**RC24648 (W0809-094) September 23, 2008**
**Computer Science**

# IBM Research Report

## Classification in High-Dimensional Spaces Using Markov Random Field Models with Application to fMRI Analysis

**Avishy Carmi**
Cambridge University

**Dimitri Kanevsky, Bhuvana Ramabhadran**
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

**Research Division**
**Almaden - Austin - Beijing - Cambridge - Haifa - India - T. J. Watson - Tokyo - Zurich**

**Abstract**

We present a new classification algorithm for high dimensional problems. The algorithm uses a Markov random field for modeling meaningful interactions within the training data set. The model parameters are efficiently estimated using the Kalman filter algorithm and adapted to fit the test data using a recursive matrix formulation of the extended Baum-Welch algorithm. A spatially likelihood test procedure is then used for classifying the data. The performance of the new algorithm is demonstrated in fMRI classification.

# Problem Formulation

Let the data set $X \in \mathbb{R}^m$ (of which $m$ is very large) be associated with some response $Y \in \mathscr{Y}$ where $\mathscr{Y}$ is some real-valued discrete space (i.e., classes). We are aimed at predicting $Y$, the class associated with $X$. For simplicity, we assume here only two classes $\mathscr{Y} = \{0, 1\}$. The extension of the algorithm to arbitrary number of classes is straightforward. The assumption underlying the following derivations is that both $X$ and $Y$ are real-valued random variables of which the joint probability density function (pdf) $p(X, Y)$ exists.

# Dimensionality Reduction

In what follows we derive a feature [1] extraction mechanization for reducing the computational complexity of the algorithm. We find a subset $\bar{X} \in \mathbb{R}^n$ of $X$ where $n << m$ by excluding all data points which are insignificant in terms of forming the response $Y$. This is accomplished by either approximating the cross correlation (CCR) or the mutual information (MI) of $X$ and $Y$.

### CCR Mapping

The CCR of the $i$th element $X_i \in X$ and $Y$ is given by

$$\rho_{X_i,Y} = \frac{E\{(X_i - \mu_{X_i})(Y - \mu_Y)\}}{(E\{X_i\}^2 - \mu_{X_i}^2)^{1/2}(E\{Y\}^2 - \mu_Y^2)^{1/2}} \tag{1}$$

Given $k$ training samples $X_{train} = \{X(1), ...X(k)\}$ with known responses $Y_{train} = \{Y(1), ...Y(k)\}$, Eq. (1) is approximated by

$$\hat{\rho}_{X_i,Y} = \frac{k\sum_{j=1}^{k} X_i(j)Y(j) - \sum_{j=1}^{k} X_i(j)\sum_{j=1}^{k} Y(j)}{(k\sum_j X_i(j)^2 - (\sum_j X_i(j))^2)^{1/2}(k\sum_j Y(j)^2 - (\sum_j Y(j))^2)^{1/2}} \tag{2}$$

The reduced set $\bar{X}$ is then obtained as

$$\bar{X} = \{X_i \mid \hat{\rho}_{X_i,Y} \geq \rho_{Th}\} \tag{3}$$

where $\rho_{Th} > 0$ is some predetermined threshold value.

The CCR method is usually inadequate for representing non-linear relations. This shortcoming can be alleviated by resorting to MI-based method.

### MI Mapping

The MI of $X_i$ and $Y$ is given as

$$\begin{aligned} I(X_i, Y) &= \sum_Y \sum_{X_i \in X_{train}} p(X_i, Y) \log\left[\frac{p(X_i, Y)}{p(X_i)p(Y)}\right] \\ &= \sum_Y \sum_{X_i \in X_{train}} p(X_i \mid Y)p(Y) \log\left[\frac{p(X_i \mid Y)}{p(X_i)}\right] = \sum_{Y=0,1} p(Y) \sum_{X_i \in X_{train}} p(X_i \mid Y) \log\left[\frac{p(X_i \mid Y)}{p(X_i)}\right] \end{aligned} \tag{4}$$

Assuming $p(Y = 0) = p(Y = 1) = 1/2$ (i.e., balanced training set), and

$$p(X_i \mid Y) = \mathcal{N}\left(X_i - \mu_{X_i|Y}, \sigma^2_{X_i|Y}\right) \tag{5a}$$

$$p(X_i) = \mathcal{N}\left(X_i - \mu_{X_i}, \sigma^2_{X_i}\right) \tag{5b}$$

The statistics of the Gaussian pdfs above can be approximated using the training data samples as

$$\mu_{X_i|Y} = \frac{1}{k_Y}\sum_{j=1}^{k_Y} X_i^Y(j) \tag{6a}$$

[1] Feature refers to a single data point

$$\sigma^2_{X_i|Y} = \frac{1}{k_Y - 1}\sum_{j=1}^{k_Y}(X_i^Y(j) - \mu_{X_i|Y})^2 \tag{6b}$$

$$\mu_{X_i} = \frac{1}{2}\mu_{X_i|Y=0} + \frac{1}{2}\mu_{X_i|Y=1} \tag{6c}$$

$$\sigma^2_{X_i} = \frac{k_0 - 1}{k-1}\sigma^2_{X_i|Y=0} + \frac{k_1 - 1}{k-1}\sigma^2_{X_i|Y=1} \tag{6d}$$

where

$$X_i^a(j) = \{X_i \in X(j) \cap Y(j) = a\} \tag{7}$$

and $k_Y$ denotes the number of training samples of class $Y$. Substituting the above in Eq. (4) yields

$$I(X_i, Y) = \frac{1}{2}\sum_{\theta=0,1}\sum_{j=1}^{k} C_\theta^i \exp\left\{-\frac{1}{2}\frac{(X_i(j) - \mu_{X_i|Y=\theta})^2}{\sigma^2_{X_i|Y=\theta}}\right\} \times \left[\log C_\theta^i - \frac{1}{2}\frac{(X_i(j) - \mu_{X_i|Y=\theta})^2}{\sigma^2_{X_i|Y=\theta}} - \log \bar{C}^i + \frac{1}{2}\frac{(X_i(j) - \mu_{X_i})^2}{\sigma^2_{X_i}}\right] \tag{8}$$

where $C_\theta^i$ and $\bar{C}^i$ are normalization constants

$$C_\theta^i = \left[\sum_{j=1}^{k}\exp\left\{-1/2\frac{(X_i(j) - \mu_{X_i|Y=\theta})^2}{\sigma^2_{X_i|Y=\theta}}\right\}\right]^{-1} \tag{9a}$$

$$\bar{C}^i = \left[\sum_{j=1}^{k}\exp\left\{-1/2\frac{(X_i(j) - \mu_{X_i})^2}{\sigma^2_{X_i}}\right\}\right]^{-1} \tag{9b}$$

The reduced set $\bar{X}$ is then obtained as

$$\bar{X} = \{X_i \mid I(X_i, Y) \geq I_{Th}\} \tag{10}$$

where $I_{Th} > 0$ is some predetermined threshold value.

## Connectivity Modeling

In this section we derive the pattern recognition algorithm which forms the core of the classification method. The algorithm is based on modeling statistical connections within the reduced data set $\bar{X}$.

Let $G(X, e)$ be an undirected graph, where $e$ is an edge representing statistical dependency. Let us assume that $G(X, e)$ is fully connected, i.e., $X_i \in \bar{X}$ is connected to his neighbors $G_i = \{\bar{X}\}/\{X_i\}$. For every class $Y = \theta$ we define a parametric functional relation of the form

$$\varphi_\theta(X_i, G_i, W_i) = 0 \tag{11}$$

where $W_i \sim p_{W_i}(\cdot)$ is a noise random variable representing uncertainty. Now, suppose that we can express the following relation

$$W_i = \varphi_\theta^{-1}(X_i, G_i) \tag{12}$$

then it easily follows that

$$p(X_i \mid G_i, \theta) = p_{W_i}(\varphi_\theta^{-1}(X_i, G_i))\det\left(\nabla_{X_i}\varphi_\theta^{-1}(X_i, G_i)\right) \tag{13}$$

is the pdf describing statistical relation between $X_i$ and $G_i$ for a given class $Y = \theta$. The above formulation describes a Markov random field (MRF) of which the joint pdf is given by

$$p(\bar{X} \mid Y = \theta) = \frac{1}{Z_\theta}\prod_{i=1}^{n} p(X_i \mid G_i, \theta), \qquad X_i \in \bar{X} \tag{14}$$

where $n$ denotes the total number of nodes in $\bar{X}$. The normalizing constant $Z_\theta$ is given as

$$Z_\theta = \sum_{\bar{X}\in\Omega}\prod_{i=1(X_i\in\bar{X})}^{n} p(X_i \mid G_i, \theta) \tag{15}$$

## The Estimated Class

The predicted class is taken as the one with the highest probability $p(\bar{X}_{test} \mid Y = \theta)$, where $X_{test}$ denotes the test data set. In the binary case, one has to compare

$$p(\bar{X}_{test} \mid Y = 0) \underset{<}{\overset{>}{}} p(\bar{X}_{test} \mid Y = 1) \tag{16}$$

or equivalently

$$\frac{p(\bar{X}_{test} \mid Y = 0)}{p(\bar{X}_{test} \mid Y = 1)} \underset{<}{\overset{>}{}} 1 \tag{17}$$

Further defining

$$l := \prod_{i=1}^{n} \frac{p(X_i \mid G_i, \theta = 0)}{p(X_i \mid G_i, \theta = 1)} \tag{18}$$

and

$$c := \log \frac{Z_0}{Z_1} \tag{19}$$

yields an equivalent test to (16)

$$\log l = \sum_{i=1}^{n} \log \frac{p(X_i \mid G_i, \theta = 0)}{p(X_i \mid G_i, \theta = 1)} \underset{<}{\overset{>}{}} c \tag{20}$$

where the constant $c$ (which rarely can be computed straightforwardly) can be tuned using either training or development data sets (see appendix). Using Eq. (20), the predicted class is obtained as

$$\hat{Y} = \begin{cases} 0, & \log l > c \\ 1, & \log l < c \end{cases} \tag{21}$$

## Ergodic Sums

If the following conditions hold

- The uncertainty random variables $W_i$, $\forall i$ are independent and identically distributed (iid).
- The Jacobian $\nabla_{X_i} \varphi_\theta^{-1}(X_i, G_i)$ is independent of $\theta$.

then

$$l = \prod_{i=1}^{n} \frac{p_W(\varphi_{\theta=0}^{-1}(X_i, G_i))}{p_W(\varphi_{\theta=1}^{-1}(X_i, G_i))} \underset{<}{\overset{>}{}} \exp\{c\} \tag{22}$$

can be interpreted as a "spatial" likelihood ratio test where nodes act as samples. It can be shown (using the strong ergodic theorem or the strong law of large numbers) that in this case

$$\lim_{n \to \infty} l = \begin{cases} +\infty, & \text{if } \theta = 0 \text{ is the true class} \\ 0, & \text{if } \theta = 1 \text{ is the true class} \end{cases} \tag{23}$$

The above argumentation implies that regardless of the value of $c$ the test yields the correct class for some $n > n'$, the number of nodes in the MRF model.

## Convergence to a True Class

It has been pointed out that the accuracy (i.e., convergence to the correct class) depends on the value of $c$ and the number of nodes $n$. Under the conditions previously mentioned the strong law of large numbers (SLLN) yields

$$\lim_{n \to \infty} \frac{1}{n} \log l = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \log \frac{p(X_i \mid G_i, \theta = 0)}{p(X_i \mid G_i, \theta = 1)} = \alpha \tag{24}$$

where

$$\alpha := \begin{cases} -KL\left\{p(\cdot \mid \cdot, \theta = 0) \parallel (p(\cdot \mid \cdot, \theta = 1)\right\}, & \text{if } \theta = 0 \text{ is the true class} \\ KL\left\{p(\cdot \mid \cdot, \theta = 1) \parallel (p(\cdot \mid \cdot, \theta = 0)\right\}, & \text{if } \theta = 1 \text{ is the true class} \end{cases} \tag{25}$$

and $KL\{p_1 \parallel p_2\}$ denotes the Kullback-Leibler divergence between the pdfs $p_1$ and $p_2$. Note that the definition (25) implies $\alpha < 0$ if the true class is $\theta = 0$ and $\alpha > 0$ if the true class is $\theta = 1$. According to the central limit theorem

$$\zeta = (\frac{1}{n}\log l - \alpha) \sim \mathcal{N}(0, O(1/n)) \tag{26}$$

assuming large enough $n$. Thus,

$$\frac{1}{n}\log l = \alpha + \zeta, \qquad \zeta = O(1/n^{1/2}) \tag{27}$$

or, equivalently

$$l = \exp\{n\alpha\}\exp\{O(1/n^{1/2})\} \tag{28}$$

Eqs. (22) and (28) imply

$$\exp(n\alpha)\exp(O(1/n^{1/2})) \gtrless \exp(c) \tag{29}$$

yielding

$$\exp(n\alpha) \gtrless \exp(c - O(1/n^{1/2}) \tag{30}$$

and

$$\alpha \gtrless \frac{c}{n} - O(1/n^{3/2}) \tag{31}$$

The above clearly shows that the effect of $c$ diminishes as $n \to \infty$. Moreover, the accuracy depends on $c$, $n$ and $\alpha$, the expected discrimination information of one class over the other.

## The Linear Gaussian Case

In this work we assume linear connections of the form

$$X_i = \left(\beta_i^\theta\right)^T G_i + W_i, \qquad X_i \in \bar{X} \tag{32}$$

where $G_i \in R^{n-1}$ and $\beta_i^\theta \in R^{n-1}$, $i \in [1, n]$. Following this, the conditional pdf $p(X_i \mid G_i, \theta)$ can be expressed by means of the pdf of $W_i$ as

$$p(X_i \mid G_i, \theta) = p_{W_i}\left(X_i - \left(\beta_i^\theta\right)^T G_i\right) \tag{33}$$

In practice, the random parameter vector associated with the class $\theta$, $\beta_i^\theta$, is estimated using the training data set. Let $\hat{\beta}_i^\theta$ be an estimator of $\beta_i^\theta$, then

$$\beta_i^\theta = \hat{\beta}_i^\theta + \tilde{\beta}_i^\theta \tag{34}$$

where $\tilde{\beta}_i^\theta$ is the estimation error. Substituting (34) into (32) gives

$$X_i = \left(\hat{\beta}_i^\theta + \tilde{\beta}_i^\theta\right)^T G_i + W_i \tag{35}$$

Further defining

$$\zeta_i^\theta := \left(\tilde{\beta}_i^\theta\right)^T G_i + W_i \tag{36}$$

yields

$$X_i = \left(\hat{\beta}_i^\theta\right)^T G_i + \zeta_i^\theta \tag{37}$$

which is similar to (32) with the only difference of $\beta_i^\theta$ replaced by its estimate. The conditional pdf $p(X_i \mid G_i, \theta)$ can now be expressed in terms of $\hat{\beta}_i^\theta$ instead of the unknown $\beta_i^\theta$, that is

$$p(X_i \mid G_i, \theta) = p_{\zeta_i^\theta}\left(X_i - \left(\hat{\beta}_i^\theta\right)^T G_i\right) \tag{38}$$

In what follows we shall see that $\zeta_i^\theta$ represents the innovation noise in the Kalman filtering formulation. This sequence has some well-known statistical properties which are described in.

# MRF Training via Kalman Filtering

In this work we use the Kalman filter (KF) algorithm for training the MRF models of every class in a computationally efficient manner. The KF estimates the parameters $\beta_i^\theta$, $i \in [1, n]$ sequentially using the training samples thereby allowing significant reduction of computational load.

Suppose that there are $k_\theta$ training samples for class $Y = \theta$, and let $X_{train}^\theta := \left\{\bar{X}(1), \ldots, \bar{X}(k_\theta)\right\}$ be the set of these samples. The KF is the best linear estimator in the minimum mean square error (MMSE) sense [1], that is

$$\hat{\beta}_i^\theta = \arg\min_{\hat{\beta}_i^\theta} E\left\{\left(\beta_i^\theta - \hat{\beta}_i^\theta\right)^T \left(\beta_i^\theta - \hat{\beta}_i^\theta\right)\right\} \tag{39}$$

which coincides with the general solution, $\hat{\beta}_i^\theta = E\left\{\beta_i^\theta | X_{train}^\theta\right\}$, in the linear Gaussian case (i.e., linear Gaussian connections).

Taking (32) as the measurement equation while assuming $W_i \sim \mathcal{N}(0, I)$ yields the following KF recursion which is identical to the recursive least-squares algorithm.

Initialization:

$$P_0 = \gamma^{-1} I,\ \left(\hat{\beta}_i^\theta\right)_0 = 0, \quad \gamma << 1 \tag{40}$$

Measurement update:

$$K_k = P_k G_i(k) \left[G_i(k) P_k G_i(k)^T + I\right]^{-1} \tag{41a}$$

$$\left(\hat{\beta}_i^\theta\right)_{k+1} = \left(\hat{\beta}_i^\theta\right)_k + K_k \left[X_i(k) - G_i(k)^T \left(\hat{\beta}_i^\theta\right)_k\right] \tag{41b}$$

$$P_{k+1} = \left(I - K_k G_i(k)^T\right) P_k \tag{41c}$$

It should be noted that the KF is used here for parameter estimation rather than state estimation. However, if the training samples are obtained from time-series then the conventional KF algorithm, which includes a time-propagation stage, may be more adequate. In its form above, the KF is aimed at minimizing the following objective function

$$\hat{\beta}_i^\theta = \arg\min_{\hat{\beta}_i^\theta} \sum_{j=1}^{k_\theta} \parallel X_i(j) - \left(\hat{\beta}_i^\theta\right)^T G_i(j) \parallel_2 \tag{42}$$

The next stage consists of computing the conditionals $p(X_i \mid G_i, \theta)$ forming the MRF model associated with class $\theta$. For that purpose we need to know the statistics of $\zeta_i^\theta$, the innovation. It is well known from KF theory that $(\zeta_i^\theta)_k$ is a zero-mean white Gaussian sequence [2]

$$(\zeta_i^\theta)_k \sim \mathcal{N}\left(0, G_i(k)^T P_k G_i(k) + I\right) \tag{43}$$

In this work we compute the sample covariance of $\zeta_i^\theta$ as

$$\Sigma_i^\theta = \frac{1}{k_\theta - 1} \sum_{j=1}^{k_\theta} \left[X_i(j) - \left(\hat{\beta}_i^\theta\right)_{k_\theta}^T G_i(j)\right] \left[X_i(j) - \left(\hat{\beta}_i^\theta\right)_{k_\theta}^T G_i(j)\right]^T \tag{44}$$

which in turn yields

$$p(X_i \mid G_i, \theta) = \mathcal{N}\left(X_i - \left(\hat{\beta}_i^\theta\right)_{k_\theta}^T G_i, \Sigma_i^\theta\right) \tag{45}$$

## Generalization of The KF Formulation

The linear connections (32) can be generalized as follows. Consider two sets of nodes $G_i \in R^r$ and $G_j \in R^m$ satisfying the relation

$$G_i = \beta_{ij}^\theta G_j + W_{ij} \tag{46}$$

where $\beta_{ij}^\theta \in R^{r \times m}$.

In order to implement the previously described KF scheme for estimating the matrices $\beta_{ij}^\theta$ we rewrite the above equation as follows

$$G_i = (G_j^T \otimes I_{r \times r}) \bar{\beta}_{ij}^\theta + W_{ij} \tag{47}$$

[2] The innovations process is non-stationary.

where

$$\bar{\beta}_{ij}^{\theta} := \text{Vec}\left(\beta_{ij}^{\theta}\right) \tag{48}$$

is the vectorized form of $\beta_{ij}^{\theta}$ and $\otimes$ is Kronecker product. The KF can now be applied for estimating $\bar{\beta}_{ij}^{\theta}$ using (47).

### Adaptation

Given the test data set $\bar{X}_{test} = \{\bar{X}(1), \ldots, \bar{X}(k_{test})\}$, we adapt the MRF models of every class using extended Baum-Welch (EBW) iterations as follows (see [2–4])

$$\left(\hat{\beta}_i^{\theta}\right)_{j+1} = \left[I - D_j G_i(j)^T\right]\left(\hat{\beta}_i^{\theta}\right)_j + D_j X_i(j) \tag{49}$$

where $D_j$ is some tuning matrix which can be set as the Kalman gain matrix, $K_j$ (where $j$ denotes the test sample index), to ensure convergence. Finally, the sample covariance of $\zeta_i^{\theta}$ is updated as

$$\left(\Sigma_i^{\theta}\right)_{new} = \frac{k_\theta - 1}{k_\theta + k_{test} - 1}\left(\Sigma_i^{\theta}\right)_{old} + \frac{1}{k_\theta + k_{test} - 1}\sum_{j=1}^{k_{test}}\left[X_i(j) - \left(\hat{\beta}_i^{\theta}\right)_{k_{test}+1}^T G_i(j)\right]\left[X_i(j) - \left(\hat{\beta}_i^{\theta}\right)_{k_{test}+1}^T G_i(j)\right]^T \tag{50}$$

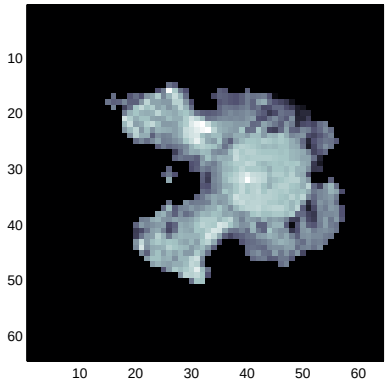
## Application to fMRI Analysis

The new classification algorithm is applied to fMRI analysis. The data $X$ is a vector consisting of $14,043$ elements (voxels). The testing scenario and the fMRI datasets are the ones used in [5]. The total number of samples is 84. In this case $Y$ represents the stimuli response which can take either of the two classes $-1$ or $+1$ (there are exactly 42 samples of each class). The training and testing data sets are obtained using cross validation, that is, at every run two testing samples (one of each class) is taken out of the original set, leaving 82 training samples. This procedure is repeated 84 times. The classification algorithm is tested using Monte Carlo runs in which the original data set, consisting of 84 samples, is randomly permuted. The number of runs varies between 10 to 20.

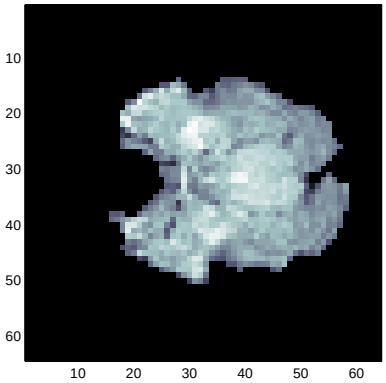
Figure 1 shows various fMRI scans of different brain sections. The corresponding CCR maps of these sections are shown in Fig. 2.

Figure 3 demonstrates the role of the adaptation scheme. In this case, the MRF model consists of only 100 nodes (taken as those with the highest MI) andtThe classification algorithm is tested with and without the adaptation stage. The distributions of the accuracy (the number of correct predictions out of 84 samples) based on 20 Monte Carlo runs of both the adaptive and non-adaptive algorithms are shown in the right and left panels of this figure, respectively. From this figure, it can be clearly recognized that the adaptation stage significantly increases the prediction accuracy.

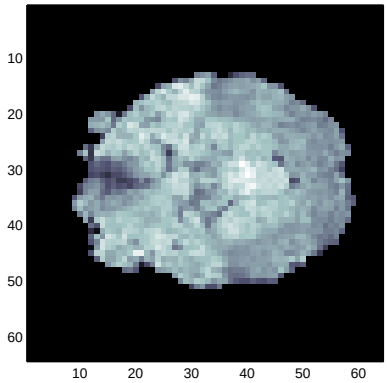
The performance of the algorithm with 307 nodes (taken as those with the highest MI) based on 10 Monte Carlo runs is shown in figures 4 and 5. The tuning constant $c$ is taken as 0 in Fig. 4 yielding mean accuracy of approximately 91 percent. Setting $c$ to its optimal value in this case (see Appendix) increases the mean accuracy to 93 percent.



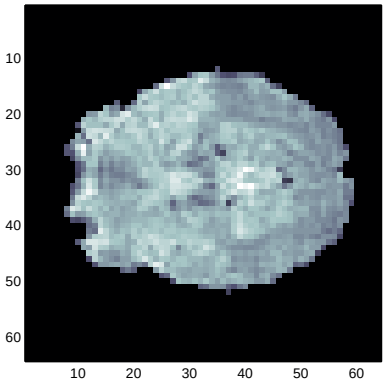

**(a) Section 9**




**(b) Section 10**

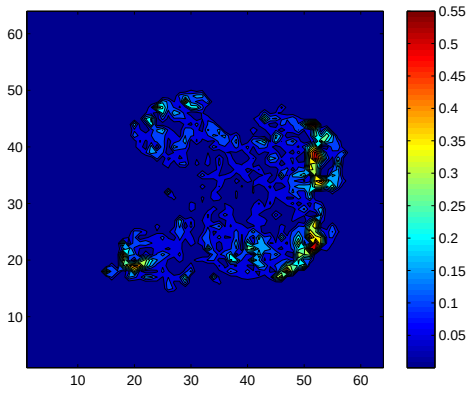



**(c) Section 12**

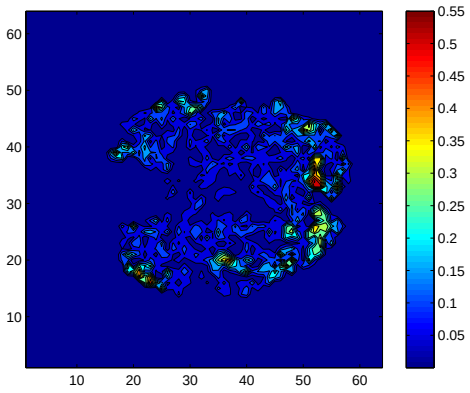



**(d) Section 13**

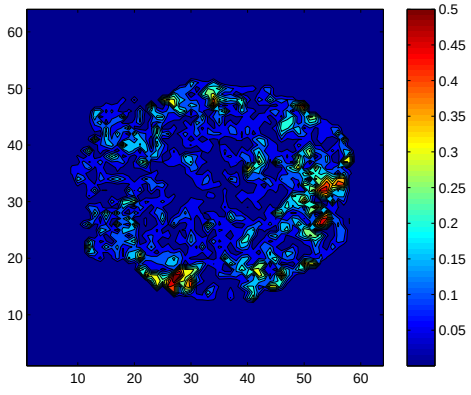
**Figure 1. fMRI scans.**



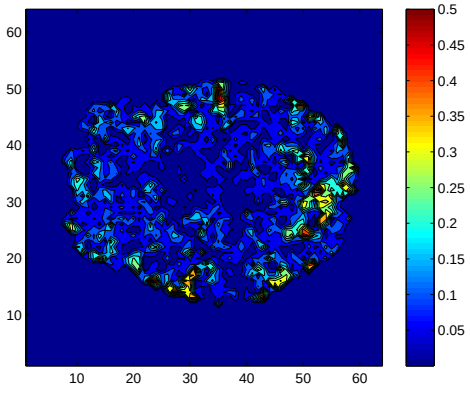

**(a) Section 9**




**(b) Section 10**

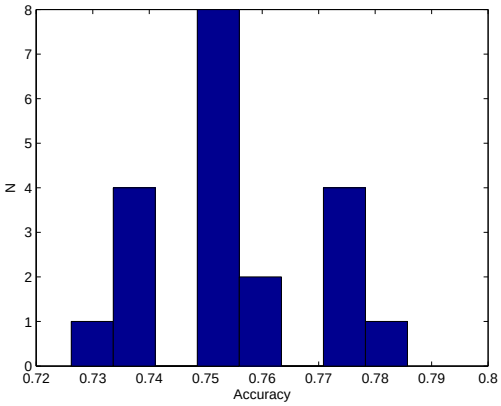



**(c) Section 12**

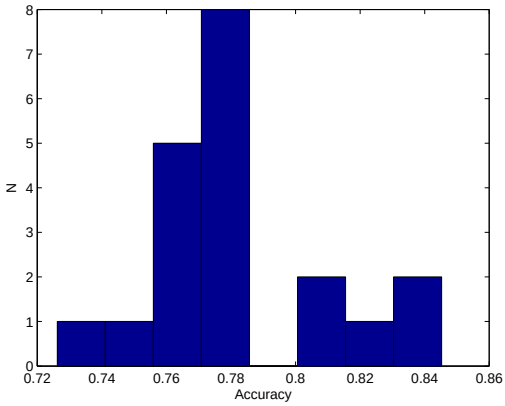



**(d) Section 13**

**Figure 2. Correlation maps.**




**(a) Without adaptation**




**(b) With adaptation**

**Figure 3. The effect of adaptation. MRF consists of 100 nodes. 20 Monte Carlo runs.**

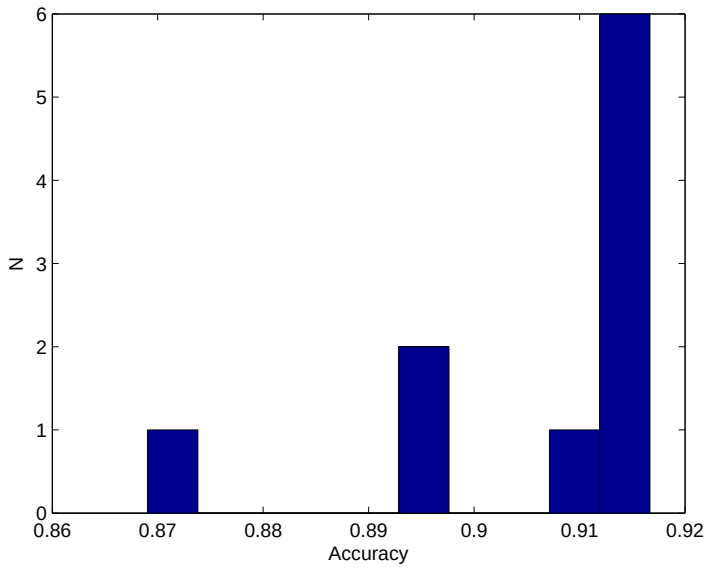



**Figure 4. Distribution of prediction accuracy for $c = 0$. MRF consists of 307 nodes. 10 Monte Carlo runs.**

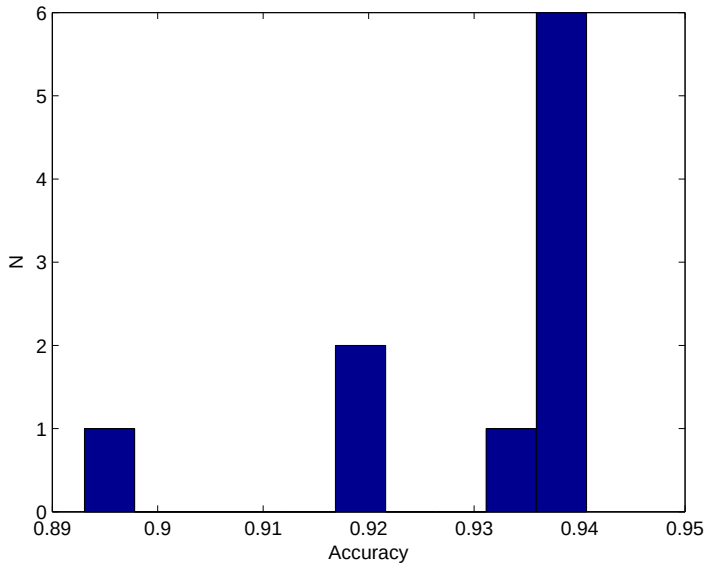



**Figure 5. Distribution of prediction accuracy for optimal $c^* = -71$. MRF consists of 307 nodes. 10 Monte Carlo runs.**

# Appendix A

## Optimal Tuning of $c$

The constant $c$ in (20) can be taken as the one maximizing the accuracy of prediction based on some development dataset. Let $X^{\theta}_{dev} = \{X_\theta(1), \ldots, X_\theta(k_\theta)\}$ be a dataset associated with the class $Y = \theta$, and let also

$$d_\theta(j) := \log p(\bar{X}_\theta(j) \mid Y = 0) - \log p(\bar{X}_\theta(j) \mid Y = 1) \tag{A.1}$$

We are aimed at minimizing the following objective function

$$c^* = \arg\max_c \left[ \eta_1 \sum_{j=1}^{k_1} \mathbf{1}\left(d_{\theta=1}(j) \leq c\right) + \eta_0 \sum_{j=1}^{k_0} \mathbf{1}\left(d_{\theta=0}(j) \geq c\right) \right] \tag{A.2}$$

where $\mathbf{1}(a \in A)$ is the indicator function of the event $a \in A$ (i.e., a function which takes the value 1 if $a \in A$, and takes the value 0 otherwise). The constants $\eta_0$ and $\eta_1$ are the relative counts of both classes, that is, $\eta_0 := k_0/(k_0 + k_1)$ and $\eta_1 := k_1/(k_0 + k_1)$.

# Bibliography


[1] Mendel, J. M., *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, Prentice Hall, 1995.

[2] Gopalakrishnan, P., Kanevsky, D., Nahamoo, D., and Nadas, A., "An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems," *IEEE Trans. Information Theory*, Vol. 37, No. 1, January 1991.

[3] Kanevsky, D., "Extended Baum Transformations For General Functions, II," Tech. Rep. RC23645(W0506-120), Human Language Technologies, IBM, 2005.

[4] Carmi, A. and Kanevsky, D., "Matrix form of Extended Baum Transformations," Tech. Rep. RC, Human Language Technologies, IBM, 2008.

[5] Rish, I., Grabarnik, G., Cecchi, G., Periera, F., and Gordon, G. J., "Closed-Form Supervised Dimensionality Reduction with Generalized Linear Models," The 25th International Conference on Machine Learning, Helsinki, Finland, 2008.