O-IPCAC and its application to EEG classification

A. Rozza, G. Lombardi, M. Rosa, and E. Casiraghi
{rozza,lombardi,rosa,casiragh}@dsi.unimi.it

Università degli Studi di Milano
Outline

1 Introduction
   - EEG classification
   - IPCAC

2 O-IPCAC
   - Theoretical Problems
   - The Algorithm

3 Experimental Evaluation
   - EEG Dataset
   - Results

4 Conclusions
   - References

5 Appendix
EEG classification

- Recently this problem is raising a wide interest since it is the fundamental step of Brain to Computer Interface (BCI) systems: the translation of the brain activity into commands for computers;
- The task of EEG classification is a hard problem:
  - The data are high dimensional;
  - The classes to be discriminated are often highly unbalanced;
  - The selection of discriminative information is difficult;
  - The cardinality of the training set is often lower than the space dimensionality.
Existing Approaches

- Feature extraction/selection techniques are generally used;
- This approach causes loss of discriminative information, and might affect the classification accuracy.

Different Approach

- Develop an efficient classifier that deals with high dimensional datasets whose cardinality is lower than the space dimensionality.
- Apply it to the raw data.
**Existing Approaches**

- Feature extraction/selection techniques are generally used;
- This approach causes loss of discriminative information, and might affect the classification accuracy.

**Different Approach**

- Develop an efficient classifier that deals with high dimensional datasets whose cardinality is lower than the space dimensionality.
  - Apply it to the raw data.
Isotropic Principal Component Analysis Classifier [5]

**IPCAC**

A linear two-class classification algorithm, based on a new estimation of the Fisher Subspace [1], assuming points drawn by an isotropic Mixture of two Gaussian Functions.

- The Fisher subspace is spanned by the one-dimensional vector defined as follows:

\[
F = \frac{\mu_A - \mu_B}{\|\mu_A - \mu_B\|} \tag{1}
\]

**Training task:** In this phase the classifier exploits the training set to estimate the Fisher subspace \( F \) and the thresholding value \( \gamma \).

**Classification task:** An unknown test point \( p \) is classified by projecting it on \( F \) and then thresholding with \( \gamma \).
Isotropic Principal Component Analysis Classifier [5]

IPCAC

A linear two-class classification algorithm, based on a new estimation of the Fisher Subspace [1], assuming points drawn by an isotropic Mixture of two Gaussian Functions.

- The Fisher subspace is spanned by the one-dimensional vector defined as follows:

\[
F = \frac{\mu_A - \mu_B}{\|\mu_A - \mu_B\|} \tag{1}
\]

**Training task:** In this phase the classifier exploits the training set to estimate the Fisher subspace \( F \) and the thresholding value \( \gamma \).

**Classification task:** An unknown test point \( p \) is classified by projecting it on \( F \) and then thresholding with \( \gamma \).
IPCA-based Classifier - Training phase

Data whitening

- The probability distribution related to several classification tasks is not mean-centered, and its random variables are often correlated; To avoid this problem data whitening is performed ($W$ is the whitening matrix).

Fisher subspace estimation

- The whitened training points are employed to compute the class means $\mu_A$ and $\mu_B$, and $F$ (see Equation (1)).

Thresholding value

$$\gamma = \left\{ \arg\max_{\{\tilde{\gamma}\}} \left\{ w \cdot (p_i - \tilde{\mu}) \right\} \right\}$$

$Score(\tilde{\gamma})$
Theoretical Problems in High Dimensionality

Covariance Matrix Estimation Problem

Given the matrix $P \in \mathbb{R}^{D \times N}$, representing a training dataset $P = P_A \cup P_B, |P| = N = N_A + N_B$, let $\alpha$ be the ratio $D/N$.

If $\alpha \approx 1$, the sample covariance matrix $\tilde{\Sigma} = \frac{1}{N-1} PP^T$ is not a consistent estimator of the population covariance matrix $\Sigma$ [3].
Noise Problem

- Assuming that $\Sigma = \Sigma^* + \sigma^2 I$, where $\Sigma^*$ has rank $k < D$ and $\sigma^2 I$ represents the contribution of a zero mean Gaussian noise affecting the data;
- Calling $\sigma^2 = \lambda_1 = \ldots = \lambda_{D-k-1} < \ldots < \lambda_D$ the ordered eigenvalues of $\Sigma$;

Only the portion of the spectrum of $\Sigma$ above $\sigma^2 + \sqrt{\alpha}$ can be correctly estimated from the sample [4].

- Denoting with $\tilde{\lambda}_1 < \ldots < \tilde{\lambda}_D$ the ordered eigenvalues of $\tilde{\Sigma}$;

If $\alpha \approx 1$ the estimates of the smallest eigenvalues $\tilde{\lambda}_i$ can be much larger than the real ones, and the corresponding estimated eigenvectors are uncorrelated with the real ones.
### Theoretical Problems (2)

#### Noise Problem

- Assuming that \( \Sigma = \Sigma^* + \sigma^2I \), where \( \Sigma^* \) has rank \( k < D \) and \( \sigma^2I \) represents the contribution of a zero mean Gaussian noise affecting the data;
- Calling \( \sigma^2 = \lambda_1 = \ldots = \lambda_{D-k-1} < \ldots < \lambda_D \) the ordered eigenvalues of \( \Sigma \);

Only the portion of the spectrum of \( \Sigma \) above \( \sigma^2 + \sqrt{\alpha} \) can be correctly estimated from the sample [4].

- Denoting with \( \tilde{\lambda}_1 < \ldots < \tilde{\lambda}_D \) the ordered eigenvalues of \( \tilde{\Sigma} \);

If \( \alpha \approx 1 \) the estimates of the smallest eigenvalues \( \tilde{\lambda}_i \) can be much larger than the real ones, and the corresponding estimated eigenvectors are uncorrelated with the real ones.
Problems with dimensionality reduction

- Dimensionality reduction might delete discriminative information, decreasing the classification performance;

Consider two classes with the shape of parallel pancakes in \( \mathbb{R}^D \):

1. if the direction defined by the Fisher subspace in the original space is orthogonal to the subspace \( \pi_d \) defined by the first \( d \leq D \) principal components, the dimensionality reduction process projects the data on \( \pi_d \), obtaining an isotropic mixture of two completely overlapped Gaussian distributions.
Problems with dimensionality reduction (2)

Figure: Parallel Pancakes
O-IPCAC: the algorithm (1)

- To estimate the linear transformation $W$, which represents the partial whitening operator, we apply the Truncated Singular Value Decomposition;

- The $d$ largest singular values on the diagonal of $Q_d$, and the associated left singular vectors, are employed to project the points in $P$ on the subspace $SP_d$ spanned by the columns of $U_d$, and to perform the whitening, as follows:

$$
\bar{P}_{W_d} = q_d Q_d^{-1} P_{SP_d} = q_d Q_d^{-1} U_d^T P = W_d P
$$
To avoid this information loss, we add to the partially whitened data the residuals $R$ of the points in $P$ with respect to their projections on $SP_d$:

$$R = P - U_d P⊥SP_d = P - U_d U_d^T P$$

$$\bar{P}_W_d = U_d \bar{P}_W_d + R = U_d W_d P + P - U_d U_d^T P$$

$$= \left(q_d U_d Q_d^{-1} U_d^T + I - U_d U_d^T\right) P$$

$$= WP$$

where $W \in \mathbb{R}^{D \times D}$ represents the linear transformation that whitens the data along the first $d$ principal components, while keeping unaltered the information along the remaining components.
O-IPCAC: the algorithm (3)

- The Fisher subspace is estimated by exploiting the whitened class means, \( \mu_A \) and \( \mu_B \), obtained by the class means in the original space \( \hat{\mu}_A \) and \( \hat{\mu}_B \) as follows:

\[
\mu_A = W \hat{\mu}_A \\
= \left( q_d U_d Q_d^{-1} U_d^T + I - U_d U_d^T \right) \hat{\mu}_A \\
= q_d U_d Q_d^{-1} U_d^T \hat{\mu}_A + \hat{\mu}_A - U_d U_d^T \hat{\mu}_A
\]

- Using these quantities we estimate \( f = \frac{\mu_A - \mu_B}{\|\mu_A - \mu_B\|} \).

- We process an unknown point \( p \) by transforming it with \( W \), and projecting it on \( f \):

\[
w = W^T f = q_d U_d^T Q_d^{-1} U_d f + f - U_d^T U_d f
\]

- Given a thresholding value \( \gamma \), \( p \) is assigned to class \( A \) if \( w \cdot p < \gamma \), to class \( B \) otherwise.
O-IPCAC: the algorithm (4)

- We never explicitly compute the matrix $W$, but we perform the matrix times vector operations thus preventing a quadratic time/space complexity.
The Online algorithm

- With training sets of high cardinality, or when mini-batches of training data are dynamically supplied, subsequent training phases must be applied to update the classification model.

- To this aim, the algorithm has been extended to perform **online/incremental** training by updating:
  
  $N_k, N_{A,k}, N_{B,k}$ : number of training points seen until the k-th training phase;
  
  $\mu_k, \hat{\mu}_{A,k}, \hat{\mu}_{B,k}$ : the means employed to obtain the centered sets $\mathcal{P}_k, \mathcal{P}_{A,k}$, and $\mathcal{P}_{B,k}$ respectively;
  
  $U_{d_k}, Q_{d_k}, V_{d_k}$ : the SVD matrices related to $\mathcal{P}_k$, truncated to $d_k$ principal components;
  
  $\sigma_A, \sigma_B$ : the standard deviations of the projections $w_k^T \mathcal{P}_{A,k}$ and $w_k^T \mathcal{P}_{B,k}$.
The data used in our tests have been distributed by the organizers of the MLSP 2010 [2] competition and consist of EEG brain signals collected while the subject viewed satellite images and tried to detect those containing a predefined target:

- 64 channels of EEG data;
- The total number of samples is 176378, and the sampling rate is 256Hz;
- During the EEG recording 2775 satellite images were shown, partitioned in 75 activation blocks with 37 images per block;
- The classifier must analyze the brain activity to recognize those images containing the target.
Pre-processing

- We pre-processed each channel with a Gaussian filter with cut-frequency of 2.2Hz, and we subtracted the filtered data from the original one to obtain high-pass filtered signals.
- These signals were then used to extract $64 \times 97$ image blocks, where each image block starts exactly 65 time samples ($\approx 250$ms) after the corresponding image trigger.
- The extracted blocks are serialized in 2775 vectors in $\mathbb{R}^{6208}$, of which only 58 points represent images with target.
Performance evaluation

To evaluate the performance of our classifier:

- We computed the Receiver Operating Characteristic (ROC) curve;
- We estimated the Area Under the Curve (AUC).

To obtain an unbiased evaluation, we performed ten-fold cross validation, and we averaged the computed sensitivity and specificity values.
Results

Figure: ROC curves

- Alma2
- OISVM
- PA
- O-IPCAC
- SOP
- ILDA
- Perceptron
## Results and Comparison

### Table: AUC per classifier

<table>
<thead>
<tr>
<th>Classifier</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>O-IPCAC</td>
<td>0.9541</td>
</tr>
<tr>
<td>OISVM</td>
<td>0.8766</td>
</tr>
<tr>
<td>SOP</td>
<td>0.8479</td>
</tr>
<tr>
<td>ILDA</td>
<td>0.5315</td>
</tr>
<tr>
<td>Alma</td>
<td>0.5110</td>
</tr>
<tr>
<td>PA</td>
<td>0.4835</td>
</tr>
<tr>
<td>Perceptron</td>
<td>0.4507</td>
</tr>
</tbody>
</table>
Conclusions and Future Works

Conclusions

We propose an online/incremental linear binary classifier that has been developed to deal with:

1. High dimensional data;
2. Classification problems where the cardinality of the point set is high;
3. Data dynamically supplied;
4. Highly unbalanced training sets whose cardinality is lower than the space dimensionality.

These peculiarities allow to manage EEG classification problem:

1. Without focusing on complex features extraction/selection techniques;
2. Dealing with the raw data;
3. Achieving good results.
Conclusions and Future Works

Future Works

- Apply our method to biological data (such as Microarray) where the datasets are characterized by a very large ratio between dimension and training points.
- Develop an adaptive version of O-IPCAC, to cope with classification problems where the probability distribution underlying the data changes with time.


References II

- **D. Paul.**
  Asymptotics of sample eigenstructure for a large dimensional spiked covariance model.

- **A. Rozza, G. Lombardi, and E. Casiraghi.**
  Novel ipca-based classifiers and their application to spam filtering.
Any questions?
Whitening Process

1. estimate the expectation \( \tilde{\mu} = N^{-1} \sum_i p_i \), and the covariance matrix \( \tilde{\Sigma} = N^{-1} \sum_i (p_i - \tilde{\mu})(p_i - \tilde{\mu})^T \);

2. estimate the principal components through the covariance matrix Eigen-decomposition \( X \Lambda X^T = \tilde{\Sigma} \);

3. estimate the whitening matrix as \( W = X \Lambda^{-\frac{1}{2}} X^T \).