Optimizing Multi-class Spatio-Spectral Filters via Bayes Error Estimation for EEG Classification

Wenming Zheng\textsuperscript{1} and Zhouchen Lin\textsuperscript{2}

\textsuperscript{1}Southeast University
\textsuperscript{2}Microsoft Research Asia

Dec. 8, 2009
Outline

1 Motivation
   - Background
   - Related Work
   - Our Work

2 Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3 Experiments

4 Conclusions
Outline

1 Motivation
   - Background
   - Related Work
   - Our Work

2 Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3 Experiments

4 Conclusions
EEG Classification

- useful for brain-computer interface (BCI)
- challenging because: very noisy and close class means

An EEG recording cap

EEG signals
EEG Classification

- useful for brain-computer interface (BCI)
- challenging because: very noisy and close class means

An EEG recording cap

EEG signals
Outline

1 Motivation
   - Background
   - Related Work
   - Our Work

2 Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3 Experiments

4 Conclusions
Common Spatial Patterns (CSPs): find directions that are most discriminant for one of the classes:

\[
\omega = \arg \max_{\omega} \max \left\{ \frac{\omega^T \Sigma_1 \omega}{\omega^T \Sigma_2 \omega}, \frac{\omega^T \Sigma_2 \omega}{\omega^T \Sigma_1 \omega} \right\}.
\]

Solve by:

1. whitening \(\Sigma_1 + \Sigma_2\) with \(P\):

\[
P^T (\Sigma_1 + \Sigma_2) P = I,
\]

2. then diagonalizing \(P^T \Sigma_1 P\) with \(Q = [q_1, \ldots, q_d]\):

\[
(PQ)^T \Sigma_1 (PQ) = \text{diag}(\lambda_1, \ldots, \lambda_d),
\]

3. and finally collecting the eigenvectors \(q_1\) and \(q_d\):

\[
W = [\omega_1, \omega_2] = P[q_1, q_d].
\]

Common Spatial Patterns (CSPs): find directions that are most discriminant for one of the classes:

\[ \omega = \arg \max_{\omega} \max \left\{ \frac{\omega^T \Sigma_1 \omega}{\omega^T \Sigma_2 \omega}, \frac{\omega^T \Sigma_2 \omega}{\omega^T \Sigma_1 \omega} \right\}. \]

Solve by:
1. whitening \( \Sigma_1 + \Sigma_2 \) with \( P \):

\[ P^T(\Sigma_1 + \Sigma_2)P = I, \]

2. then diagonalizing \( P^T \Sigma_1 P \) with \( Q = [q_1, \cdots, q_d] \):

\[ (PQ)^T \Sigma_1 (PQ) = \text{diag}(\lambda_1, \cdots, \lambda_d), \]

3. and finally collecting the eigenvectors \( q_1 \) and \( q_d \):

\[ W = [\omega_1, \omega_2] = P[q_1, q_d]. \]


Zheng and Lin MCSSPs for EEG Classification
Common Spatial Patterns (CSPs): find directions that are most discriminant for one of the classes:

\[ \omega = \arg \max_{\omega} \max \left\{ \frac{\omega^T \Sigma_1 \omega}{\omega^T \Sigma_2 \omega}, \frac{\omega^T \Sigma_2 \omega}{\omega^T \Sigma_1 \omega} \right\}. \]

Solve by:

1. whitening \( \Sigma_1 + \Sigma_2 \) with \( P \):

\[ P^T (\Sigma_1 + \Sigma_2) P = I, \]

2. then diagonalizing \( P^T \Sigma_1 P \) with \( Q = [q_1, \cdots, q_d] \):

\[ (PQ)^T \Sigma_1 (PQ) = \text{diag}(\lambda_1, \cdots, \lambda_d), \]

3. and finally collecting the eigenvectors \( q_1 \) and \( q_d \):

\[ W = [\omega_1, \omega_2] = P[q_1, q_d]. \]

Common Spatial Patterns (CSPs): find directions that are most discriminant for one of the classes:

\[
\omega = \arg \max_{\omega} \max \left\{ \frac{\omega^T \Sigma_1 \omega}{\omega^T \Sigma_2 \omega}, \frac{\omega^T \Sigma_2 \omega}{\omega^T \Sigma_1 \omega} \right\}.
\]

Solve by:
1. whitening \( \Sigma_1 + \Sigma_2 \) with \( P \):
   \[
P^T (\Sigma_1 + \Sigma_2) P = I,
   \]
2. then diagonalizing \( P^T \Sigma_1 P \) with \( Q = [q_1, \ldots, q_d] \):
   \[
   (PQ)^T \Sigma_1 (PQ) = \text{diag}(\lambda_1, \ldots, \lambda_d),
   \]
3. and finally collecting the eigenvectors \( q_1 \) and \( q_d \):
   \[
   W = [\omega_1, \omega_2] = P[q_1, q_d].
   \]

Common Spatial Patterns (CSPs): find directions that are most discriminant for one of the classes:

\[ \omega = \arg \max_{\omega} \max \left\{ \frac{\omega^T \Sigma_1 \omega}{\omega^T \Sigma_2 \omega}, \frac{\omega^T \Sigma_2 \omega}{\omega^T \Sigma_1 \omega} \right\}. \]

Solve by:
1. whitening \( \Sigma_1 + \Sigma_2 \) with \( P \):
   \[ P^T (\Sigma_1 + \Sigma_2) P = I, \]
2. then diagonalizing \( P^T \Sigma_1 P \) with \( Q = [q_1, \cdots, q_d] \):
   \[ (PQ)^T \Sigma_1 (PQ) = \text{diag}(\lambda_1, \cdots, \lambda_d), \]
3. and finally collecting the eigenvectors \( q_1 \) and \( q_d \):
   \[ W = [\omega_1, \omega_2] = P[q_1, q_d]. \]


Zheng and Lin. MCSSPs for EEG Classification
Feature extraction:
1. $Y_j = \omega_j^T X$;  
2. $f = [f_1, f_2]^T$, where  
$$f_j = \log \left( \frac{\text{var}(Y_j)}{\sum_k \text{var}(Y_k)} \right).$$

This is inherited by all the subsequent methods.
Related work - CSPs (2)

Feature extraction:
1. $Y_j = \omega_j^T X$;
2. $f = [f_1, f_2]^T$, where

$$f_j = \log \left( \frac{\text{var}(Y_j)}{\sum_k \text{var}(Y_k)} \right).$$

This is inherited by all the subsequent methods.
Feature extraction:
1. $Y_j = \omega_j^T X$;
2. $f = [f_1, f_2]^T$, where

$$f_j = \log \left( \frac{\text{var}(Y_j)}{\sum_k \text{var}(Y_k)} \right).$$

This is inherited by all the subsequent methods.
Common Spatial Patterns (CSPs): find $W$ that

$$W^T \Sigma_i W = D_i, \quad (i = 1, 2).$$

Multiclass Common Spatial Patterns (MCSPs): find $W$ that

$$W^T \Sigma_i W \approx D_i, \quad (i = 1, \ldots, c)$$

Solve by:
1. Joint Approximate Diagonalization (JAD) technique for $W$;
2. Choose the columns of $W$ with highest “mutual information”.

Related work - MCSPs [2,3]

- **Common Spatial Patterns (CSPs):** find $W$ that
  \[ W^T \Sigma_i W = D_i, \quad (i = 1, 2). \]

- **Multiclass Common Spatial Patterns (MCSPs):** find $W$ that
  \[ W^T \Sigma_i W \approx D_i, \quad (i = 1, \cdots, c) \]

Solve by:
1. Joint Approximate Diagonalization (JAD) technique for $W$;
2. Choose the columns of $W$ with highest “mutual information”.

---

Common Spatio-Spectral Patterns (CSSPs): further utilize the temporal information.

Define

$$\delta^\tau(x_{i,j}^t) = x_{i,j-\tau}^t, \quad \hat{x}_i^t = \begin{pmatrix} x_i^t \\ \delta^\tau(x_i^t) \end{pmatrix}.$$  

Then process $\hat{x}_i^t$ using the CSPs method.

Common Spatio-Spectral Patterns (CSSPs): further utilize the temporal information.

Define

$$\delta^\tau (x_{i,j}^t) = x_{i,j-\tau}^t,$$

$$\hat{X}_i^t = \begin{pmatrix} x_{i,j}^t \\ \delta^\tau (x_{i,j}^t) \end{pmatrix}.$$

Then process $\hat{X}_i^t$ using the CSPs method.

Outline

1 Motivation
   - Background
   - Related Work
   - Our Work

2 Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3 Experiments

4 Conclusions
Multiclass Common Spatio-Spectral Patterns (MCSSPs): generalize CSSPs to multiclass case.
Why not JAD technique?

JAD is based on

\[ \varepsilon(\omega) \leq 1 - 2^{I(c, \omega^T x) - H(c)}. \]

No closed-form solution for the mutual information. A lot of approximations follow.

So we aim at deducing our method via minimizing the classification error directly.
Outline

1 Motivation
   - Background
   - Related Work
   - Our Work

2 Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3 Experiments

4 Conclusions
Two classes case - Bhattacharyya bound

\[ \varepsilon_{ij} = \int \min \{ P_i p_i(x), P_j p_j(x) \} \, dx. \]

Suppose

\[ p_i(x) = N(0, \Sigma_i), \quad p_j(x) = N(0, \Sigma_j), \]

then the Bhattacharyya bound is

\[ \varepsilon_{ij} \leq \sqrt{P_i P_j} \left( \frac{|\Sigma_{ij}|}{\sqrt{|\Sigma_i||\Sigma_j|}} \right)^{-\frac{1}{2}}, \]

where

\[ \Sigma_{ij} = \frac{1}{2} (\Sigma_i + \Sigma_j). \]
Suppose

\[ p_i(x) = N(0, \Sigma_i), \quad p_j(x) = N(0, \Sigma_j), \]

then the Bhattacharyya bound is

\[ \varepsilon_{ij} \leq \sqrt{P_i P_j} \left( \frac{\bar{\Sigma}_{ij}}{\sqrt{||\Sigma_i|| \cdot ||\Sigma_j||}} \right)^{-\frac{1}{2}}, \]

where

\[ \bar{\Sigma}_{ij} = \frac{1}{2} (\Sigma_i + \Sigma_j). \]
Two classes case - Bhattacharyya bound

\[ \varepsilon_{ij} = \int \min \{ P_i p_i(x), P_j p_j(x) \} \, dx. \]

Suppose

\[ p_i(x) = N(0, \Sigma_i), \quad p_j(x) = N(0, \Sigma_j), \]

then the Bhattacharyya bound is

\[ \varepsilon_{ij} \leq \sqrt{P_i P_j} \left( \frac{|\bar{\Sigma}_{ij}|}{\sqrt{\Sigma_i \Sigma_j}} \right)^{-\frac{1}{2}}, \]

where

\[ \bar{\Sigma}_{ij} = \frac{1}{2}(\Sigma_i + \Sigma_j). \]
If we project the samples to 1D by a vector $\omega$, then the distributions of the projected samples become:

$$\tilde{p}_i(x) = N(0, \omega^T \Sigma_i \omega), \quad \tilde{p}_j(x) = N(0, \omega^T \Sigma_j \omega),$$

and the upper bound of $\varepsilon_{ij}$ becomes:

$$\varepsilon_{ij}(\omega) \leq \sqrt{P_i P_j} \left( \frac{\omega^T \Sigma_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}}.$$
If we project the samples to 1D by a vector $\omega$, then the distributions of the projected samples become:

$$\tilde{p}_i(x) = N(0, \omega^T \Sigma_i \omega), \quad \tilde{p}_j(x) = N(0, \omega^T \Sigma_j \omega),$$

and the upper bound of $\varepsilon_{ij}$ becomes:

$$\varepsilon_{ij}(\omega) \leq \sqrt{P_i P_j} \left( \frac{\omega^T \tilde{\Sigma}_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}}.$$
Bound of total error

\[ \varepsilon_{ij}(\omega) \leq \sqrt{P_i P_j} \left( \frac{\omega^T \tilde{\Sigma}_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}}. \]

For \( c \)-class problems,

\[ \varepsilon \leq \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} \varepsilon_{ij}. \]

Difficult!
Bound of total error

\[ \varepsilon_{ij}(\omega) \leq \sqrt{P_i P_j} \left( \frac{\omega^T \bar{\Sigma}_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}}. \]

For \(c\)-class problems,

\[ \varepsilon \leq \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} \varepsilon_{ij}. \]

Difficult!
Fisher-like criterion:

\[ \varepsilon(\omega) \leq C_1 - C_2 \frac{\omega^T A \omega}{\omega^T B \omega}. \]
Simplification

Using inequalities

\[
\left( \frac{\omega^T \bar{\Sigma}_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}} \leq 1 - \frac{1}{4} \left( \frac{\omega^T \Delta \Sigma_{ij} \omega}{\omega^T \bar{\Sigma}_{ij} \omega} \right)^2 ;
\]

\[
\sum_i \left( \frac{a_i}{b_i} \right)^2 \geq \left( \frac{\sum_i a_i}{\sum_i b_i} \right)^2 , \quad \forall a_i \geq 0; b_i > 0 ;
\]

\[
|\omega^T (\Sigma_i - \bar{\Sigma}) \omega| \leq P \sum_{i=1}^c |\omega^T (\Sigma_i - \Sigma_j) \omega| ,
\]

where \( \Delta \Sigma_{ij} = \frac{1}{2} (\Sigma_i - \Sigma_j) \), \( \bar{\Sigma} = \sum_{i=1}^c P_i \Sigma_i \), and assuming equal prior probability: \( P_i = P = 1/c \), we can obtain

\[
\varepsilon(\omega) \leq c(c - 1) P^2 - \frac{1}{8} P^3 \left( \frac{\sum_{i=1}^c |\omega^T (\Sigma_i - \bar{\Sigma}) \omega|}{2 \omega^T \bar{\Sigma} \omega} \right)^2 .
\]
Simplification

Using inequalities

\[
\left( \frac{\omega^T \tilde{\Sigma}_{ij} \omega}{\sqrt{(\omega^T \Sigma_i \omega)(\omega^T \Sigma_j \omega)}} \right)^{-\frac{1}{2}} \leq 1 - \frac{1}{4} \left( \frac{\omega^T \Delta \Sigma_{ij} \omega}{\omega^T \tilde{\Sigma}_{ij} \omega} \right)^2 ;
\]

\[
\sum_i \left( \frac{a_i}{b_i} \right)^2 \geq \left( \frac{\sum_i a_i}{\sum_i b_i} \right)^2 , \quad \forall a_i \geq 0; b_i > 0 ;
\]

\[
|\omega^T (\Sigma_i - \tilde{\Sigma}) \omega| \leq P \sum_{i=1}^c |\omega^T (\Sigma_i - \Sigma_j) \omega| ,
\]

where \( \Delta \Sigma_{ij} = \frac{1}{2}(\Sigma_i - \Sigma_j) \), \( \tilde{\Sigma} = \sum_{i=1}^c P_i \Sigma_i \), and assuming equal prior probability: \( P_i = P = 1/c \), we can obtain

\[
\varepsilon(\omega) \leq c(c - 1)P/2 - \frac{1}{8} P^3 \left( \sum_{i=1}^c |\omega^T (\Sigma_i - \tilde{\Sigma}) \omega| \right)^2 .
\]
Outline

1 Motivation
   • Background
   • Related Work
   • Our Work

2 Theory
   • Error Bound
   • Our MCSSPs Filters
   • Solution Method

3 Experiments

4 Conclusions
Resulting criterion

\[ \omega = \arg \max J(\omega), \]

where

\[ J(\omega) = \sum_{i=1}^{c} \frac{|\omega^T (\Sigma_i - \bar{\Sigma}) \omega|}{\omega^T \bar{\Sigma} \omega}. \]
Optimal discriminant vectors

\[ J(\omega) = \sum_{i=1}^{c} \frac{|\omega^T (\Sigma_i - \bar{\Sigma}) \omega|}{\omega^T \bar{\Sigma} \omega}. \]

\[ \omega_1 = \arg \max_{\omega} J(\omega), \]
\[ \ldots \]
\[ \omega_k = \arg \max_{\omega^T \bar{\Sigma} \omega_j = 0, j = 1, \ldots, k-1} J(\omega). \]
Outline

1. Motivation
   - Background
   - Related Work
   - Our Work

2. Theory
   - Error Bound
   - Our MCSSPs Filters
   - Solution Method

3. Experiments

4. Conclusions
Do transform: \( \hat{\Sigma}_i = \bar{\Sigma}^{-\frac{1}{2}} \Sigma_i \bar{\Sigma}^{-\frac{1}{2}} \), \( \alpha = \bar{\Sigma}^{\frac{1}{2}} \omega \) to change the problem to

\[
\begin{align*}
\alpha_1 & = \arg \max_{\alpha} \hat{J}(\alpha), \\
\vdots \\
\alpha_k & = \arg \max_{\alpha^T \mathbf{U}_{k-1} = 0} \hat{J}(\alpha),
\end{align*}
\]

where \( \hat{J}(\alpha) = \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| \), \( \mathbf{U}_{k-1} = [\alpha_1, \cdots, \alpha_{k-1}] \).
Introduce the sign vector $s_i \in \{+1, -1\}$ and define

$$T(s) = \sum_{i=1}^{c} s_i (\hat{\Sigma}_i - I).$$

If the signs are correct, then

$$\alpha^T T(s) \alpha = \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \hat{J}(\alpha).$$

And we have:

$$\max_{\|\alpha\|=1} \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \max_{s \in S} \max_{\|\alpha\|=1} \alpha^T T(s) \alpha.$$
Sketch of solution - Introducing Sign Vector

Introduce the sign vector $s_i \in \{+1, -1\}$ and define

$$T(s) = \sum_{i=1}^{c} s_i (\hat{\Sigma}_i - I).$$

If the signs are correct, then

$$\alpha^T T(s) \alpha = \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \hat{J}(\alpha).$$

And we have:

$$\max_{\|\alpha\| = 1} \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \max_{s \in S} \max_{\|\alpha\| = 1} \alpha^T T(s) \alpha.$$
Sketch of solution - Introducing Sign Vector

Introduce the sign vector $s_i \in \{+1, -1\}$ and define

$$T(s) = \sum_{i=1}^{c} s_i (\hat{\Sigma}_i - I).$$

If the signs are correct, then

$$\alpha^T T(s) \alpha = \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \hat{J}(\alpha).$$

And we have:

$$\max_{\|\alpha\|=1} \sum_{i=1}^{c} |\alpha^T (\hat{\Sigma}_i - I) \alpha| = \max_{s \in S} \max_{\|\alpha\|=1} \alpha^T T(s) \alpha.$$
Theorem 0 (Improving the sign vector).

Suppose that $\alpha^{(i)}$ is the principal eigenvector of $T(s_i), i = 1, 2$, and $s_2$ is defined as

$$(s_2)_i = \text{sign}((\alpha^{(1)})^T(\hat{\Sigma}_i - I)\alpha^{(1)}).$$

Then $(\alpha^{(2)})^T T(s_2) \alpha^{(2)} \geq (\alpha^{(1)})^T T(s_1) \alpha^{(1)}$. 
Sketch of solution - Rank-One-Update Technique

\[ \mathbf{U}_k = [\alpha_1, \cdots, \alpha_k]. \]

**Theorem 1 (Solution of \( \alpha_k \)).**

Let \( \mathbf{Q}_k \mathbf{R}_k \) be the QR decomposition of \( \mathbf{U}_k \). Then \( \alpha_{k+1} \) is the principal eigenvector of

\[ (\mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T) \mathbf{T}(\mathbf{s})(\mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T). \]

**Theorem 2 (Update \( \mathbf{Q}_k \)).**

Suppose that \( \mathbf{Q}_k \mathbf{R}_k \) is the QR decomposition of \( \mathbf{U}_k \). Then

\[
\begin{pmatrix}
\mathbf{Q}_k & \frac{\mathbf{q}}{\|\mathbf{q}\|}
\end{pmatrix}
\begin{pmatrix}
\mathbf{R}_k & \mathbf{Q}_k^T \alpha_{k+1} \\
0 & \frac{\|\mathbf{q}\|}{\|\mathbf{q}\|}
\end{pmatrix}
\]

is the QR decomposition of \( \mathbf{U}_{k+1} \), where \( \mathbf{q} = \alpha_{k+1} - \mathbf{Q}_k (\mathbf{Q}_k^T \alpha_{k+1}). \)

\[ \mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T = \prod_{i=1}^{k} (\mathbf{I}_d - \mathbf{q}_i \mathbf{q}_i^T) = (\mathbf{I}_d - \mathbf{Q}_{k-1} \mathbf{Q}_{k-1}^T)(\mathbf{I}_d - \mathbf{q}_k \mathbf{q}_k^T). \]
### Sketch of solution - Rank-One-Update Technique

\[ \mathbf{U}_k = [\alpha_1, \cdots, \alpha_k]. \]

#### Theorem 1 (Solution of \( \alpha_k \)).

Let \( \mathbf{Q}_k \mathbf{R}_k \) be the QR decomposition of \( \mathbf{U}_k \). Then \( \alpha_{k+1} \) is the principal eigenvector of

\[ (\mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T) \mathbf{T}(\mathbf{s})(\mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T). \]

#### Theorem 2 (Update \( \mathbf{Q}_k \)).

Suppose that \( \mathbf{Q}_k \mathbf{R}_k \) is the QR decomposition of \( \mathbf{U}_k \). Then

\[
\begin{pmatrix}
\mathbf{Q}_k & \mathbf{q} / \| \mathbf{q} \| \\
0 & \| \mathbf{q} \|
\end{pmatrix}
\begin{pmatrix}
\mathbf{R}_k & \mathbf{Q}_k \alpha_{k+1}^T \\
0 & \| \mathbf{q} \|
\end{pmatrix}
\]

is the QR decomposition of \( \mathbf{U}_{k+1} \), where \( \mathbf{q} = \alpha_{k+1} - \mathbf{Q}_k (\mathbf{Q}_k \alpha_{k+1}). \)

\[ \mathbf{I}_d - \mathbf{Q}_k \mathbf{Q}_k^T = \prod_{i=1}^{k} (\mathbf{I}_d - \mathbf{q}_i \mathbf{q}_i^T) = (\mathbf{I}_d - \mathbf{Q}_{k-1} \mathbf{Q}_{k-1}^T)(\mathbf{I}_d - \mathbf{q}_k \mathbf{q}_k^T). \]

Zheng and Lin: MCSSPs for EEG Classification
Sketch of solution - Rank-One-Update Technique

\( \mathbf{U}_k = [\alpha_1, \cdots, \alpha_k] \).

**Theorem 1 (Solution of \( \alpha_k \)).**

Let \( \mathbf{Q}_k \mathbf{R}_k \) be the QR decomposition of \( \mathbf{U}_k \). Then \( \alpha_{k+1} \) is the principal eigenvector of

\[
(I_d - \mathbf{Q}_k \mathbf{Q}_k^T) \mathbf{T}(s)(I_d - \mathbf{Q}_k \mathbf{Q}_k^T).
\]

**Theorem 2 (Update \( \mathbf{Q}_k \)).**

Suppose that \( \mathbf{Q}_k \mathbf{R}_k \) is the QR decomposition of \( \mathbf{U}_k \). Then

\[
\begin{pmatrix}
\mathbf{Q}_k & \mathbf{q} \\
\mathbf{q}^T
\end{pmatrix}
\begin{pmatrix}
\mathbf{R}_k & \mathbf{Q}_k^T \alpha_{k+1} \\
0 & \|\mathbf{q}\|
\end{pmatrix}
\]

is the QR decomposition of \( \mathbf{U}_{k+1} \), where \( \mathbf{q} = \alpha_{k+1} - \mathbf{Q}_k (\mathbf{Q}_k^T \alpha_{k+1}) \).

\[
I_d - \mathbf{Q}_k \mathbf{Q}_k^T = \prod_{i=1}^{k} (I_d - \mathbf{q}_i \mathbf{q}_i^T) = (I_d - \mathbf{Q}_{k-1} \mathbf{Q}_{k-1}^T) (I_d - \mathbf{q}_k \mathbf{q}_k^T).
\]
Experiments (1)

Data set: BCI competition 2005 data set IIIa; subjects: k3b, k6b, and l1b; classes: imagery left/right/up/down motions of left hand, right hand, foot, tongue; channels: 60; trials per class: 60. Two-fold cross validation; five rounds of experiments; 10 recognition rates in total. Classifier: 7-NN

Table: Comparison of the classification rates (%) versus standard deviations (%) between MCSPs and MCSSPs.

<table>
<thead>
<tr>
<th>Subject</th>
<th>MCSPs [2]</th>
<th>MCSPs [3]</th>
<th>MCSSPs/Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>k3b</td>
<td>46.17 (6.15)</td>
<td>84.89 (2.74)</td>
<td>85.83 (2.23)</td>
</tr>
<tr>
<td>k6b</td>
<td>33.54 (4.27)</td>
<td>50.09 (2.59)</td>
<td>56.28 (3.87)</td>
</tr>
<tr>
<td>l1b</td>
<td>35.17 (3.92)</td>
<td>62.08 (3.99)</td>
<td>68.58 (6.16)</td>
</tr>
</tbody>
</table>


Figure: The classification rates (%) of our MCSSPs method with different choices of $\tau$. 
Conclusions

- Proposed a multiclass common spatio-spectral patterns (MCSSPs) method for EEG classification.
- Used the Bayes error bound to guide us to find the optimization criterion.
- Experimental results testified to the superiority of our MCSSPs method.
- More elaborate treatments are possible to produce even better performance.

A more general and rigorous theory, for Gaussian mixture models and non-identical class means, is under review.
Conclusions

- Proposed a multiclass common spatio-spectral patterns (MCSSPs) method for EEG classification.
- Used the Bayes error bound to guide us to find the optimization criterion.
- Experimental results testified to the superiority of our MCSSPs method.
- More elaborate treatments are possible to produce even better performance.

A more general and rigorous theory, for Gaussian mixture models and non-identical class class means, is under review.
For questions that I could not answer, please contact the first author Wenming Zheng (wenming.zheng@seu.edu.cn).