Sparse Classification Methods for High Dimensional Data

Panos M. Pardalos

Distinguished Professor
Center For Applied Optimization
Industrial and Systems Engineering, University of Florida, USA.

Leading Scientist
National Research University Higher School of Economics
Laboratory of Algorithms and Technologies for Network Analysis (LATNA), Russia

http://www.ise.ufl.edu/pardalos/

December 7, 2012
Massive amounts of high-throughput data can be collected simultaneously due to technological advances.

Each observation is characterized with thousands of features (\( p \)).

- MRI and FMRI images
- Gene-expression microarrays
- Spectroscopic studies
- Web documents

Expensive measurement costs limit the size (\( n \)) of most datasets to tens or low hundreds.

**High Dimension Low Sample Sizes (HDLSS) -** \( p >> n \).
Classification is a **supervised** machine learning technique that maps some combination of input variables into pre-defined classes.

Classification models estimate a **decision rule** from training data that helps to predict the class of an unknown sample.

Classification problems appear in several applications:

- Discrimination of cancer cells from non-cancer cells
- Web-document classification
- Categorization of images in Remote-Sensing applications

Several classification methods exist in literature like,

- Support Vector Machines
- Neural Networks
- Logistic Regression
- Linear Discriminant Analysis
- Random Forests
- Adaboost
Classification on HDLSS datasets

- The high-dimensional data poses significant challenges to standard classification methods:
  - Poor generalization ability - *curse of dimensionality*
  - Geometric distortion - *equidistant points*
  - Unreliable parameter estimation - *class covariance*

Motivation & Significance

- Poor performance of standard classification methods.
- Continued technological advances.
- **Biomarker-type** information in biomedical applications.

**Scalable** and **efficient** classification models with good **generalization ability** along with **model interpretability** for high dimensional data problems.
Dimensionality Reduction

- The dimensionality reduction techniques decrease the complexity of the classification model and thus improve the classification performance.

- Dimensionality reduction techniques can be categorized as:
  - **Feature Extraction**
    - Transform the input data into a set of *meta*-features that extract relevant information from the input data for classification.
    - Limited model interpretability.
  - **Feature Selection**
    - Select a subset of features based on some *optimality* criteria
    - Advantage of model interpretability by a domain expert.
    - *Biomarker-type* information in biomedical applications.
    - Combinatorial optimization.
Feature Selection can be broadly classified as:

- **Filter methods**
- **Wrapper methods**
- **Embedded methods**

Filter Methods

- Feature subsets are ranked using a *feature relevance* score and low-ranking features are removed.
- Filter methods are independent of the classification method.
- Filter methods can be broadly categorized as:
  - **Univariate techniques**
    - Computationally efficient
    - Scalability
    - Ignore feature dependencies
  - **Multivariate techniques**
    - Feature dependencies
    - **NP-hard** problem
    - Higher computational complexity
    - Prone to over-fitting
Wrapper Methods

- Wrapper methods integrate the classifier hypothesis search within the feature subset search.
- A search procedure is defined in the feature space to select subsets of features.
- A specific feature subset is evaluated by training and testing a specific classification model.

Advantages:
- Feature dependencies
- Interaction between feature subset selection and model selection

Disadvantages:
- Over-fitting
- Computationally intensive
Embedded methods also integrate the classifier hypothesis search within the feature subset search.

Feature selection is part of model building and is generally achieved by regularization techniques.

Specific to a classification model.

Selects common subset of features for all classes. - global sparsity
Current Research

- **Sparse Proximal Support Vector Machines (sPSVMs)**
- **Fisher-based Feature Selection Combined with Support Vector Machines to Characterize Breast Cell Lines using Raman Spectroscopy.**
Fisher-based Feature Selection Combined with Support Vector Machines to Characterize Breast Cell Lines using Raman Spectroscopy
Introduction - Cancer

- **Cancer** remains one of the leading causes of death throughout the world.
- **Breast cancer** is the most common type of cancer in women, excluding skin cancers.
- In 2009, approximately 40,107 women died from breast cancer, and over 250,000 new cases were diagnosed.
- Lack of cell and tumor specific treatments - personalized medicine.
- **Classify** and **characterize** cell types for the selection of therapies for use in-vivo.
- Extract **biomarker-type** information that contribute to the differences between cell-types.
Raman Spectroscopy has demonstrated the potential to significantly aid in the research, diagnosis and treatment of various cancers.

Raman spectroscopic analysis of biological specimens provides a spectral fingerprint rich in molecular compositional information without disrupting the biological environment.

Research Objective

Construct a classification framework that would combine feature selection and classification to characterize Breast cell lines using Raman Spectroscopy.
Data Collection

- Raman spectra of five breast cell lines **MCF7, BT474, MDA-MB-231** (cancer cell lines) and **MCF10A, MCF12A** (non-cancer cell lines) are collected by Renishaw 2000 InVia Spectrometer System coupled to a Leica Microscope.

- **25-40** spectra \( n \) were collected from each cell line.

- Apparent outliers were removed by visual inspection.
Data Preprocessing

- X-axis standardization
- Savitsky-Golay Smoothing
- Background Subtraction
- Normalization

Each spectrum is characterized by 1200 measurements \((p)\) between wavenumbers 601 cm\(^{-1}\) and 1800 cm\(^{-1}\). **Raman spectral datasets** \((p \gg n)\) can be characterized as **HDLSS datasets**.
Several comparative studies have been performed on univariate and multivariate filter techniques for gene expression datasets.

Surprisingly, it has been shown that the univariate selection techniques yield consistently better results than multivariate techniques.

The differences are attributed to the difficulty in extracting the feature dependencies from limited sample sizes.

In a Raman spectrum, most biologically relevant molecular species correspond to the peaks.

A univariate filter-based technique based on Fisher Criterion called Fisher-based Feature Selection (FFS) is developed and involves the following stages:

- Peak finding
- Peak coalescing
- Feature ranking
The set of peaks $S$ for a specific cell line are defined as local maxima given by:

$$S = \{x^* | f(x^*) \geq f(x) \quad \forall x \in \mathcal{N}_\epsilon(x^*)\},$$

where $x^*$ represents the peak location, $f(x^*)$ is the corresponding intensity value of the average spectrum and $\mathcal{N}_\epsilon(x^*)$ represents an $\epsilon$-neighborhood around $x^*$. 

![Graph showing average MCF10A cell spectrum with peaks annotated.](image)
FFS - Peak Coalescing

- The number of clusters $N_C$ is defined as:

$$N_C = \arg\min_c \sum_c \sum_{i=1}^c \sum_{x_j \in C_i} (x_j - \mu_i)^2 \quad i = 1, 2, \ldots, c$$  (2)

$C_i$ represents the cluster $i$, $\mu_i$ is the mean of cluster $i$, $x_j$ is the peak $j$ assigned to cluster $i$. 

![Graph showing wavenumber vs intensity with peaks indicated by arrows]
The features are ranked based on **Fisher Criterion**.

For a given feature \( i \), the fisher score is defined as:

\[
J_i = \frac{(\mu_1^i - \mu_2^i)^2}{\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \forall i \in S,
\]

where, \( \mu_j^i \), \( (s_j^i)^2 \) and \( n_j \) are the sample mean, variance and the number of data samples in class \( j \) and \( S \) is the set of selected peaks.

Fisher scores would be high for features having high mean inter-class separation while the total within-class variance is small.
Support Vector Machines (SVMs)

- Binary classifier
- Linearly separable datasets
- Margin maximization

Consider binary classification problem with the training set $S$ defined as:

$$S = \{(x_i, y_i) | x_i \in \mathbb{R}^p, y_i \in \{-1, 1\}, \quad i = 1, 2, \ldots, n\} \tag{4}$$

Let the separating hyperplane $P$ that maximizes the margin be defined as:

$$P = \{x \in \mathbb{R}^p \mid \langle \mathbf{w}, x \rangle - b = 0\} \tag{5}$$

The optimal $(\mathbf{w}, b)$ is found by solving the following optimization problem:

$$\min_{\mathbf{w}, b} \frac{1}{2} ||\mathbf{w}||^2 \quad \text{s.t.} \quad y_i(\langle \mathbf{w}, x_i \rangle - b) \geq 1 \quad \forall i = 1, 2, \ldots, n \tag{6}$$
C-SVMs

- SVMs are susceptible to the presence of outliers.
- Linear separation in real-world datasets.
- SVMs are modified as:

\[
\min_{\mathbf{w}, b, \xi} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \\
\text{subject to} \quad y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle - b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i = 1, 2, \ldots, n
\]
Multi-class SVMs

- Two general approaches to extend SVMs to multi-class problems:
  - One-against-One (OAO) - $n(n - 1)/2$ binary classification tasks
  - One-against-All (OAA) - $n$ binary classification tasks
- Instead, SVMs is extended using **hierarchical clustering**.
- An agglomerative hierarchical cluster tree is generated from the pairwise euclidean distances of the average spectra of cell lines.

- Four binary classification tasks:
  - Cancer Vs. Non-Cancer
  - MCF7 Vs. Rest Cancer
  - MCF10A Vs. MCF12A
  - MDA-MB-231 Vs. BT474
FFS-SVMs Classification framework

Given any two cell lines, the classification framework is built as:

- Spectral Preprocessing
- Fisher-based Feature Selection
  - Peak Finding
  - Peak Coalescing
  - Feature Ranking
- C-SVMs Classification
- Cross Validation using repeated random sub-sampling (100 repetitions).
## Classification Accuracies

<table>
<thead>
<tr>
<th>Classification Task</th>
<th># of selected features</th>
<th>Accuracy (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancer Vs Non-Cancer</td>
<td>38</td>
<td>99.5</td>
<td>99.8</td>
<td>98.6</td>
</tr>
<tr>
<td>MCF7 Vs Rest-Cancer</td>
<td>32</td>
<td>99.3</td>
<td>96.6</td>
<td>100</td>
</tr>
<tr>
<td>BT474 Vs MDA-MB231</td>
<td>42</td>
<td>97.4</td>
<td>91.7</td>
<td>100</td>
</tr>
<tr>
<td>MCF10A Vs MCF12A</td>
<td>42</td>
<td>91</td>
<td>97.1</td>
<td>62</td>
</tr>
</tbody>
</table>

**Table:** Sensitivity, Specificity and average classification accuracy for the four binary classification tasks obtained from C-SVMs and validated using random sub-sampling (100 repetitions).
## Accuracy Comparison

<table>
<thead>
<tr>
<th></th>
<th>Cancer vs. Non-Cancer</th>
<th>MCF7 vs. Rest-Cancer</th>
<th>BT474 vs. MDA-MB231</th>
<th>MCF10A vs. MCF12A</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SVMs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy(%)</td>
<td>99.2</td>
<td>100</td>
<td>97.6</td>
<td>93.4</td>
</tr>
<tr>
<td>Sensitivity(%)</td>
<td>100</td>
<td>100</td>
<td>94.8</td>
<td>100</td>
</tr>
<tr>
<td>Specificity(%)</td>
<td>99.4</td>
<td>100</td>
<td>99.5</td>
<td>80.6</td>
</tr>
<tr>
<td><strong>PCA-SVMs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy(%)</td>
<td>99.4</td>
<td>98.4</td>
<td>98.6</td>
<td>92.8</td>
</tr>
<tr>
<td>Sensitivity(%)</td>
<td>100</td>
<td>95.1</td>
<td>96.4</td>
<td>99.3</td>
</tr>
<tr>
<td>Specificity(%)</td>
<td>98.2</td>
<td>100</td>
<td>99.5</td>
<td>72.9</td>
</tr>
<tr>
<td><strong>PCA-LDA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy(%)</td>
<td>99.5</td>
<td>98.3</td>
<td>96.4</td>
<td>85.8</td>
</tr>
<tr>
<td>Sensitivity(%)</td>
<td>99.9</td>
<td>98.9</td>
<td>88.2</td>
<td>82.8</td>
</tr>
<tr>
<td>Specificity(%)</td>
<td>98.6</td>
<td>97.6</td>
<td>99.3</td>
<td>96.6</td>
</tr>
<tr>
<td><strong>FFS-SVMs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy(%)</td>
<td>97.3</td>
<td>98.9</td>
<td>98.0</td>
<td>89.0</td>
</tr>
<tr>
<td>Sensitivity(%)</td>
<td>100</td>
<td>96.7</td>
<td>93.4</td>
<td>97.6</td>
</tr>
<tr>
<td>Specificity(%)</td>
<td>93.3</td>
<td>100</td>
<td>100</td>
<td>62.3</td>
</tr>
</tbody>
</table>

**Table:** Sensitivity, Specificity and average classification accuracies of four frameworks SVMs, PCA-SVMs, PCA-LDA and FFS-SVMs for the four binary classification tasks. The classification accuracies are obtained from cross-validation using random subsampling (100 repetitions).
Selected Features

<table>
<thead>
<tr>
<th>Cancer vs. Non-Cancer</th>
<th>MCF7 vs. Rest-Cancer</th>
<th>BT474 vs. MDA-MB231</th>
<th>MCF10A vs. MCF12A</th>
</tr>
</thead>
<tbody>
<tr>
<td>1047</td>
<td>1341</td>
<td>1049</td>
<td>1047</td>
</tr>
<tr>
<td>811</td>
<td>986</td>
<td>1063</td>
<td>1320</td>
</tr>
<tr>
<td>823</td>
<td>1322</td>
<td>760</td>
<td>1156</td>
</tr>
<tr>
<td>765</td>
<td>1658</td>
<td>830</td>
<td>1174</td>
</tr>
<tr>
<td>1450</td>
<td>1405</td>
<td>1085</td>
<td>1211</td>
</tr>
<tr>
<td>1660</td>
<td>1066</td>
<td>1318</td>
<td>941</td>
</tr>
<tr>
<td>829</td>
<td>622</td>
<td>1518</td>
<td>811</td>
</tr>
<tr>
<td>1086</td>
<td>1159</td>
<td>604</td>
<td>1338</td>
</tr>
<tr>
<td>1621</td>
<td>1799</td>
<td>1129</td>
<td>719</td>
</tr>
<tr>
<td>785</td>
<td>1316</td>
<td>1661</td>
<td>967</td>
</tr>
</tbody>
</table>

**Table:** The top 10 features selected by FFS for the four binary classification tasks.
Biological Relevance of Selected Features

- **Cancer Vs. Non-Cancer**
  - Five of the top ten discriminative features (811, 823, 765, 829, and 785 cm\(^{-1}\)) all correlate to DNA and RNA vibrational modes.
  - The features 1086, 1450, 1621, and 1660 cm\(^{-1}\) indicate differences in cell membrane composition and cell morphology.

- **MCF7 Vs. Rest-Cancer**
  - The majority of the features correlate to vibrations observed from structural proteins and the secondary protein structure.

- **MCF10A Vs. MCF12A**
  - The analysis of features reveal that the most significant differences may be related to lipid composition.

- **MDA-MB-231 Vs. BT474**
  - Several of the features listed have assignments related to fatty acids and lipids.
Sparse Proximal Support Vector Machines (sPSVMs)
Motivation

- Several embedded methods like Regularized Logistic Regression (RLRs), Sparse Support Vector Machines (S-SVMs) etc., induce *global* sparsity.
- Class-specific features - *local* sparsity.
- *Biomarker-type* information in biomedical applications.

**Research Objective:**

Construct a new binary classifier that incorporates *class-specific* feature selection.

**Sparse Proximal Support Vector Machines (sPSVMs)**
Proximal Support Vector Machines (PSVMs)

- Binary Classifier
- Non-parallel hyperplanes
- Closest to one class and farthest from the other class
- Two generalized eigenvalue problems

PSVMs formulation

Let $A \in \mathbb{R}^{m \times p}$ and $B \in \mathbb{R}^{n \times p}$ represent the two classes. The hyperplane close to class A is given by:

$$P_A = \{ x \in \mathbb{R}^p \mid \langle w_A, x \rangle - b_A = 0 \}$$  \hspace{1cm} (8)

The hyperplane $P_A$ is found by solving the following optimization problem:

$$\min_{w_A \in \mathbb{R}^p, b_A \in \mathbb{R}} \frac{\|Aw_A - eb_A\|^2}{\|Bw_A - eb_A\|^2}$$  \hspace{1cm} (9)
Adding Tikhonov regularization term to (9),

\[
\min_{\mathbf{w}_A \in \mathbb{R}^p, b_A \in \mathbb{R}} \frac{\| A\mathbf{w}_A - e b_A \|^2 + \nu \| [\mathbf{w}_A^\prime \ b_A] \|^2}{\| B\mathbf{w}_A - e b_A \|^2}
\]  

(10)

\(\nu\) is the regularization term.

Let,

\[
G_A = [A - e]'[A - e] + \nu I, \quad H_B = [B - e]'[B - e], \quad z' = [\mathbf{w}_A^\prime \ b_A]
\]  

(11)

Re-writing (10),

\[
\min_{z \in \mathbb{R}^{p+1}} r(z) = \frac{z'G_A z}{z'H_B z}
\]  

(12)

Rayleigh Quotient Problem
Rayleigh Quotient Properties

\[
\min_{z \in \mathbb{R}^{p+1}} r(z) = \frac{z' G z}{z' H z}
\]  

(13)

- **Boundedness:**
  Assuming $H$ is positive definite, $r(z)$ is bounded between $[\lambda_1, \lambda_{p+1}]$, where $\lambda_1$ and $\lambda_{p+1}$ are the minimum and maximum eigenvalues of the following generalized eigenvalue problem $\text{GEV}(G,H)$:

\[
G z = \lambda H z
\]  

(14)

- **Stationarity:**

\[
\nabla r(z) = \frac{G z - r(z) H z}{z' H z}
\]  

(15)

The stationary points are given by the eigenvectors of the generalized eigenvalue problem (14).
PSVMs Solution - Hyperplane $P_A$

\[
\min_{z \in \mathbb{R}^{p+1}} r(z) = \frac{z'G_Az}{z'H_Bz} \quad (16)
\]

or,

\[
\max_{z \in \mathbb{R}^{p+1}} r(z) = \frac{z'H_Bz}{z'G_Az} \quad (17)
\]

The solution is given by the eigenvector corresponding to the maximum eigenvalue of the following generalized eigenvalue problem $\text{GEV}(H_B, G_A)$:

\[
H_Bz = \lambda G_Az \quad (18)
\]
Similarly, the hyperplane $P_B$ (closest to class B and farthest from class A) given by:

$$P_B = \{ x \in \mathbb{R}^p \mid \langle w_B, x \rangle - b_B = 0 \}$$  \hfill (19)$$

can be found by solving for the eigenvector corresponding to maximum eigenvalue of the following generalized eigenvalue problem $\text{GEV}(H_A, G_B)$:

$$H_A z = \lambda G_B z$$  \hfill (20)

$$G_B = [B - e]'[B - e] + \nu I, H_A = [A - e]'[A - e], z' = [w_B' \ b_B]$$  \hfill (21)
Sparse Proximal Support Vector Machines (sPSVMs)

- sPSVMs are constructed by inducing **sparsity** in the hyperplanes obtained from PSVMs.
- Sparsity is defined as the optimal vectors $z_A^*$ and $z_B^*$ having only *few* non-zero components.
- The non-zero coefficients of optimal **sparse** vectors $\hat{z}_A^*$ and $\hat{z}_B^*$ may be interpreted as **class-specific** features.
Regularization in Linear Regression (LR)

- **Sparsity** via regularization has been well studied in the context of linear regression.

Given a dataset $S$ defined as:

$$S = \{(x_i, y_i) \mid x_i \in \mathbb{R}^p, y_i \in \mathbb{R}\}, \quad i = 1, 2, \ldots, n \quad (22)$$

the linear regression problem finds a coefficient vector $w$ that best maps the input vector $x$ to the output $y$.

- The following **least squares (LS)** problem is solved to obtain $w$:

$$\min_{w} \|y - Xw\|^2 \quad (23)$$

$X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, $w \in \mathbb{R}^p$
Sparsity is induced in linear regression problems via $l_1$-norm

$$\min_w ||y - Xw||_2^2 + \lambda ||w||_1$$

Well known efficient algorithms like Least Angle Regression (LARS) exist in literature to solve (24)

Idea:

Transform **PSVMs** to an equivalent **least-squares (LS)** problem and induce **sparsity** via \( l_1 \)-norm
Equivalence between Eigendecomposition and Linear Regression

**Theorem 1:** Consider a real matrix $X \in \mathbb{R}^{n \times p}$ with rank $r \leq \min(n, p)$. Let matrices $V \in \mathbb{R}^{p \times p}$ and $D \in \mathbb{R}^{p \times p}$ satisfy the following relation:

$$V^T (X^T X) V = D$$

where, $D = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots \sigma_r^2, 0, 0, \ldots, 0)_{p \times p}$. Assume $\sigma_1^2 \geq \sigma_2^2 \geq \ldots \geq \sigma_r^2$. For the following least-squares problem,

$$\min_{\alpha, \beta} \sum_{i=1}^{n} ||X_i - \alpha \beta^T X_i||^2 + \lambda \beta^T \beta$$

subject to $\alpha^T \alpha = 1$

$\beta_{opt} \propto V_1$, where $X_i$ is the $ith - row$ of matrix $X$ and $V_1$ is the eigenvector corresponding to the largest eigenvalue $\sigma_1^2$.

PSVMs via Least-Squares Approach

- Consider the **generalized eigenvalue problem** in PSVMs given by:

\[ H_B \mathbf{z} = \lambda G_A \mathbf{z} \]  \hspace{1cm} (27)

\[ G_A = [A - e]'[A - e] + \nu I, \quad H_B = [B - e]'[B - e], \quad \mathbf{z}' = [\mathbf{w}_A' \ b_A] \]  \hspace{1cm} (28)

- Assuming \( G_A \) and \( H_B \) are **positive-definite**, the **cholesky decomposition** of the matrices give:

\[ G_A = L_A L_A^T = U_A^T U_A \]  \hspace{1cm} (29)

\[ H_B = L_B L_B^T = U_B^T U_B \]  \hspace{1cm} (30)

\( L_A, L_B \) are lower triangular matrices, and \( U_A, U_B \) are upper triangular matrices.
Relation between generalized eigenvalue problems and SVD

- Substituting (29) and (30) in (27),

\[
H_B z = \lambda G_A z \quad (31)
\]

\[L_B L_B^T z = \lambda U_A^T U_A z \quad (32)\]

\[
U_A^{-T} L_B L_B^T z = \lambda U_A z \quad (33)
\]

\[
U_A^{-T} L_B L_B^T U_A^{-1} U_A z = \lambda U_A z \quad (34)
\]

\[
(L_B^T U_A^{-1})^T (L_B^T U_A^{-1}) U_A z = \lambda U_A z \quad (35)
\]

Let, \( \hat{X} = L_B^T U_A^{-1} \) and \( v = U_A z \)

\[
(\hat{X}^T \hat{X}) v = \lambda v \quad (36)
\]
PSVMs via Least-Squares Approach

- **PSVMs** can now be solved by an equivalent least squares problem.
- Using **Theorem 1** and substituting $X = L_B^T U_A^{-1}$, $\beta = U_A \hat{\beta}$ in (26),

$$
\min_{\alpha, \hat{\beta}} \sum_{i=1}^{n} \| (L_B^T U_A^{-1})_i - \alpha \hat{\beta}^T U_A (L_B^T U_A^{-1})_i \|^2 + \lambda \hat{\beta}^T U_A U_A \hat{\beta}
$$

s.t. $\alpha^T \alpha = 1$

- Substituting $U_A^T U_A = G_A$ and $(L_B^T U_A^{-1})_i = U_A^{-T} U_B,i$,

$$
\min_{\alpha, \hat{\beta}} \sum_{i=1}^{n} \| U_A^{-T} U_B,i - \alpha \hat{\beta}^T U_B,i \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta}
$$

s.t. $\alpha^T \alpha = 1$
PSVMs via Least-Squares Approach

Re-writing in a nicer way,

\[
\min_{\alpha, \hat{\beta}} \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta}
\]

\[\text{s.t.} \quad \alpha^T \alpha = 1\]

\(\hat{\beta}_{opt}\) is proportional to \(z_A^*\) representing the hyperplane \(P_A\) in PSVMs.
Solution Strategy

\[
\min_{\alpha, \hat{\beta}} \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta}
\]
\[\text{s.t.} \quad \alpha^T \alpha = 1 \]  

Strategy:
The optimization problem is solved by alternating over \( \alpha \) and \( \hat{\beta} \).
Solving for $\alpha$

- The **PSVMs-via-LS** is given by:

$$\min_{\alpha, \hat{\beta}} \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G A \hat{\beta}$$

s.t. \( \alpha^T \alpha = 1 \) \hspace{1cm} (41)

- For a fixed \( \hat{\beta} \), the following optimization problem is solved to obtain $\alpha$.

$$\min_{\alpha, \hat{\beta}} \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2$$

s.t. \( \alpha^T \alpha = 1 \) \hspace{1cm} (42)
Solving for $\alpha$

Expanding the objective function,

\[
(U_B U_A^{-1} - U_B \hat{\beta} \alpha^T)^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T) 
\approx -2\alpha^T U_A^{-T} H_B \hat{\beta} + \alpha^T \alpha \hat{\beta} H_B \hat{\beta}
\]

Subsituting $\alpha^T \alpha = 1$, the optimization problem in (42) reduces to:

\[
\max_{\alpha} \quad \alpha^T U_A^{-T} H_B \hat{\beta} \\
\text{s.t.} \quad \alpha^T \alpha = 1
\]

An analytical solution for this problem exists and the $\alpha_{opt}$ is given by,

\[
\alpha_{opt} = \frac{U_A^{-T} H_B \hat{\beta}}{\|U_A^{-T} H_B \hat{\beta}\|}
\]
Solving for $\hat{\beta}$

- The PSVMs-via-LS is given by:

$$\min_{\alpha, \hat{\beta}} \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta}$$

s.t. $\alpha^T \alpha = 1$ (47)

- Let $\hat{A}$ be an orthogonal matrix such that $[\alpha; \hat{A}]$ is $p \times p$ orthogonal. Then the objective function can be written as,

$$\| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta} \quad (48)$$

$$\approx tr(U_B U_A^{-1} - U_B \hat{\beta} \alpha^T)^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T) \quad (49)$$

$$\approx tr([\alpha; \hat{A}] [\alpha; \hat{A}]^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T)^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T) \quad (50)$$

$$\approx tr([\alpha; \hat{A}]^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T)^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T) [\alpha; \hat{A}] \quad (51)$$

$$\approx tr((U_B U_A^{-1} - U_B \hat{\beta} \alpha^T [\alpha; \hat{A}])^T (U_B U_A^{-1} - U_B \hat{\beta} \alpha^T [\alpha; \hat{A}]) \quad (52)$$
Solving for $\hat{\beta}$

\[
\approx \text{tr}((\alpha^T \hat{A}^T) U_A^{-T} U_B U_B U_A^{-1}[\alpha; \hat{A}] - (\alpha^T \hat{A}^T) U_A^{-T} U_B U_B \hat{\beta} \alpha^T [\alpha; \hat{A}]
- (\alpha^T \hat{A}^T) \alpha \hat{\beta}^T U_B U_B U_A^{-1}[\alpha; \hat{A}] + (\alpha^T \hat{A}^T) \alpha \hat{\beta}^T U_B U_B \hat{\beta} \alpha^T [\alpha; \hat{A}])
\approx \text{tr}((\alpha^T \hat{A}^T) U_A^{-T} U_B U_B U_A^{-1}[\alpha; \hat{A}] - (\alpha^T \hat{A}^T) U_A^{-T} U_B U_B \hat{\beta}
- \hat{\beta}^T U_B U_B U_A^{-1}[\alpha; \hat{A}] + \hat{\beta}^T U_B U_B \hat{\beta})
\approx \text{tr}((U_B U_A^{-1}[\alpha; \hat{A}])^T (U_B U_A^{-1}[\alpha; \hat{A}]) + (U_B \hat{\beta})^T (U_B \hat{\beta})
- 2(U_B \hat{\beta})^T U_B U_A^{-1}[\alpha; \hat{A}])
\approx \text{tr}((U_B U_A^{-1}[\alpha; \hat{A}] - U_B \hat{\beta})^T (U_B U_A^{-1}[\alpha; \hat{A}] - U_B \hat{\beta}))
\approx \|U_B U_A^{-1}[\alpha; \hat{A}] - U_B \hat{\beta}\|^2
\approx \|U_B U_A^{-1} \alpha - U_B \hat{\beta}\|^2 + \|U_B U_A^{-1} \hat{A}\|^2
\]
Solving for $\hat{\beta}$

For a fixed $\alpha$, utilizing (53), the optimization problem in (47) reduces to ridge-regression:

$$\min_{\beta} \| U_B U_A^{-1} \alpha - U_B \hat{\beta} \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta}$$  \hspace{1cm} (54)

An analytical solution exists and $\hat{\beta}_{opt}$ can be found by:

$$\hat{\beta}_{opt} = (H_B + \lambda G_A)^{-1} H_B U_A^{-1} \alpha$$  \hspace{1cm} (55)
Algorithm 1 PSVMs-via-LS \((H_B, G_A)\)

1. Initialize \(\hat{\beta}\).
2. Find the upper triangular matrix \(U_A\) from the cholesky decomposition of \(G_A\).
3. Find \(\alpha\) from the following relation:

\[
\alpha = \frac{U_A^{-T} H_B \hat{\beta}}{\| U_A^{-T} H_B \hat{\beta} \|} \quad (56)
\]

4. Find \(\hat{\beta}\) as follows:

\[
\hat{\beta} = (H_B + \lambda G_A)^{-1} H_B U_A^{-1} \alpha \quad (57)
\]

5. Alternate between 3 and 4 until convergence.
Sparse Proximal Support Vector Machines (sPSVMs)

- The PSVMs-via-LS is given by:

\[
\begin{align*}
\min_{\alpha, \hat{\beta}} & \quad \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta} \\
\text{s.t.} & \quad \alpha^T \alpha = 1
\end{align*}
\]  

(58)

- Sparsity is introduced by adding $l_1$-norm in the above problem.

\[
\begin{align*}
\min_{\alpha, \hat{\beta}} & \quad \| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta} + \delta \| \hat{\beta} \|_1 \\
\text{s.t.} & \quad \alpha^T \alpha = 1
\end{align*}
\]  

(59)

The sPSVMs (59) is again solved by alternating over $\alpha$ and $\hat{\beta}$. 
Solving for $\alpha$

- The sPSVMs is given by:

\[
\min_{\alpha, \hat{\beta}} \left\| U_B U_A^{-1} - U_B \hat{\beta} \alpha^T \right\|^2 + \lambda \hat{\beta}^T G_A \hat{\beta} + \delta \| \hat{\beta} \|_1
\]

s.t. $\alpha^T \alpha = 1$  (60)

- For a fixed $\hat{\beta}$, an analytical solution exists for $\alpha$ and is given by,

\[
\alpha_{opt} = \frac{U_A^{-T} H_B \hat{\beta}}{\| U_A^{-T} H_B \hat{\beta} \|}
\]

(61)
Solving for $\hat{\beta}$

- For a fixed $\alpha$, utilizing (52), sPSVMs in (59) can be written as:

$$\min_{\hat{\beta}} \| U_B U_A^{-1} \alpha - U_B \hat{\beta} \|^2 + \lambda \hat{\beta}^T G_A \hat{\beta} + \delta \| \hat{\beta} \|_1$$  \hspace{1cm} (62)

- Expanding (62),

\[
\begin{align*}
\min_{\hat{\beta}} & \quad (U_B U_A^{-1} \alpha - U_B \hat{\beta})^T (U_B U_A^{-1} \alpha - U_B \hat{\beta}) + \lambda \hat{\beta}^T G_A \hat{\beta} + \delta \| \hat{\beta} \|_1 \\
\min_{\hat{\beta}} & \quad - \alpha^T U_A^{-T} H_B^T \hat{\beta} - \hat{\beta}^T H_B U_A^{-1} \alpha + \hat{\beta}^T H_B \hat{\beta} + \lambda \hat{\beta}^T G_A \hat{\beta} + \delta \| \hat{\beta} \|_1 \\
\min_{\hat{\beta}} & \quad \hat{\beta}^T (H_B + \lambda G_A) \hat{\beta} - \alpha^T U_A^{-T} H_B^T \hat{\beta} - \hat{\beta}^T H_B U_A^{-1} \alpha + \delta \| \hat{\beta} \|_1 \\
\min_{\hat{\beta}} & \quad \hat{\beta}^T (H_B + \lambda G_A) \hat{\beta} - 2 \alpha^T U_A^{-T} H_B \hat{\beta} + \delta \| \hat{\beta} \|_1
\end{align*}
\]  \hspace{1cm} (63)
Solving for $\hat{\beta}$

Assuming, $W^T = [U_B \sqrt{(\lambda)} U_A], y^T = [U_B U_A^{-1} \alpha \ 0],$

$$\min_{\hat{\beta}} \hat{\beta}^T W^T W \hat{\beta} - 2y^T W \hat{\beta} + \delta \|\hat{\beta}\|_1$$ (64)

LASSO Regression

Efficient algorithms like Least Angle Regression (LARS) exist to solve (64).
Algorithm 2 sPSVMs \((H_B, G_A)\)

1. Initialize \(\hat{\beta}\)
2. Find the upper triangular matrix \(U_A\) and \(U_B\) from the cholesky decomposition of \(G_A\) and \(H_B\).
3. Find \(\alpha\) from the following equation:

\[
\alpha = \frac{U_A^{-T}H_B\hat{\beta}}{\|U_A^{-T}H_B\hat{\beta}\|} \tag{65}
\]

4. Construct \(W\) and \(y\) as follows:

\[
W = [U_B \quad \sqrt{\lambda}U_A]^T, \quad y = [U_B U_A^{-1} \alpha \quad 0]^T \tag{66}
\]

and solve the following LASSO regression to obtain \(\hat{\beta}\):

\[
\min_{\hat{\beta}} \quad \hat{\beta}^T W^T W \hat{\beta} - 2y^T W \hat{\beta} + \delta \|\hat{\beta}\|_1 \tag{67}
\]

5. Alternate between 3 and 4 until convergence.
Results

- **sPSVMs** is compared with other classification methods **SVMs**, **LDA** and **PSVMs** on publicly available datasets.

- **10-fold** cross validation is performed and the average accuracies are reported.

- For each fold, $\lambda$ is chosen as zero and a grid search is performed over different values of $\nu$ and $\delta$ to choose the best values that yield the highest classification accuracy.

- Final model for testing is chosen as the one that yields the highest accuracy among the 10 folds.
**Results - Example (Spambase dataset)**

- The **spambase** dataset consists of 4601 samples and 57 features with 1813 samples in class 1 and 2788 samples in class 2.
- $\nu$ and $\delta$ are varied in logspace between $10^{-3} - 10^4$ and $10^{-5} - 1$ respectively.

<table>
<thead>
<tr>
<th>Fold</th>
<th>Nu</th>
<th>Delta</th>
<th>Accuracy*</th>
<th># Features&lt;sub&gt;A&lt;/sub&gt;</th>
<th># Features&lt;sub&gt;B&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>0.1</td>
<td>72.6%</td>
<td>14</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>$10^{-3}$</td>
<td>$10^{-5}$</td>
<td>69.8%</td>
<td>58</td>
<td>55</td>
</tr>
<tr>
<td>3</td>
<td>0.01</td>
<td>0.1</td>
<td>71.5%</td>
<td>17</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>$10^{-5}$</td>
<td>73.9%</td>
<td>55</td>
<td>37</td>
</tr>
<tr>
<td>5</td>
<td>0.01</td>
<td>0.1</td>
<td><strong>80.9%</strong></td>
<td>14</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>$10^{-3}$</td>
<td>0.1</td>
<td>76.3%</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>$10^{-3}$</td>
<td>0.1</td>
<td>75%</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>0.1</td>
<td>73.5%</td>
<td>15</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>0.01</td>
<td>73.7%</td>
<td>28</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>0.01</td>
<td>74.6%</td>
<td>28</td>
<td>11</td>
</tr>
</tbody>
</table>

*Table: Classification accuracies of the 10-folds for Spambase dataset*
Results

- All the classification methods have been implemented in MATLAB.
- **LibSVM** package is used for SVMs.
- LDA is solved using the 'classify' function in MATLAB.
- PSVMs are solved using the 'eig' function in MATLAB.
- **LARS** package provided by the authors on their website is used for sPSVMs.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Dimensions</th>
<th>SVMs</th>
<th>LDA</th>
<th>PSVMs</th>
<th>sPSVMs</th>
<th># features(_A)</th>
<th># features(_B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WDBC</td>
<td>569*30</td>
<td>96.8%</td>
<td>95.6%</td>
<td>95.4%</td>
<td>97.5%</td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>Spambase</td>
<td>4601*57</td>
<td>77.1%</td>
<td><strong>90.7%</strong></td>
<td>71.6%</td>
<td>78.6%</td>
<td>14</td>
<td>4</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>351*34</td>
<td><strong>91.2%</strong></td>
<td>88.3%</td>
<td>84.6%</td>
<td>85.5%</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>WPBC</td>
<td>198*33</td>
<td><strong>84.9%</strong></td>
<td>72.2%</td>
<td>77.7%</td>
<td>82.8%</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>Mushroom</td>
<td>8124*126</td>
<td>98.6%</td>
<td>99.2%</td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
<td>42</td>
<td>41</td>
</tr>
<tr>
<td>German</td>
<td>1000*20</td>
<td><strong>76.4%</strong></td>
<td>71.8%</td>
<td>68.7%</td>
<td>71.7%</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Waveform</td>
<td>5000*21</td>
<td><strong>88.6%</strong></td>
<td>82.9%</td>
<td>78.5%</td>
<td>78%</td>
<td>8</td>
<td>14</td>
</tr>
</tbody>
</table>

Table: Classification accuracies for different classification methods on publicly available datasets.
Results - HDLSS datasets

- **sPSVMs** has been tested on three publicly available HDLSS datasets.
- The results are compared with other classification frameworks that combine dimensionality reduction techniques with a standard classification model.
- The chosen dimensionality reduction techniques are **Principal Component Analysis (PCA)**, **Fisher-based Feature Selection (FFS)** and **Correlation-based Feature Selection (CFS)**.
- The number of principal components in **PCA** are chosen such that they account for 80% of the total variance in data.
- The standard classification methods tested are **SVMs**, **LDA** and **PSVMs**.
- Classification accuracies are obtained using **10-fold** cross validation.
Results - HDLSS datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Dimensions</th>
<th>SVMs</th>
<th>PSVMs</th>
<th>sPSVMs</th>
<th># features_A</th>
<th># features_B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Colon</td>
<td>62*2000</td>
<td>75.9%</td>
<td>87.1%</td>
<td>89%</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td>DBWorld</td>
<td>64*4702</td>
<td>88.1%</td>
<td>90.7%</td>
<td>92.4%</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>DLBCL</td>
<td>77*5469</td>
<td><strong>94.8%</strong></td>
<td>81.8%</td>
<td>81.8%</td>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

Table: Classification accuracies for publicly available HDLSS datasets using SVMs, PSVMs, and sPSVMs.

Colon dataset

<table>
<thead>
<tr>
<th></th>
<th>SVMs</th>
<th>PSVMs</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFS</td>
<td>92.4%</td>
<td>97%</td>
</tr>
<tr>
<td>CFS</td>
<td>83.9%</td>
<td>88.8%</td>
</tr>
<tr>
<td>PCA</td>
<td>90.7%</td>
<td>87.4%</td>
</tr>
</tbody>
</table>

DBWorld dataset

<table>
<thead>
<tr>
<th></th>
<th>SVMs</th>
<th>PSVMs</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFS</td>
<td>94.1%</td>
<td>97.1%</td>
</tr>
<tr>
<td>CFS</td>
<td>97.1%</td>
<td>97.1%</td>
</tr>
<tr>
<td>PCA</td>
<td>89.5%</td>
<td>82%</td>
</tr>
</tbody>
</table>

DLBCL dataset

<table>
<thead>
<tr>
<th></th>
<th>SVMs</th>
<th>PSVMs</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFS</td>
<td>96.3%</td>
<td>91.1%</td>
</tr>
<tr>
<td>CFS</td>
<td>98.8%</td>
<td>79.3%</td>
</tr>
<tr>
<td>PCA</td>
<td>96.3%</td>
<td>83.4%</td>
</tr>
</tbody>
</table>
Results - HDLSS datasets

- **sPSVMs** is compared with other embedded methods **Regularized Logistic Regression (RLR)** and **Sparse SVMs (S-SVMs)** on the HDLSS datasets.
- Classification accuracies are obtained using a **10-fold** cross validation.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>RLR</th>
<th># features</th>
<th>S-SVMs</th>
<th># features</th>
<th>sPSVMs</th>
<th># features A</th>
<th># features B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Colon</td>
<td>83.9%</td>
<td>12</td>
<td>69.5%</td>
<td>16</td>
<td>89%</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td>DBWorld</td>
<td>82.8%</td>
<td>9</td>
<td>82.6%</td>
<td>14</td>
<td>92.4%</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>DLBCL</td>
<td>96.1%</td>
<td>25</td>
<td>88.2%</td>
<td>12</td>
<td>81.8%</td>
<td>2</td>
<td>7</td>
</tr>
</tbody>
</table>

Table: Classification accuracies for different classification methods on publicly available HDLSS datasets.
Publications

- M. Fenn, V. Pappu, P. Xanthopoulos & P.M. Pardalos, *Data Mining and Optimization Applied to Raman Spectroscopy for Oncology Applications* - BIOMAT (2011)
- M.B. Fenn, & V. Pappu, *Data Mining for Cancer Biomarkers with Raman Spectroscopy* - Data Mining for Biomarker Discovery (2012)
Books