<th>SDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leukemia</td>
<td>Golub et al. (1999)</td>
<td>72 (47/25)</td>
<td>7,129</td>
<td>0.010</td>
</tr>
<tr>
<td>Colon</td>
<td>Alon et al. (1999)</td>
<td>62 (40/22)</td>
<td>2,000</td>
<td>0.031</td>
</tr>
<tr>
<td>Lymphoma</td>
<td>Alizadeh et al. (2000)</td>
<td>45 (22/23)</td>
<td>4,026</td>
<td>0.011</td>
</tr>
<tr>
<td>Prostate</td>
<td>Singh et al. (2002)</td>
<td>102 (52/50)</td>
<td>6,033</td>
<td>0.017</td>
</tr>
</tbody>
</table>

SDR refers to the ratio between the number of samples and the number of dimensions (or features).

The lymphoma dataset: comes from a study on diffuse large B-cell Lymphoma in discriminate between two types of lymphoma based on gene expression.

The prostate dataset: was first published in One of the tasks addressed by the authors is to build a model able to discriminate between normal and tumor prostate tissue.
Microarray Expression Normalization

The objective of data normalization is to enhance the similarity of genes sharing a common expression pattern throughout the data, but in different ranges of absolute expression values.

IQR-normalization:
The normalized expression value $\bar{f}_{ij}$ is defined as follows.

$$\bar{f}_{ij} = \frac{f_{ij} - m_j}{IQR_j/1.35}$$

where $f_{ij}$ is the original expression value of gene $j$ from sample $i$, $m_j$ is the median of expression of this gene over all samples and $IQR_j$ stands for the gene-specific interquartile range.

Microarray Expression Normalization

\[
\tilde{f}_{ij} = \frac{f_{ij} - m_j}{IQR_j / 1.35}
\]

The IQR-normalization is more robust to the presence of outliers than a classical Z-score (centering to the mean with unit SD), but the 1.35 scaling factor makes both normalization equivalent whenever the data happens to be normally distributed.

* The normalization parameters for each gene are always estimated from the training samples only and applied subsequently to the validation samples

Stability Evaluation

Stability Concept: adding or deleting a few samples should not drastically modify the top-ranked markers identified by the algorithm.

slight variations of the original dataset, and compare the outcome of the marker selection algorithm across these different variations.

Variations: subsampling the original dataset without replacement containing 90% of the samples of the original dataset.
Stability Evaluation

Stability analysis process:
• Dataset $X = \{x_1, \ldots, x_M\}$ with $M$ instances and $N$ features. Then, $k$ subsamplings of size $xM$ ($0 < x < 1$) are drawn randomly from $X$, where in our experiments $k=500$ and $x=0.9$.

• Feature selection is performed on each of the $k$ subsamplings, and a marker set—further referred to as a signature—of a given size is selected.

• Similarity of the signatures of the $k$ subsamples are evaluated for stability.
**Stability Measure**

The more similar all signatures are, the higher the stability measure will be.

**Stability**: Defined as the average over all pairwise similarity comparisons between all signatures on the $k$ subsamplings.

$$S_{tot} = \frac{2 \sum_{i=1}^{k} \sum_{j=i+1}^{k} \text{KL}(f_i, f_j)}{k(k-1)}$$

$f_i$: the signature obtained by the selection method on subsampling $i$ ($1 \leq i \leq k$),

Stability Measure

Kuncheva Index which is a stability index between $f_i$ and $f_j$

$$s = |f_i| = |f_j|$$ : signature size

$$r = |f_i \cap f_j|$$ : number of common elements in both signatures

$$KI(f_i, f_j) = \frac{r \cdot N - s^2}{s \cdot (N - s)} = \frac{r - (s^2/N)}{s - (s^2/N)}$$

bias correction term: selecting common features at random

$-1 < KI(f_i, f_j) \leq 1$ and the greater its value, the larger the number of commonly selected features in both signatures

Classification Evaluation

Use the same subsamplings—each containing 90% of the original dataset—as training sets to select features and estimate the performance of a classifier. The remaining 10% of the data can be used each time as an independent validation set to evaluate classification performance.

Area under the curve of receiver operator curve (AUC ROC):
Statistical measures of the performance of a binary classification test
## True disease state vs. Test result

<table>
<thead>
<tr>
<th>Disease (D)</th>
<th>Test</th>
<th>Null H. not rejected (Negative test outcome)</th>
<th>Null H. rejected (Positive test outcome)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No disease (D = 0)</td>
<td>![Smiley Face] (Specificity (TN rate))</td>
<td>![X] (Type I error (FP rate) α)</td>
<td></td>
</tr>
<tr>
<td>Disease (D = 1)</td>
<td>![X] (Type II error (FN rate) β)</td>
<td>![Smiley Face] (Power 1 - β; Sensitivity (TP rate; recall))</td>
<td></td>
</tr>
</tbody>
</table>

**Precision** = TP/(TP+FP)

**Accuracy** = (TP + TN)/(TP+TN+FP+FN)
## Confusion Matrix

<table>
<thead>
<tr>
<th>True condition</th>
<th>Total population</th>
<th>Condition positive</th>
<th>Condition negative</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prevalence</strong></td>
<td>$\Sigma$ Condition positive/$\Sigma$ Total population</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Accuracy (ACC)</strong></td>
<td>$\Sigma$ True positive + $\Sigma$ True negative/$\Sigma$ Total population</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Positive predictive value (PPV), Precision</strong></td>
<td>$\Sigma$ True positive/$\Sigma$ Test outcome positive</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>False discovery rate (FDR)</strong></td>
<td>$\Sigma$ False positive/$\Sigma$ Test outcome positive</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>False omission rate (FOR)</strong></td>
<td>$\Sigma$ False negative/$\Sigma$ Test outcome negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Negative predictive value (NPV)</strong></td>
<td>$\Sigma$ True negative/$\Sigma$ Test outcome negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>True positive rate (TPR), Sensitivity, Recall</strong></td>
<td>$\Sigma$ True positive/$\Sigma$ Condition positive</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>False positive rate (FPR), Fall-out</strong></td>
<td>$\Sigma$ False positive/$\Sigma$ Condition negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Positive likelihood ratio (LR+)</strong></td>
<td>$\text{TPR/FPR}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Diagnostic odds ratio (DOR)</strong></td>
<td>$\text{LR+}/\text{LR−}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>True negative rate (TNR), Specificity (SPC)</strong></td>
<td>$\Sigma$ True negative/$\Sigma$ Condition negative</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Negative likelihood ratio (LR−)</strong></td>
<td>$\text{FNR/TN}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Predicted condition
- **Predicted condition positive**
  - **True positive**
  - **False positive** (Type I error)
- **Predicted condition negative**
  - **False negative** (Type II error)
  - **True negative**

Specific Example

Pts without the disease

Pts with disease

Test Result
Threshold

Call these patients “negative”

Call these patients “positive”

Test Result
Some definitions ...
Call these patients “negative”

without the disease

Call these patients “positive”

with the disease

Test Result

False
Positives
Call these patients “negative”

Call these patients “positive”

Test Result

without the disease

with the disease

True negatives
Call these patients “negative”

Call these patients “positive”

False negatives

Test Result

without the disease

with the disease
Moving the Threshold: right

without the disease
with the disease
Moving the Threshold: left

without the disease
with the disease
ROC curve

True Positive Rate (sensitivity)

False Positive Rate (1-specificity)

100%

0%
ROC curve comparison

A good test:

A poor test:
ROC curve extremes

Best Test:

The distributions don’t overlap at all

Worst test:

The distributions overlap completely
Area under ROC curve (AUC)

- Overall measure of test performance
- Comparisons between two tests based on differences between (estimated) AUC
- For continuous data, AUC equivalent to Mann-Whitney U-statistic (nonparametric test of difference in location between two populations)
AUC for ROC curves

- **AUC = 100%**

- **AUC = 50%**

- **AUC = 90%**

- **AUC = 65%**
Interpretation of AUC

“AUC can be interpreted as the probability that the test result from a randomly chosen diseased individual is more indicative of disease than that from a randomly chosen nondiseased individual”: 

\[ P(X_i \geq X_j | D_i = 1, D_j = 0) \]

So can think of this as a nonparametric distance between disease/nondisease test results

From “Statistics in the 21st Century”
Problems with AUC

- No clinically relevant meaning

- A lot of the area is coming from the range of large false positive values, no one cares what’s going on in that region (need to examine restricted regions)

- The curves might cross, so that there might be a meaningful difference in performance that is not picked up by AUC
Embedded Feature selection with SVM

Recursive Feature Elimination: a type of backward feature elimination

1. Train the classifier (optimize the weights $w_i$ with respect to error function $J$).
2. Compute the ranking criterion $((w_i)^2)$ for all features.
3. Remove the feature with smallest ranking criterion.

- If features are removed one at a time, there is also a corresponding feature ranking.
- However, the features that are top ranked (eliminated last) are not necessarily the ones that are individually most relevant. Only taken together the features of a subset $F_m$ are optimal in some sense.

Embedded Feature selection with SVM

A linear SVM essentially consists of a separating hyperplane in the input space. -> The absolute values of the weights of each dimension in the hyperplane can be regarded as the contribution (importance) of each dimension (feature) to the multivariate decision of the hyperplane.

Feature ranking with Support Vector Machine- Recursive Feature Elimination

**Algorithm SVM-train**

Inputs: Training examples \( \{x_1, x_2, \ldots, x_k, \ldots, x_\ell\} \) and class labels \( \{y_1, y_2, \ldots, y_k, \ldots, y_\ell\} \).

Minimize over \( \alpha_k \):

\[
J = \frac{1}{2} \sum_{hk} y_h y_k \alpha_h \alpha_k (x_h \cdot x_k + \lambda \delta_{hk}) - \sum_k \alpha_k
\]

subject to:

\[
0 \leq \alpha_k \leq C \quad \text{and} \quad \sum_k \alpha_k y_k = 0
\]

Outputs: Parameters \( \alpha_k \).

\[
D(x) = w \cdot x + b
\]

\[
w = \sum \alpha_k y_k x_k \quad \text{and} \quad b = \langle y_k - w \cdot x_k \rangle
\]
Feature ranking with Support Vector Machine - Recursive Feature Elimination

**Algorithm SVM-RFE:**

**Inputs:**
Training examples
\[ X_0 = [x_1, x_2, \ldots x_k, \ldots x_\ell]^T \]
Class labels
\[ y = [y_1, y_2, \ldots y_k, \ldots y_\ell]^T \]
Initialize:
Subset of surviving features
\[ s = [1, 2, \ldots n] \]
Feature ranked list
\[ r = [] \]
Repeat until \( s = [] \)

Restrict training examples to good feature indices
\[ X = X_0(:, s) \]
Train the classifier
\[ \alpha = SVM\text{-}train(X, y) \]
Compute the weight vector of dimension length(s)
\[ w = \sum_k \alpha_k y_k x_k \]
Compute the ranking criteria
\[ c_i = (w_i)^2, \quad \text{for all } i \]
Find the feature with smallest ranking criterion
\[ f = \text{argmin}(c) \]
Update feature ranked list
\[ r = [s(f), r] \]
Eliminate the feature with smallest ranking criterion
\[ s = s(1:f - 1, f + 1:length(s)) \]
Output:
Feature ranked list \( r \).
**Ensemble Feature Selection**

**Idea:** Aggregate the feature rankings provided by the single feature selectors into a final consensus ranking.

Consider an ensemble $E$ consisting of $s$ feature selectors, $E = \{F_1, F_2, \ldots, F_s\}$, Assuming that each $F_i$ provides a feature ranking

$$f_i = (f_i^1, \ldots, f_i^N),$$

which are aggregated into a consensus feature ranking $f$ by weighted voting:

Ensemble feature selection

weighted voting:

\[
f^l = \sum_{i=1}^{t} w(f_i^l)
\]

where \(w(.)\) denotes a weighting function. If a linear aggregation is performed using \(w(f_i^l) = f_i^l\), this results in a sum where features contribute in a linear way with respect to their rank.

- Weights can be used to incorporate prior knowledge.

Ensemble of linear SVM and SVM-RFE feature selection methods

Ensemble feature selection applied by Abeel et al.

Starting from a particular training set, i.e. one of the 500 subsamplings containing 90% of the data,

- Generate a diverse set of RFE feature selections.
  - -> Because the RFE procedure is deterministic, this is done by generating different sample sets using the particular training set.
  - -> random sampling with replacement from the the particular training
Ensemble feature selection applied by Abeel et al.

Ensemble EFS consisting of \( t \) feature selectors, 
\[
EFS = \{ F_1, F_2, \ldots, F_t \},
\]
then we assume each \( F_i \) provides a feature ranking 
\[
f_i = (f_i^1, \ldots, f_i^N),
\]
where \( f_i^j \) denotes the rank of feature \( j \) in bootstrap \( i \).

A general formulation for the ensemble ranking \( f \), obtained by summing the ranks over all bootstrap samples is as follows:

\[
f = \left( \sum_{i=1}^{t} w_i(f_i^1), \ldots, \sum_{i=1}^{t} w_i(f_i^t) \right)
\]
Results

Fig. 1. Stability of the baseline method (original RFE) and the ensemble methods for prostate. We used 40 bootstraps and RFE with $E = 20\%$.

Fig. 2. Classification performances of the baseline method (original RFE) and the ensemble methods for prostate. We used 40 bootstraps and RFE with $E = 20\%$. 
Results: Changing numbers of bootstrap

Fig. 4. Stability for several numbers of bootstrap rounds for the construction of an ensemble signature for prostate. We used the CLA aggregation method and eliminated 20% of the features at each iteration of RFE. The baseline is the original RFE on the full training sets without bootstrap.

Fig. 5. Classification performances for several numbers of bootstrap rounds for the construction of an ensemble signature for prostate. We used the CLA aggregation method and eliminated 20% of the features at each iteration of RFE. The baseline is the original RFE on the full training sets without bootstrap.
Results: varying number of features to eliminate during RFE

**Fig. 6.** Stability with regard to a varying number of features to eliminate during RFE. Results represent the CLA aggregation and constructed using 40 bootstrap samples.

**Fig. 7.** Classification performance with regard to a varying number of features to eliminate during RFE. Results represent the CLA aggregation method and were constructed using 40 bootstrap samples.