BCI signal classification using a Riemannian-based kernel

A. Barachant\textsuperscript{1,2}, S. Bonnet\textsuperscript{1}, M. Congedo\textsuperscript{2}, C. Jutten\textsuperscript{2}

\textsuperscript{1} CEA-LETI/DTBS, Grenoble, France
\textsuperscript{2} GIPSA-lab, Grenoble, France

25-April-2012
Brain-Computer Interface (BCI)

- **Acquisition**:  
  - EEG system with scalp electrodes (system 10-20)  
  - BCI paradigm using motor imagery (asynchronous BCI)

- **Signal Processing**:  
  - Pre-processing: artefact removal, EEG signal band-pass filtering  
  - Two-class classification: Discriminate between two MI tasks
BCI Motor imagery (standard approach)

- **Spatial Filtering** step (usually data-driven)
  - CSP Criterion: promote variance difference between two classes
  - Joint diagonalization of class-conditional mean spatial covariance matrices
  - Select N_f spatial filters: Loss of information

**Frequency filtering**
8-30 Hz, i.e. $\mu$, $\beta$

**Spatial filtering**
$Z = W^T X$

**Feature extraction**
Log Variance

**Signal Classification**
LDA / SVM

- $X_p \in \mathbb{R}^{E \times T}$
- $C_p = \Sigma(X_p)$
- $C_p = \frac{1}{T-1} X_p^T X_p$
Can we avoid the spatial filtering?

- Re-interpret CSP-based linear classification
  - CSP decision function:
    \[
    h(X) = v_0 + \sum_{n=1}^{N_f} v_n \log [W^T \Sigma(X) W]_{n,n}
    \]

  - Omitting log() operator, this yields a feature dimension space \(E^* = E \times (E+1)\)

  - Spatial filter matrix \(W\) can be estimated from \(u\), since
    \[
    u = \text{vec}[W \text{diag}(v) W]
    \]

- [Reuderink, 2011] Direct covariance classification on whitened trials
  \[
  \hat{X} = P^T X
  \]
  \[
  h(X) = u_0 + u^T \text{vec}[P^T \Sigma(X) P]
  \]
Can we go further?

How to classify covariance matrices in MI-based BCI?

- Bayesian Framework (Wishart distribution)
- Algebraic Framework (geometric approach)
- Information Geometry Framework

To do that, we need a dedicated metric.
Riemannian manifold of SPD matrices

- Space of Symmetric Positive-Definite (SPD) matrices
- Differentiable manifold \((\text{dimension } E^*)\)
  - Covariance matrices are points in this manifold and Riemannian distance can be computed between two points [Barachant, 2012].
  - At each point \(C\) (i.e. each covariance matrix), a scalar product can be defined in the associated tangent space

\[
\langle S_1, S_2 \rangle_C = \text{tr}(S_1 C^{-1} S_2 C^{-1}).
\]

- Distance between two SPD matrices (along the geodesic)

\[
d_R(C_i, C_j) = \|\text{logm} \left( C_i^{-1} C_j \right) \|_F
\]
Logarithmic/Exponential map of SPD matrices

- Project locally a covariance matrix $C_p$ onto the tangent plane

$$S_p = \log_C(C_p) = C^{1/2} \logm \left( C^{-1/2} C_p C^{-1/2} \right) C^{1/2}$$
Proposed kernel

- An usual approach consists in mapping data in another feature space (usually with higher dimensionality)
  - Empirical kernel choice
  - Most employed: RBF kernel
    \[ k(x_i, x_j) = \exp[-\gamma \|x_i - x_j\|^2] \]
- Riemannian geometry provides a natural kernel to deal with covariance matrices
  - Mapping function
    \[ \phi(C) = \log_{C_{\text{ref}}}(C) \]
    \[
    k(\text{vec}(C_i), \text{vec}(C_j)) = \langle \phi(C_i), \phi(C_j) \rangle_{C_{\text{ref}}}
    = \text{tr} \left[ \logm \left( C_{\text{ref}}^{-1/2} C_i C_{\text{ref}}^{-1/2} \right) \logm \left( C_{\text{ref}}^{-1/2} C_j C_{\text{ref}}^{-1/2} \right) \right].
    \]
- Application to SVM classification
SVM classification

- Linear (separable) two-class SVM
  - Supervised classification with a set of labelled feature vectors \( \{(x_p, y_p)\}_{p=1}^P \)
  - Seeks to linearly separate data by finding an hyperplane maximising the margin \( M \)

\[
h(x) = b + w^T x = 0
\]
\[
h(x^+) = +1
\]
\[
h(x^-) = -1
\]
\[
M = 2M_h = \frac{2}{\|w\|}
\]

- Decision function

\[
h(x) = b + \sum_{p=1}^P \alpha_p y_p \langle x_p, x \rangle
\]

non-zero for support vectors

Cost function:

\[
\text{argmin}_{w, b} \frac{1}{2} \|w\|^2 \text{ s.c. } y_p h(x_p) \geq 1
\]
SVM (kernel trick)

- **Kernel trick:**
  - Map data in a new feature space (where hopefully separable)
  - The mapping function is rarely expressed, the kernel function is key in the computation

\[ k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle_H \]

- **Decision function:**

\[ h(x) = b + \sum_{p=1}^{P} \alpha_p y_p \langle \phi(x_p), \phi(x) \rangle_H \]

- Use new kernel based on Riemannian geometry
Choice of free parameter $C_{\text{ref}}$

- $C_{\text{ref}} :=$ Point in SPD space where the tangent plane is computed

- @ Identity (log-Euclidean kernel)
  \[
  C_{\text{ref}} = I_E
  \]
  \[
  k(\text{vec}(C_i), \text{vec}(C_j)) = \text{tr} \left[ \logm(C_i) \logm(C_j) \right]
  = \langle \logm(C_i), \logm(C_j) \rangle_F
  \]

- @ geometric mean of the P labeled covariance matrices
  \[
  C_{\text{ref}} = \arg\min_C \sum_{p=1}^{P} d_R^2(C, C_p)
  \]
Experiments

- **Asynchronous MI-based BCI** (BCI competition IV, dataset 2a)
  - 9 subjects
  - 22 electrodes
  - Reference electrode on the left mastoid
  - 8-35 Hz (general) band-pass filter
  - 4-class dataset: RH, LH, TO, BF (144 trials per class)

- **Objectives**:
  - Average performance across subjects and across all pairs of binary classification
  - Performance comparison
    - Standard CSP method
    - SVM applied on vectorized covariance matrices
    - Covariance kernel-SVM @ identity
    - Covariance kernel-SVM @ geometric mean
  - 30-fold cross-validation
Results (1/3)

<table>
<thead>
<tr>
<th></th>
<th>SVM kernel</th>
<th>SVM kernel</th>
<th>SVM plain</th>
<th>CSP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Geometric Mean</td>
<td>Identity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LH/RH</td>
<td>83.4</td>
<td>82.8</td>
<td>76.9</td>
<td>81.4</td>
</tr>
<tr>
<td>LH/BF</td>
<td>89.9</td>
<td>89.5</td>
<td>82.2</td>
<td>86.9</td>
</tr>
<tr>
<td>LH/TO</td>
<td>89.3</td>
<td>88.8</td>
<td>84</td>
<td>87.3</td>
</tr>
<tr>
<td>RH/BF</td>
<td>88.8</td>
<td>88.2</td>
<td>81.1</td>
<td>86.3</td>
</tr>
<tr>
<td>RH/TO</td>
<td>88.7</td>
<td>88</td>
<td>83.4</td>
<td>85.7</td>
</tr>
<tr>
<td>BF/TO</td>
<td>82.1</td>
<td>81.4</td>
<td>74.9</td>
<td>80.8</td>
</tr>
<tr>
<td>mean</td>
<td>87</td>
<td>86.4</td>
<td>80.4</td>
<td>84.7</td>
</tr>
</tbody>
</table>

Table 1: Average classification accuracy across the 9 subjects for 6 pairs of mental tasks.

- R-based kernel SVM **outperforms** CSP+LDA in all cases,
- Direct covariance classification gives poor results,
- Geometric mean of the P SPD matrices is a good location to compute the tangent plane.
Individual sessions are shown

RkSVM is superior especially in almost all sessions, especially in difficult situations
Results (3/3)

- Statistical analysis (9 subjects)
  - P-values for the 6 pairs of mental tasks in a subject-independent manner.
  - Hypothesis H0 : $\mu_1 > \mu_2$, one-tailed dependent t-test (8 df) for paired samples
  - RkSVM classification is significantly better in almost all pairs of mental task.
Conclusion

- New covariance kernel for directly handling covariance matrices in classification methods
  - No need for explicit spatial filtering
  - Simple to implement (just add a new kernel in SVM toolbox!)
- New framework of R. geometry in BCI
- Successful application on a BCI competition dataset
  - outperforms significantly the conventional CSP method (two-class)
- Future work
  - Multi-class classification
  - Online application w/ location update of the tangent space between BCI sessions
  - Investigate the minimum # of trials required to properly estimate the classifier
  - Use regularized version of SVM to deal with high-dimensional (E*) features
Merci de votre attention