LETTER **Communicated by Eugene Demidenko**

Kernels for Longitudinal Data with Variable Sequence Length and Sampling Intervals

Zhengdong Lu
zhengdong@gmail.com
Microsoft Research Asia, Beijing 100080, P.R.C.

Todd K. Leen
leent@ohsu.edu
Department of Biomedical Engineering, Oregon Health & Science University, Beaverton, OR 97006, U.S.A.

Jeffrey Kaye
kaye@ohsu.edu
Layton Aging & Alzheimer's Disease Center, Oregon Health & Science University, Portland, OR 97239, U.S.A.

We develop several kernel methods for classification of longitudinal data and apply them to detect cognitive decline in the elderly. We first develop mixed-effects models, a type of hierarchical empirical Bayes generative models, for the time series. After demonstrating their utility in likelihood ratio classifiers (and the improvement over standard regression models for such classifiers), we develop novel Fisher kernels based on mixture of mixed-effects models and use them in support vector machine classifiers. The hierarchical generative model allows us to handle variations in sequence length and sampling interval gracefully. We also give nonparametric kernels not based on generative models, but rather on the reproducing kernel Hilbert space. We apply the methods to detecting cognitive decline from longitudinal clinical data on motor and neuropsychological tests. The likelihood ratio classifiers based on the neuropsychological tests perform better than than classifiers based on the motor behavior. Discriminant classifiers performed better than likelihood ratio classifiers for the motor behavior tests.

1 Introduction

Early detection of cognitive decline provides the opportunity for more effective medical intervention, planning for compensation strategies, and assistance (Boise, Morgan, Kaye, & Camicioli, 1999; Gwyther, 2000; Riefler & Larson, 1988). A large body of literature indicates that there are presymptomatic clinical markers of future cognitive decline that can be readily

assessed over time. These studies require a longitudinal cohort design such that baseline or early measures of function are then used to prospectively predict those who will develop mild cognitive impairment (MCI) or dementia. (See section 2 for the clinical definition of MCI.) This body of work consistently indicates that baseline cognitive and motor function assessed up to decades prior to developing dementia is highly predictive of later cognitive decline. Tests assessing cognitive function, such as delayed recall of information (episodic memory), as well as motor function, are predictive (Marquis et al., 2002; Richards, Stern, & Mayeux, 1993; Verghese et al., 2002; Wilson, Schneider, Bienias, Evans, & Bennett, 2003; Howieson et al., 1997; Chen, Ratcliff, Phil, Belle, Cauley, Kosky, et al., 2000).

There is a consistent pattern of change in cognitive and motor domains that occurs presymptomatically, leading to MCI and to dementia. Although these test domains predict group outcomes years later, they are difficult to apply to individual subjects to predict decline with a degree of certainty that is clinically useful. This limitation comes from the analysis methods used as well as the data types employed to date.

In this letter, we address the prediction problem at the individual level as a classification problem. The aim is to predict if an individual will become cognitively impaired based on his or her motor and cognitive test data from longitudinal clinical assessments. The difficulties we face are the extreme sparseness of the observation and the high variability among subjects.

Classification of longitudinal data is a supervised learning problem aimed at labeling (temporal) sequences of variable length and variable sampling intervals. There are two basic methods for such problems. In the first, one builds a generative model for the sequences and uses likelihood ratio or posterior probability to determine class membership (Seymore, McCallum, & Rosenfeld, 1999). In the second, one directly trains a classifier, which requires transforming the sequences into vectors of attributes (or features) (Keogh & Pazzani, 1998). As many have pointed out, classifiers based on generative models and likelihood ratio tests often yield poor performance relative to those obtained by training a discriminant function directly (Jaakkola, Meila, & Jebara, 1999). However, feature extraction is still more art than science, and the performance of discriminants depends heavily on the designer's prior knowledge and the particular heuristics implemented. In a hybrid approach, one extracts discriminative features from a generative model, such as the Fisher kernel proposed by Jaakkola & Haussler (1998).

In this letter, we develop both generative and discriminative methods designed for classifying clinical longitudinal data. Our generative models are the mixed-effects models (Laird & Ware, 1982), a type of empirical Bayes model commonly used in biostatistics for its ability to describe variability among individuals. From these, we build likelihood ratio classifiers. For discriminative-trained classifiers, we use support vector machines with new kernels designed for the longitudinal data.

Our new kernels extend the usual Fisher kernel by exploiting the structure of the mixed-effect model to deal properly with time series of unequal length and variable sampling intervals. The formulation allows the Fisher information (metric) to be explicitly calculated, even when sequence lengths and sampling intervals are variable. This avoids the usual ad hoc replacement of the Fisher information with an identity matrix, and hence we retain the information-geometric invariance enjoyed by the Fisher kernel. Use of the correct Fisher information also improves performance of the support vector machine (SVM) since the Fisher scores receive the proper scaling. We also examine kernels based on parametric and nonparametric feature extraction methods independent of the mixed-effect models. The nonparametric method gives a new distance measure that is potentially useful in a variety of variable-sequence-length problems.

In the next section, we describe the six types of the clinical observations we use. In section 3, we give a brief introduction to the mixed-effect model, together with the fitting result on our data. In section 4, we present the detection results based on the mixed-effect models. In sections 5 and 6, we discuss in detail the discriminative models based on the Fisher kernel extension and other feature extraction routines. In section 7, we present the empirical comparison of the classification algorithms we discuss. Finally, section 8 summarizes the letter and points the direction of future research.

2 Data Description

Our research uses clinical motor behavior and psychometric data from the Oregon Brain Aging Study (OBAS) (Green, Kaye, & Ball, 2000). The cohort consists of 216 subjects—91 males and 125 females. All subjects are normal at entry, and when the data were drawn, 78 of them had developed into mild cognitive impairment (MCI) or worse, while 138 remained cognitively healthy. A subject is diagnosed with MCI if he or she has two consecutive Clinical Dementia Rating (CDR) scores of 0.5 or greater.[1] If the CDR is over 0.5 at least once but never in two consecutive clinical visits, the subjects are tagged *questionable dementia*. We split the subject pool into an impaired group (or class) and a normal group according to their state when the data were drawn from the database. In our current study, we include the questionable dementia subjects with the normal group.

Since we are interested in the prediagnosis prediction, we use only the measurements before a clinical diagnosis of MCI or dementia is made. For a reliable prediction for individual subjects, we consider only subjects with at least four motor measurements before the cut-off date, which reduces the number of qualified subjects to fewer than 150, 46 in the impaired

[1]The CDR takes values 0, 0.5, 1, and 2, where 0 stands for the normal and the other values stand for increasing level of impairment.

Table 1: Description of Data.

Test Type	Description	No. N
Gait speed	The time in seconds the subject takes to walk 9 meters (∼30 feet).	97
Steps	The number of steps the subject takes to walk 9 meters (∼30 feet).	97
TappingD	The number of times the subject can tap the forefinger of his or her dominant hand in 10 seconds (averaged over three trials).	97
TappingN	The number of times the subject can tap the forefinger of his or her nondominant hand in 10 seconds (averaged over three trials).	97
Delayed recall	The number of words (out of 10) a subject can recall 1 minute after the words are read to him or her.	86
Logical memory II	The subject is asked to repeat a story that was told 15 to 20 minutes ago and is graded according to the level of matching between the repeated story and the original one	82

Note: The right-most column gives the number of cognitively normal subjects.

group, and fewer than 100 (varying with the types of measurements) in the normal group. The measurements used include four motor behaviors (gait speed, steps, tappingD, tappingN) and two neuropsychological tests (delayed recall, logical memory II), as described in Table 1. For gait speed, steps, delayed recall, and logical memory II, the readings increase as the subjects age or become impaired, while for tappingD and tappingN, the trend is the opposite.[2]

Figure 1 shows a sample of the gait speed data. In the left panel, we give eight measurements of the test (across 7 years) of one subject who later developed into dementia, with each measurement plotted as a circle and consecutive measurement pairs connected by a line. In the right panel, we plot all the measurements from all 46 subjects in the impaired group. The plot in the right panel is called a spaghetti plot.

Our goal is to use the time-series data prior to a clinical diagnosis to predict whether an individual will become impaired. The evaluation of the prediction model for this requires the ground truth about individuals' final cognitive state. In our study, we group the subjects based on whether they are diagnosed with MCI or dementia when the data were drawn. The labeling is inherently inaccurate since some subjects who are normal when the data are drawn will later develop cognitive impairment. This problem is known as right censoring in survival analysis (Klein & Moeschberger, 2003). In light of this, the classification approaches (and the way they are

[2]Since we are assessing motor as well as cognitive predictors, subjects who develop motor impairment from other causes (for example Parkinson's or arthritis) are not included in the data.

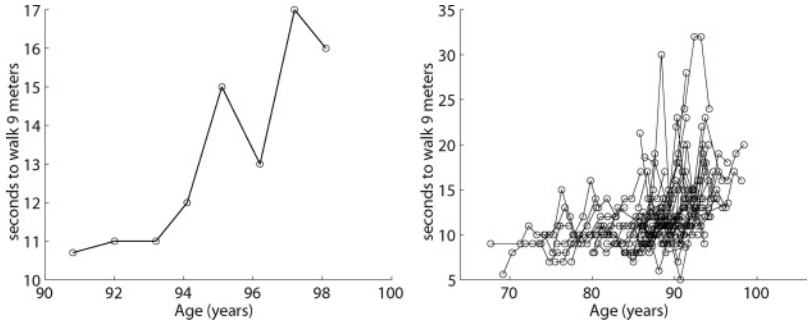

Figure 1: Sample plots of gait speed from the impaired group. (Left): One example subject with eight measurements of the gait speed test (across 7 years) who later developed cognitive impairment. Each measurement is plotted as a circle, and any two consecutive measurements are connected by a line. (Right): Spaghetti plot of seconds measurements from all 46 subjects in the impaired group.

evaluated) we discuss in sections 4 to 6 should be considered as an approximation. We expect that a future extension of our work will enable us to predict the probability that a subject becomes impaired at any future age, the survival function.

3 Mixed-Effect Models

Mixed-effect models provide a flexible and powerful tool for the analysis whenever several measurements are taken from an individual showing consistent differences from the population as a whole. They model both overall population behavior and the variations between individuals. Mixed-effect models have long been used for analyzing longitudinal data (Laird & Ware, 1982; Demidenko, 2004), and are a suitable modeling tool for our longitudinal clinical data. Mixed-effect models provide a principled way to summarize a population of time series (both general behavior and differences between individuals), and thus a means to compare populations. This property is of fundamental importance to our task of discriminating among individuals who will become cognitively impaired and those who will remain normal.

3.1 Regression Models. We confine our attention to parametric regression.[3] Suppose there are k individuals (indexed by $i = 1, \ldots, k$) contributing data to the sample, and we have observations $\{t_n^i, y_n^i\}$, $n = 1, \ldots, N^i$ as

[3]Nonparametric mixed-effect regression is discussed by Guo (2002).

a function of time t for individual i. The data are modeled as

$$y_n^i = f(t_n^i;\, \gamma^i) + \epsilon_n^i, \tag{3.1}$$

where γ^i are the regression parameters and ϵ_n^i is zero-mean white gaussian noise with (unknown) variance σ^2. The superscript on the model parameters γ^i indicates that the generative model is different for each individual contributing to the population. Since the model parameters vary across individuals, it is natural to consider them generated by the sum of a fixed and a variable piece, called the random effect:

$$\gamma^i = \alpha + \beta^i, \tag{3.2}$$

where β^i is assumed distributed $\mathcal{N}(0, \mathbf{D})$ with unknown covariance $\mathbf{D}$. The expected parameter vector α, called the fixed effect or population model, determines the model for the population as a whole. This intuition is most precise for the case in which the model is linear in parameters,

$$f(t;\gamma) = \gamma^T B(t) = \alpha^T B(t) + \beta^T B(t), \tag{3.3}$$

where $B(t) = [B_1(t), B_2(t), \ldots, B_d(t)]^T$ denotes a vector of basis function, for which α gives the average model over individuals.[4] Model fitting uses the entire collection of data $\{\mathbf{t}^i, \mathbf{y}^i\}$, $i = 1, \ldots, k$ to determine the parameters $\mathcal{M} \equiv (\alpha, \mathbf{D}, \sigma^2)$ by maximum likelihood, considering the random effects $\{\beta^i\}$ as latent variables.

3.2 Maximum Likelihood Fitting. The likelihood of the data $\{\mathbf{t}^i, \mathbf{y}^i\}$ given the mixed-effect model $\mathcal{M} = \{\alpha, \mathbf{D}, \sigma^2\}$ is

$$\begin{aligned} p(\mathbf{y}^i;\, \mathbf{t}^i, \mathcal{M}) &= \int p(\mathbf{y}^i \mid \beta^i; \mathbf{t}^i, \sigma)\, p(\beta^i \mid \mathcal{M}) d\beta^i \\ &= (2\pi)^{-N^i/2} |\Sigma^i|^{-1/2} \exp((\mathbf{y}^i - \alpha^T B(\mathbf{t}^i))^T (\Sigma^i)^{-1} \\ &\quad \times (\mathbf{y}^i - \alpha^T B(\mathbf{t}^i))), \end{aligned}$$

where

$$\Sigma^i = \sum_{n=1}^{N^i} B\left(t_n^i\right) \mathbf{D} B\left(\mathbf{t}_n^i\right)^T + \sigma^2 \mathbf{I},$$

$$B(\mathbf{t}^i) = \left[B\left(t_1^i\right), B\left(t_2^i\right), \ldots, B\left(t_n^i\right)\right]^T.$$

[4] More generally, the fixed and random effects can be associated with different basis functions.

The data likelihood for $\mathbf{Y} = \{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^k\}$ with $\mathbf{T} = \{\mathbf{t}^1, \mathbf{t}^2, \ldots, \mathbf{t}^k\}$ are then

$$p(\mathbf{Y};\ \mathbf{T}, \mathcal{M}) = \prod_{i=1}^{k} p(\mathbf{y}^i \,|\, \mathbf{t}^i; \mathcal{M}).$$

We use the expectation-maximization algorithm (Dempster, Laird, & Rubin, 1977), with $\{\beta^1, \beta^2, \ldots, \beta^k\}$ considered the latent variables, to find the maximum likelihood values of $\{\alpha, \mathbf{D}, \sigma\}$. The steps are

$$\text{E-step:}\quad Q(\mathcal{M}, \mathcal{M}^g) = E_{\{\beta^i\}}(\log p(\mathbf{Y}, \{\beta^i\}; \mathbf{T}, \mathcal{M})|\mathbf{Y}; \mathbf{T}, \mathcal{M}^g), \tag{3.4}$$

$$\text{M-step:}\quad \mathcal{M} = \arg\max_{\mathcal{M}} Q(\mathcal{M}, \mathcal{M}^g), \tag{3.5}$$

where $\mathcal{M}^g$ stands for the estimation of the mixed-effect model obtained in previous step and the expectation in the E-step is with respect to the posterior distribution of on $\{\beta^i\}$ when $\mathbf{Y}$ is known and the model parameter is $\mathcal{M}^g$. For the linear mixed-effect model in equation 2.3, the M-step has a closed form:

$$\alpha = \left(\sum_{i=1}^{k} B(\mathbf{t}^i)^T B(\mathbf{t}^i)\right)^{-1} \sum_{i=1}^{k}\sum_{n=1}^{N^i} \left(\mathbf{y}_n^i - E(\beta^i|\mathbf{y}^i, ; \mathbf{t}^i, \mathcal{M}^g)^T B(\mathbf{t}^i)\right), \tag{3.6}$$

$$\mathbf{D} = \frac{1}{k}\sum_{i=1}^{k} E(\beta^i(\beta^i)^T|\mathbf{y}^i; \mathbf{t}^i, \mathcal{M}^g), \tag{3.7}$$

$$\sigma^2 = \frac{1}{\sum_{i=1}^{k} N^i}\sum_{i=1}^{k} E(||\epsilon^i||^2 \,|\, \mathbf{y}^i; \mathbf{t}^i, \mathcal{M}^g). \tag{3.8}$$

The calculation of the expectations in equations 2.6 to 2.8, performed in the E-step, are straightforward since they are all expectation of linear or quadratic function of gaussian variables. Several methods for fitting mixed-effect models are given in the seminal literature (Laird & Ware, 1982; Laird, Lange, & Stram, 1987).

3.3 Mixed-Effect Models on OBAS Data. In this section, we present the mixed-effect models fit by maximum likelihood. We use the linear mixed-effect model with order 1 polynomial basis functions $B(t) = [1, t]^T$.[5] We trained the mixed-effect model on all six measurements. For the four motor behavior measurements, we use the logarithm of measurement as the predicted variable to improve symmetry of the residuals.

[5]Order 2 basis functions provided better generative models but worse classification performance, and so are omitted.

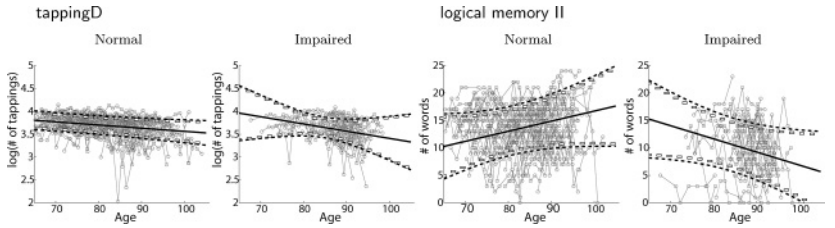

Figure 2: The fit mixed-effect models for tappingD and logical memory II. The linear mixed model with basis function $B(t)$ taken to be a linear polynomial in t. In each panel, the solid black line is the population (fixed effect) model $\alpha^T B(t)$. The two boxed curves are $\alpha^T B(t) \pm \sqrt{B(t)^T \mathbf{D} B(t)}$ (the population model $\pm$ the deviation due to the random effects β carrying the variation between individuals). The black dashed line contains the deviations from both the random effects and the observation noise ϵ (i.e., $\alpha^T B(t) \pm \sqrt{B(t)^T \mathbf{D} B(t) + \sigma^2}$).

Figure 2 shows the mixed-effect models for tappingD and logic memory II. The plots show the fixed-effect regression $\alpha^T B(t)$ (solid curve), the expected standard deviation from the random effect, which carries the variability between subjects (boxed curves), and the independent measurement noise (dashed curve; see the caption). Clearly for tappingD, the fixed-effect model for the impaired group decreases faster than the one for the normal group, and the variance from the random effect in the impaired group is larger than in the normal group. For logic memory II, the difference between models of impaired group and normal group is obvious. The figure shows that the random effects, capturing the variability between individuals, account for a large proportion of the variation from the fixed effect curve; the observation noise accounts for very little of that variability. A standard regression model would not have the random-effect terms and would ascribe all of the large variation away from the fixed-effect curves to noise. This would lower the discrimination of classifiers (as the data would appear noisier than it is when fit with a more appropriate model).

3.4 Mixture of Mixed-Effect Models. A population may consist of several subpopulations with different characteristics. Indeed, as shown in section 3.3, the motor ability of individuals destined to become cognitively impaired declines more dramatically than in individuals who remain cognitively healthy (Camicioli, Howieson, Oken, Sexton, & Kaye, 1998; Marquis et al., 2002). It is sensible to describe the population with people from the two groups with a mixture of two mixed-effect models[6]: one fit on the

[6]It is straightforward to construct such a mixture with more than two components.

normal group (denoted $\mathcal{M}_0$) and one fit on impaired group (denoted $\mathcal{M}_1$), with

$$\mathcal{M}_m = \{\alpha_m, \mathbf{D}_m, \sigma_m\}, \quad m = 0, 1.$$

Here, we use $\widetilde{\mathcal{M}} = \{\pi_0, \mathcal{M}_0, \pi_1, \mathcal{M}_1\}$ to denote the parameters of this mixture, where π_0 and π_1 are the mixing proportions (prior) estimated from the individuals in the training data. Let $z^i \in \{0, 1\}$ be the mixture latent variable, with 1 indicating $\mathbf{y}^i$ generated by the impaired component and 0 the normal component. The generative process consists of three steps:

1. Set the value of z^i as in $\{0, 1\}$ with probability π_0 and π_1. (This chooses the generating component from the mixture.)
2. Draw γ^i from the gaussian distribution $\mathcal{N}(\alpha_{z^i}, \mathbf{D}_{z^i})$, where α_{z^i} and $\mathbf{D}_{z^i}$ are, respectively, the fixed effect and the covariance of the random effect in model $\mathcal{M}_{z^i}$.
3. Let $y_n^i = (\gamma^i)^T B(t_n^i) + \epsilon_n^i$, where ϵ_n^i is drawn from $\mathcal{N}(0, \sigma_{z^i}^2)$.

4 Detecting Cognitive Decline

Our long-term goal is to use motor and cognitive test data to reliably predict cognitive decline in the individual. Ultimately such predictions would encompass an estimate of the time horizon to a clinical diagnosis, or the time horizon to decline to more severe impairment for a mildly impaired individual. Our aim here is to predict whether an individual will become impaired. We consider several solutions to this classification problem. In this section, we use likelihood ratio tests to build a classifier based on the mixed-effect models. In sections 5 and 6, we discuss classifiers based on discriminative methods.

4.1 Likelihood Ratio Classifier Based on Mixed-Effect Model. Let us consider the mixture of mixed-effect model $\widetilde{\mathcal{M}} = \{\pi_0, \mathcal{M}_0, \pi_1, \mathcal{M}_1\}$ discussed in section 3.4. For any given observation $(\mathbf{t}, \mathbf{y})$, the posterior probability that this observation is generated from $\mathcal{M}_0$ is given by

$$\begin{aligned} P(z = 0|\mathbf{y}; \mathbf{t}, \widetilde{\mathcal{M}}) &= \frac{\pi_0 p(\mathbf{y}; \mathbf{t}, \mathcal{M}_0)}{p(\mathbf{y}; \mathbf{t}, \widetilde{\mathcal{M}})} \\ &= \frac{\pi_0 p(\mathbf{y}; \mathbf{t}, \mathcal{M}_0)}{\pi_0 p(\mathbf{y}; \mathbf{t}, \mathcal{M}_0) + \pi_1 p(\mathbf{y}; \mathbf{t}, \mathcal{M}_1)}, \end{aligned} \tag{4.1}$$

where to get $p(\mathbf{y}; \mathbf{t}, \mathcal{M}_m)$, we need to integrate out the random-effect parameter β; that is,

$$\begin{aligned} p(\mathbf{y}; \mathbf{t}, \mathcal{M}_m) &= \int_{\mathbb{R}^d} p(\mathbf{y}; \mathbf{t}, \alpha_m + \beta) p(\beta; \mathcal{M}_m)\, d\beta \\ &= (2\pi)^{-n/2} |\Sigma_m|^{-1/2} \exp((\mathbf{y} - \alpha_m^T B)^T (\Sigma_m)^{-1} (\mathbf{y} - \alpha_m^T B)), \end{aligned}$$

where

$$\Sigma_m = \sum_{n=1}^{N} B(t_n)\mathbf{D}_m B(t_n)^T + \sigma_m^2 \mathbf{I}_{n\times n},$$
$$\mathbf{B} = [B(t_1), B(t_2), \ldots, B(t_n)]^T.$$

The classification decision can be made based on the posterior probability, the group index given by

$$c = \begin{cases} 0 & P(z=0|\mathbf{t}, \mathbf{y}; \widetilde{\mathcal{M}}) \geq 0.5 \\ 1 & \text{otherwise} \end{cases}. \tag{4.2}$$

Equation 4.2 is the optimal Bayesian classifier that minimizes the expected the 0-1 loss,

$$P(z=0|\mathbf{t}, \mathbf{y}; \widetilde{\mathcal{M}})I(c \neq 0) + P(z=1|\mathbf{t}, \mathbf{y}; \widetilde{\mathcal{M}})I(c \neq 1), \tag{4.3}$$

where $I(\cdot)$ is a function with Boolean input and binary output:

$$I(\omega) = \begin{cases} 1 & \omega \text{ is true} \\ 0 & \text{otherwise} \end{cases}.$$

If misclassification of each group carries a different cost, we instead minimize

$$C_I\, P(z=0|\mathbf{t}, \mathbf{y}; \widetilde{\mathcal{M}})I(c \neq 0) + C_N\, P(z=1|\mathbf{t}, \mathbf{y}; \widetilde{\mathcal{M}})I(c \neq 1), \tag{4.4}$$

where C_N is the cost of misclassifying a normal individual and C_I is the cost of misclassifying an impaired individual. The optimal classifier is given by

$$c = \begin{cases} 0 & \dfrac{p(\mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_0)}{p(\mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_1)} \geq \dfrac{\pi_1}{\pi_0}\dfrac{C_N}{C_I} \\ 1 & \text{otherwise} \end{cases}, \tag{4.5}$$

where $\frac{p(\mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_0)}{p(\mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_1)}$ is referred to as a likelihood ratio.

We define the detection rate as the fraction of impaired subjects who are correctly identified and the false alarm rate as the fraction of normal subjects who are incorrectly identified as impaired. The graph of detection rate as a function of the false alarm rate is the receiver operating characteristic (ROC) curve (Pepe, 2003), which we use to evaluate and compare classifiers.

4.2 Evaluation of Classifiers. The performance of a classifier is assessed by the area under the ROC curve (AUC), which can be empirically estimated (Pepe, 2003) by

$$\mathrm{AUC} = \frac{1}{k_I k_N} \sum_{i=1}^{k_I} \sum_{j=1}^{k_H} \left\{ I\left(Y_I^i > Y_N^j\right) + \frac{1}{2} I\left(Y_N^i = Y_I^j\right) \right\}, \tag{4.6}$$

where Y_I^i is the classifier output for subject i in impaired group and Y_N^j is the classifier output for subject j in the normal group. To compare two classifiers A and B, we calculate the difference between the two corresponding AUCs:

$$\Delta\mathrm{AUC} = \mathrm{AUC}_A - \mathrm{AUC}_B.$$

We test the null hypothesis $\mathrm{AUC}_A = \mathrm{AUC}_B$ by comparing the $\Delta\mathrm{AUC}/\sqrt{\mathrm{var}\{\Delta\mathrm{AUC}\}}$ with a standard gaussian distribution (Z-test), where var{ΔAUC} is the sample variability, estimated by jack-knife. We will refer to the p-values for significance tests of the statement, "Classifier A is different from classifier B." The estimation of var{ΔAUC} can be found in Pepe (2003).

Throughout the study, we use a leave-one-out cross-validation to evaluate the classifiers. In each validation round, we use the data from $k-1$ subjects to train a classifier, including the mixed-effect models (where used) and the support vector machines described later, and we test the trained classifier on the data from the single left-out subject. We report test classification results averaged over all k validation rounds. The same training test strategy is used with design of Fisher kernel extension in section 5.

4.3 Classification with and Without Random Effects. To demonstrate the discriminative power we gain by including both the random and fixed (population) effects, we consider a model without random effects. This model attributes the large differences among individuals in the same group as observation noise. The maximum-likelihood fitting of this model is a simple least-squares regression. We use $\mathcal{S}_0$ and $\mathcal{S}_1$ to denote the simplified model fit on the normal group and impaired group, with $\mathcal{S}_m = \{\mu_m, s_m\}$, where μ_m are the regression parameters and s_m is the observation noise.[7] Once the models are fit, we can calculate the likelihood of any novel sample (sequence) $\mathbf{y}$ under each model as

$$p(\mathbf{y}; t, \mathcal{S}_m) = (2\pi)^{-n/2} \sigma^{-n} \exp\left(\frac{||\mathbf{y} - \mu_m^T \mathbf{B}||^2}{2s_m^2} \right),$$

[7] Generally $\mu_m \neq \alpha_m$, although in our experiments they are fairly close.

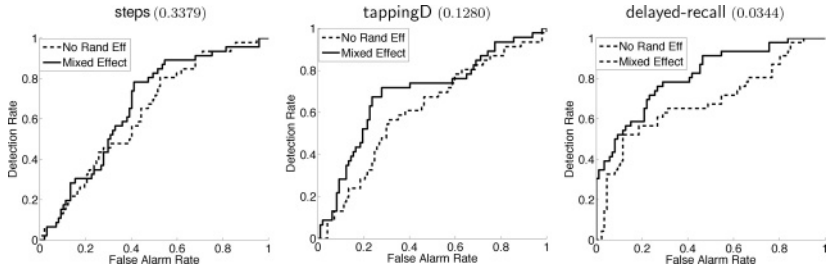

Figure 3: The ROC curves of the likelihood ratio classifiers. The number in the parentheses is the p-value (Z-test) for the null hypothesis: AUC of mixed-effect model is the same as the AUC of the one trained without the random effects.

with $\mathbf{B} = [B(t_1), B(t_2), \ldots, B(t_n)]^T$, based on which we build the likelihood ratio classifier.

We expect that classifiers based on mixed-effect models should outperform this baseline classifier since the random effects describe the systematic variation of an individual sequence from the overall (fixed effect) population curve as a structured and consistent difference rather than as random, uncorrelated noise. The comparison between the two classifiers on steps, tappingD, and delayed recall is in Figure 3. Our result shows that the mixed-effect model is generally better than the simplified model in terms of AUC, but this superiority is statistically significant (at the 0.05 level) only on the delayed recall.

4.4 Discriminative Methods. Direct discriminative methods often perform better than likelihood ratio tests. One reason, discussed by many authors, is that generative models trained by a maximum likelihood (ML) or maximum a posteriori (MAP) criterion may concentrate modeling resources on typical data from each class rather than on parts of the distribution close to the required decision boundary. Generative models trained on each group separately may also expend the resources modeling aspects of the data common to all classes rather than aspects that distinguish the classes (Druck, Pal, Zhu, & McCallum, 2007; Jaakkola, Meila, & Jebara, 1999). Nevertheless, generative models have advantages over discriminative models. They make it easy to incorporate prior knowledge, are better at dealing with missing data than discriminative methods, and classifiers based on them may be less prone to overfitting (Ng & Jordan, 2002; Holub, Welling, & Perona, 2005). Moreover, generative models may contain useful information about the group distribution that is hard to capture with simple discriminative models. As we showed in section 3.3, mixed-effect models trained on the normal and impaired groups separately manifest the difference between the two populations, whereas there is no simple way to do so with a discriminative model.

It is now common practice to take advantage of both types of models, combining generative and discriminative approaches to construct classifiers. In section 4.1, we showed that linear mixed-effect models are a good fit for our longitudinal data, modeling both overall population behavior and individual variations. We will show in section 5 that the mixed-effect model can also be used in feature extraction for discriminative models. Following that, in section 6, we present the performance of discriminative models with distinct feature extraction routines we developed for longitudinal data.

4.5 Optimizing ROC Curves. With a likelihood ratio test, we can get an optimal ROC simply by sweeping the threshold in the test. If the group distributions are accurately modeled, an optimal classification decision will follow from the likelihood ratio test in equation 4.5 no matter the value of the misclassification costs $\{C_N, C_I\}$.

Things are more involved for discriminative methods such as the SVM, where we model the decision boundary instead of the group distribution. Here, the parameters in the classifiers are tuned for one particular value of the misclassification costs, called the design point. All the parameters (rather than just the threshold) may need to be changed to obtain an optimal decision boundary as the misclassification costs change. One can still produce an ROC curve by varying only the threshold of the classifier, but this curve may not be optimal; that is, one might be able to do better by changing the threshold and the other parameters determining the boundary.

With arbitrarily large amounts of training samples and fitting time, we could construct an optimal ROC curve by training a new classifier for each point on the curve (one design point optimized for each value of the false alarm rate). In practice, one can train a small number of classifiers, each at a different design point, and build the ROC by assigning a region of the false alarm axis to each classifier. That is, we can improve the ROC relative to that based on a single classifier trained at a specific design point with only the threshold left free to vary, by concatenating segments of ROCs from different classifiers. Those trained with small $\frac{C_I}{C_N}$ are assigned to the region of the small false alarm rate, and those trained with large $\frac{C_I}{C_N}$ are assigned to the region of the large false alarm. We do this as follows:

1. Train classifiers with several pairs (C_N, C_I), and for each classifier, obtain its ROC curve by varying the threshold h. In the data reported here, we consider only two settings: $C_N = 10, C_I = 10$, and $C_N = 10, C_I = 20$.
2. For each classifier, identify the regime of the false alarm rate in which it performs the best and keep the segment of ROC curve for that regime. Here, since we use only two cost settings, we divide the false alarm axis into two pieces.
3. Concatenate the curve segments from step 2 into a complete curve.

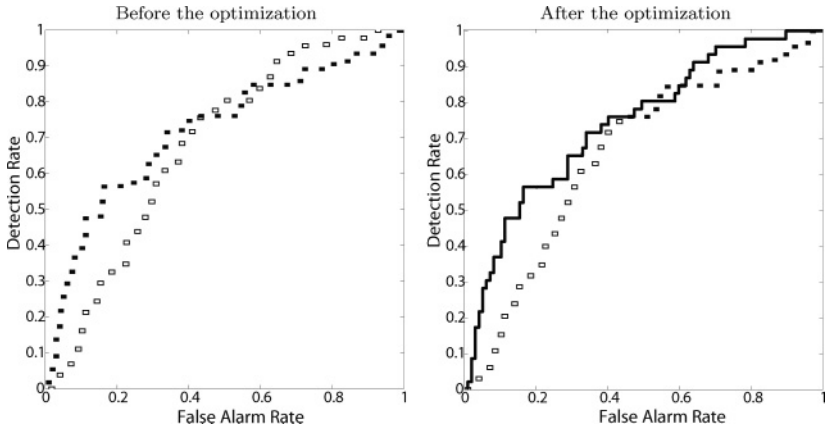

Figure 4: Optimization of the ROC curve over two regions. (Left) The solid boxes indicate the ROC of the classifier trained at $C_I = C_N = 10$, and the white boxes indicate the ROC of the classifier trained at $C_I = 20$, $C_N = 10$. (Right) The solid black line is the composite ROC curve.

A simple example is given in the right panel of Figure 4, in which we concatenate the ROCs for two classifiers shown in the left panel.

5 Improved Fisher Kernels

5.1 Fisher Kernel. The Fisher kernel (Jaakkola & Haussler, 1998) provides a way to extract features from a generative model for use in a discriminative classifier. For any θ-parameterized model $p(x; \theta)$, the Fisher kernel between two points x^i and x^j is

$$K(x^i, x^j) = (\nabla_\theta \log p(x^i; \theta))^T \; \mathbf{I}^{-1} \; \nabla_\theta \log p(x^j; \theta), \tag{5.1}$$

where $\mathbf{I}$ is the Fisher information matrix with (n, m) element:

$$\mathbf{I}_{n,m} = \int_x \frac{\partial \log p(x; \theta)}{\partial \theta_n} \frac{\partial \log p(x; \theta)}{\partial \theta_m} p(x|\theta)\, dx. \tag{5.2}$$

The Fisher kernel entry $K(x^i, x^j)$ is the inner product of the gradient $\nabla\theta p(x; \theta)$ at x^i with that at x^j, with the Fisher information $\mathbf{I}$ as metric. On the manifold of models, the kernel is a scalar invariant, that is, invariant under change of coordinates θ. Jaakkola and Haussler (1998) show that logistic regression with the Fisher kernel returns a classification result at least as good as the likelihood ratio test based on the generative model.

When the data are sequences with differing lengths, and possibly differing sampling times (as for us), the model needs to give the distribution over the sample lengths and times in order to calculate the Fisher information in equation 5.2. Those distributions are not available.[8] In such cases, it is common to replace the Fisher information with the identity matrix, leaving the kernel

$$K(x^i, x^j) = (\nabla_\theta \log p(x^i|\theta))^T \quad \nabla_\theta \log p(x^j|\theta). \tag{5.3}$$

Indeed, this suggestion was made in the original introduction of the Fisher kernel (Jaakkola, Diekhaus, & Haussler, 1999), and it is followed by several researchers (Moreno & Rifkin, 2000; Druck et al., 2007).

This ad hoc simplification raises two problems. First, it singles out the original coordinates θ as special since the metric is defined to be the identity in those coordinates. Further, if the identity matrix is kept as the metric after a coordinate change $\theta \to \theta'$, the invariance of the kernel is ruined. This is a significant issue: the particular coordinate system (parameterization) used to describe the distribution is immaterial. Furthermore, using the identity matrix as a metric discards the proper weighting of the features (Fisher scores) provided by the Fisher information. In our case, discarding this weighting and using the identity matrix leads to reduced classifier performance when the kernel is applied in an SVM.[9]

For us, given the time series $\{\mathbf{t}^i, \mathbf{y}^i\}$, the Fisher score is

$$\phi_{\mathbf{y}^i} = \nabla_{\widetilde{\mathcal{M}}} \log p(\mathbf{y}^i|\mathbf{t}^i, \widetilde{\mathcal{M}}).$$

Due to the difference between the individual observation times, without a distribution on the $\mathbf{t}^i$, we cannot calculate the Fisher information matrix as defined in equation 5.2. As discussed above, usually this problem is circumvented by ad hoc replacing the Fisher information matrix with the identity matrix. This choice is not suitable for us since in mixed-effect models, some parameters can have a vastly different influence on the distribution than others do and the identity metric does not suitably account for this. For our linear mixed-effect models with polynomials as basis functions, the Fisher score will be dominated by the entry associated with the slope and higher-order term. Instead of reweighting the Fisher score entries based on some

[8] While the distribution over sequence lengths could be estimated, data are often too sparse. Certainly for us, the data are far too sparse to estimate the required joint distribution of sequence lengths and sampling times.

[9] Jaakkola and Haussler (1999) show that for probabilistic kernel regression, the problem does not arise when one has unlimited training samples. However for other application of this kernel design, specifically the widely used support vector machine (Vapnik, 1998), this difference cannot be neglected.

heuristic, we propose a principled extension to the Fisher kernel that allows the proper calculation of the information metric.

Our kernel design is based on the generative hierarchy of the mixed-effect models. In this model, the latent variables giving group membership z^i and regression parameters γ^i are drawn from a common distribution (that we fit by maximum likelihood). After that, the sampling times $\mathbf{t}^i$ are drawn at the last step (and we lack distributions for them).

The key idea is to build a Fisher kernel with a proper metric for the latent variables and then project this kernel into the observation space to obtain a kernel between different sequences. We use v^i to denote the latent variables for individual i and $K(v^i, v^j)$ for the Fisher kernel between v^i and v^j. The kernel for $\mathbf{y}^i$ and $\mathbf{y}^j$ is defined as the expectation of $K(v^i, v^j)$ given the observation $\mathbf{y}^i$ and $\mathbf{y}^j$:

$$K(\mathbf{y}^i, \mathbf{y}^j) = E(K(v^i, v^j) | \mathbf{y}^i, \mathbf{y}^j; \mathbf{t}^i, \mathbf{t}^j, \widetilde{\mathcal{M}}) \tag{5.4}$$

$$= \iint K(v^i, v^j) p(v^i | \mathbf{y}^i; \mathbf{t}^i, \widetilde{\mathcal{M}}) p(v^j | \mathbf{y}^j; \mathbf{t}^j, \widetilde{\mathcal{M}}) dv^i dv^j. \tag{5.5}$$

5.2 Possible Design Strategies. Based on the particular choice of latent variable v and the consequent kernel form for $K(v^i, v^j)$, we have several possible design strategies that we make explicit below. This extension to the Fisher kernel enables us to deal with time series with unequal length and differing sampling intervals. This appears to be a novel construction.

5.2.1 Design A. For this design, we take $\{\gamma^i\}$ as the latent variable and marginalize out latent variable $\{z^i\}$. That is, we consider each individual's regression model parameters γ to be drawn from the mixture of gaussian distributions,

$$p(\gamma | \widetilde{\mathcal{M}}) = \pi_0 \; p(\gamma; \alpha_0, \mathbf{D}_0) + \pi_1 \; p(\gamma; \alpha_1, \mathbf{D}_1) \equiv p(\gamma; \tilde{\Theta}),$$

where $\tilde{\Theta} = \{\pi_0, \alpha_0, \mathbf{D}_0, \pi_1, \alpha_1, \mathbf{D}_1\}$ are the parameters of the corresponding gaussian mixture model, and $p(\gamma; \alpha_m, \mathbf{D}_m)$ $(m = 0, 1)$ is simply a gaussian distribution on γ with mean α_m and covariance $\mathbf{D}_m$. This generative process is similar to that in section 3.4 but with the latent variable z^i marginalized out.[10]

The Fisher kernel for γ is

$$K(\gamma^i, \gamma^j) = (\nabla_{\tilde{\Theta}} \log p(\gamma^i | \tilde{\Theta}))^T (\mathbf{I}^\gamma)^{-1} \nabla_{\tilde{\Theta}} \log p(\gamma^i | \tilde{\Theta}), \tag{5.6}$$

[10]Strictly speaking, we cannot integrate out z^i at this step since the group membership is used later in deciding the variance of the observation noise σ_i^2. However, this is a reasonable approximation here since the observation noise specified by $\mathcal{M}_0$ and $\mathcal{M}_1$ has almost the same variance.

where the Fisher score is

$$\nabla_{\tilde{\Theta}} \log p(\gamma^i; \tilde{\Theta}) = \left[\frac{\partial \log p}{\partial \pi_0}; \frac{\partial \log p}{\partial \alpha_0}; \frac{\partial \log p}{\partial \mathbf{D}_0}; \frac{\partial \log p}{\partial \pi_1}; \frac{\partial \log p}{\partial \alpha_1}; \frac{\partial \log p}{\partial \mathbf{D}_1} \right]^T,$$

and the Fisher information matrix $\mathbf{I}^\gamma$ is

$$\mathbf{I}^\gamma_{n,m} = \int_x \frac{\partial \log p(\gamma; \tilde{\Theta})}{\partial \tilde{\Theta}_n} \frac{\partial \log p(\gamma; \tilde{\Theta})}{\partial \tilde{\Theta}_m} p(\gamma|\tilde{\Theta})\, d\gamma. \tag{5.7}$$

Once $K(\gamma^i, \gamma^j)$ is obtained, we can define the kernel between $\mathbf{y}^i$ and $\mathbf{y}^j$ as the expectation of $K(\gamma^i, \gamma^j)$ given $\mathbf{y}^i$ and $\mathbf{y}^j$:

$$\begin{aligned} K(\mathbf{y}^i, \mathbf{y}^j) &= E(K(\gamma^i, \gamma^j) | \mathbf{y}^i, \mathbf{y}^j; \mathbf{t}^i, \mathbf{t}^j, \widetilde{\mathcal{M}}) \\ &= \iint K(\gamma^i, \gamma^j) p(\gamma^i|\mathbf{y}^i; \mathbf{t}^i, \widetilde{\mathcal{M}}) p(\gamma^j|\mathbf{y}^j; \mathbf{t}^j, \widetilde{\mathcal{M}})\ d\gamma^i\, d\gamma^j \\ &= \left(\int \nabla_{\tilde{\Theta}} \log p(\gamma^i|\tilde{\Theta}) p(\gamma^i | \mathbf{y}^i; \mathbf{t}^i, \widetilde{\mathcal{M}})\, d\gamma^i \right)^T (\mathbf{I}^\gamma)^{-1} \\ &\quad \times \int \nabla_{\tilde{\Theta}} \log p(\gamma^j|\tilde{\Theta}) p(\gamma^j | \mathbf{y}^j; \mathbf{t}^j \widetilde{\mathcal{M}} \Big) \, d\gamma^j. \end{aligned} \tag{5.8}$$

The drawback of this design is that the integrals required to evaluate $\mathbf{I}^\gamma$ and

$$\int \nabla_{\tilde{\Theta}} \log p(\gamma^j|\tilde{\Theta}) p(\gamma^j | \mathbf{y}^j; \mathbf{t}^j \widetilde{\mathcal{M}})\, d\gamma^j$$

are generally not tractable. In our experiments, we estimated the integrals by Monte Carlo sampling (Chen, Shao, & Ibrahim, 2000).

5.2.2 Design B. For this design, we use γ^i and z^i jointly as latent variables. The probability model is

$$p(z^i, \gamma^i; \tilde{\Theta}) = \pi_{z^i} p(\gamma_i; \alpha_{z^i}, \mathbf{D}_{z^i}),$$

where $\tilde{\Theta}$ is the same as in design A.

The Fisher score is

$$\nabla_{\tilde{\Theta}} \log p(z^i, \gamma^i; \tilde{\Theta}) = \left[\frac{\partial \log p}{\partial \pi_0}; \frac{\partial \log p}{\partial \alpha_0}; \frac{\partial \log p}{\partial \mathbf{D}_0}; \frac{\partial \log}{\partial \pi_1}; \frac{\partial \log p}{\partial \alpha_1}; \frac{\partial \log p}{\partial \mathbf{D}_1} \right]^T,$$

and the Fisher kernel for the joint variable (γ^i, z^i) is

$$K((z^i, \gamma^i), (z^j, \gamma^j)) = (\nabla_{\tilde{\Theta}} \log p(\gamma^i|\tilde{\Theta}))^T (\mathbf{I}^{z,\gamma})^{-1} \nabla_{\tilde{\Theta}} \log p(\gamma^i|\tilde{\Theta}) \quad , \tag{5.9}$$

where $\mathbf{I}^{z,\gamma}$ is the Fisher information matrix. The kernel for $\mathbf{y}^i$ and $\mathbf{y}^j$ is defined similarly to that in design A:

$$\begin{aligned} K(\mathbf{y}^i, \mathbf{y}^j) &= E_{z^i,\gamma^i,z^j,\gamma^j}\big(K\big((z^i,\gamma^i),(z^j,\gamma^j)\big)\big|\,\mathbf{y}^i,\mathbf{y}^j;\mathbf{t}^i,\mathbf{t}^j\widetilde{\mathcal{M}}\big) \\ &= \int\!\!\int \sum_{z^i}\sum_{z^j} K\big((z^i,\gamma^i),(z^j,\gamma^j)\big)\, p\big(z^i,\gamma^i\,|\,\mathbf{y}^i;\mathbf{t}^i,\widetilde{\mathcal{M}}\big) \\ &\quad \times p\big(z^j,\gamma^j\,|\,\mathbf{y}^j;\mathbf{t}^j,\widetilde{\mathcal{M}}\big)\, d\gamma^i\, d\gamma^j \end{aligned} \tag{5.10}$$

This design is related to the marginalized kernel proposed by Tsuda, Kin, and Asai (2002). Their kernel also uses a distribution with discrete latent variable h (indicating the generating component) and observable x, which form a complete variable $\mathbf{x} = (h, x)$. They define the kernel for observables x^i and x^j as

$$K(x^i, x^j) = \sum_{h^i}\sum_{h^j} P(h^i|x^i)P(h^j|x^j)K(\mathbf{x}^i,\mathbf{x}^j),$$

where $K(\mathbf{x}^i,\mathbf{x}^j)$ is the joint kernel for complete variables. The latter takes the form

$$K(\mathbf{x}^i,\mathbf{x}^j) = \delta(h^i,h^j)K_{h^i}(x^i,x^j), \tag{5.11}$$

where $K_{h^i}(x^i,x^j)$ is the kernel defined for the h^i component generative model. It is clear from equation 5.11 that $K(\mathbf{x}^i,\mathbf{x}^j)$ is 0 if x^i and x^j are generated from different component models (i.e., $h^i \neq h^j$); otherwise, it takes the value of kernel defined for the mth component model if $h^i = h^j = m$.

As an alternative to equation 5.9, we can define a joint kernel for (z^i,γ^i) similar to Tsuda's marginalized kernel with

$$\widetilde{K}((z^i,\gamma^i),(z^j,\gamma^j)) = K_{z^i}(\gamma^i,\gamma^j)\delta(z^i,z^j), \tag{5.12}$$

where $K_m(\gamma^i,\gamma^j)$ is the Fisher kernel between γ^i and γ^j with the mth component in mixture $\tilde{\Theta}$ as the generative model:

$$K_m(\gamma^i,\gamma^j) = (\nabla_{\Theta_m}\log p(\gamma^i;\alpha_m,\mathbf{D}_m))^T\ \mathbf{I}_m^{-1}\ \nabla_{\Theta_m}\log p(\gamma^i;\alpha_m,\mathbf{D}_m). \tag{5.13}$$

Clearly the kernel between (z^i,γ^i) and (z^j,γ^j) is nonzero only if they are drawn from the same component mixed-effect model. Again we define the kernel between $\mathbf{y}^i$ and $\mathbf{y}^j$ as

$$\widetilde{K}(\mathbf{y}^i,\mathbf{y}^j) = E_{z^i,\gamma^i,z^j,\gamma^j}(\widetilde{K}((z^i,\gamma^i),(z^j,\gamma^j))|\,\mathbf{y}^i,\mathbf{y}^j;\mathbf{t}^i,\mathbf{t}^j,\widetilde{\mathcal{M}}). \tag{5.14}$$

The kernels $K((z^i, \gamma^i), (z^j, \gamma^j))$ from equation 5.9 and $\widetilde{K}((z^i, \gamma^i), (z^j, \gamma^j))$ in equation 5.12 are related by

$$K((z^i, \gamma^i), (z^j, \gamma^j)) = \frac{1}{\pi_{z^i}} \widetilde{K}((z^i, \gamma^i), (z^j, \gamma^j)) + \frac{1}{\pi_{z^i}} \delta(z^i, z^j).$$

A full derivation is in appendix A.

5.2.3 Design C. We can also base the kernel design on one mixed-effect model component instead on the mixture. Equivalently, we assume that the mixture model contains only one component in design A or B.

For the mixed-effect model for the group indexed m, the Fisher score for the ith individual,

$$\nabla_{\Theta_m} \log p(\gamma^i; \Theta_m),$$

describes how the log probability $p(\gamma^i; \Theta_m)$ responds to the change of mixed-effect model parameters Θ_m. This is a valid feature for classification since the likelihood of γ_i for individuals from different groups is likely to have a different response to the change of parameters Θ_m. The kernel between γ^i and γ^j is same as the one defined in equation 5.13 (design B):

$$K_m(\gamma^i, \gamma^j) = (\nabla_{\Theta_m} \log p(\gamma^i; \alpha_m, \mathbf{D}_m))^T \ \mathbf{I}_m^{-1} \ \nabla_{\Theta_m} \log p(\gamma^i; \alpha_m, \mathbf{D}_m),$$
$$m = 0, 1.$$

The kernel for $\mathbf{y}^i$ and $\mathbf{y}^j$ is

$$\begin{aligned} K(\mathbf{y}^i, \mathbf{y}^j) &= E(K(\gamma^i, \gamma^j) | \mathbf{y}^i, \mathbf{y}^j; \mathbf{t}^i, \mathbf{t}^j, \mathcal{M}_m) \\ &= \int\int K(\gamma^i, \gamma^j) p(\gamma^i | \mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_m) p(\gamma^j | \mathbf{y}^j; \mathbf{t}^j, \mathcal{M}_m) d\gamma^i d\gamma^j \\ &= \left(\int \nabla_{\tilde{\Theta}} \log p(\gamma^i | \tilde{\Theta}) p(\gamma^i | \mathbf{y}^i; \mathbf{t}^i, \mathcal{M}_m) \, d\gamma^i \right)^T \mathbf{I}_m^{-1} \\ &\quad \times \int \nabla_{\tilde{\Theta}} \log p(\gamma^j | \tilde{\Theta}) p(\gamma^j | \mathbf{y}^j; \mathbf{t}^j, \mathcal{M}_m) \, d\gamma^j. \end{aligned} \tag{5.15}$$

We can use the mixed-effect model trained on either the impaired group or the normal group. Not surprisingly, the mixed-effect models fit on the different groups describe the data quite differently with consequently different kernels. Our experiments show that the kernel based on the impaired group is significantly better for classification than the one based on the normal group. Therefore, the results we show for design C are based on the

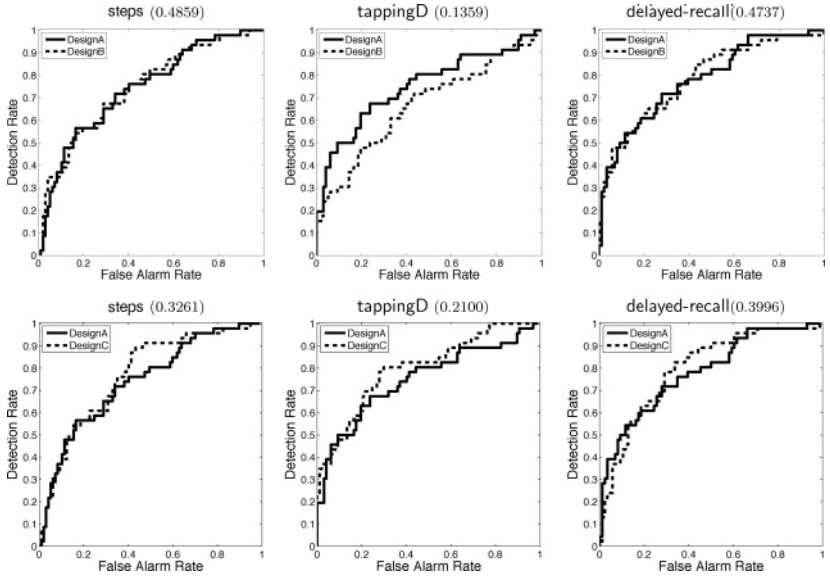

Figure 5: (Upper row) Design A versus design B. (Lower row) Design A versus design C. The number in the parentheses is the p-value (Z-test) for the null hypothesis: the AUC of design A is the same as the AUC of design B (or C).

impaired group model. This kernel is essentially the special case of design A or B with $\pi_0 = 1$ and $\pi_1 = 0$.

5.3 Empirical Comparison Between Kernels. We tested the three novel Fisher kernel designs on the four motor behaviors, gait speed, seconds, tappingD, and tappingN, and the two neuropsychological tests, delayed recall and logical memory II, with the mixed-effect models for each measurement trained separately. We use order 1 polynomials for the mixed-effect models.[11] For each measurement, the constructed kernels are used in support vector machines for classification. We compare the performance of two classifiers by comparing their respective ROC curves (see section 4.2). The ROC curves are estimated by a leave-one-out cross-validation and the optimization procedure described in section 4.5.

We compare designs A and B in the upper row of Figure 5. We show only the steps, tappingD, and delayed recall data. The two kernels have very comparable performance except on the tappingD time series, for which design A is slightly better than design B (at significance $p = 0.136$). We

[11] The order 2 polynomials (quadratic) model yields worse classification results than order 1 polynomials, and the result is omitted.

compare designs A and C in the lower row of Figure 5, finding that design C yields slightly better ROC curves than design A on the motor behaviors and comparable performance on delayed recall. (On gait speed, not shown, design C outperformed design A at $p = 0.158$.)

6 Kernels Without a Generative Model

In this section, we discuss two other feature extraction routines that are not based on generative models. The first models each subject with a polynomial curve and uses the least-square fitting coefficients as the feature vector. The second takes a nonparametric approach, fitting the observations of each subject with a smooth curve, and uses it as the summarizing feature for the classification afterward.

6.1 Parametric Feature Extraction. We summarize each individual time series with the least-squares fit coefficients for a d-degree polynomial regression model. That is, for subject i, the $d+1$-dimensional feature $\mathbf{p}^i$ is

$$\mathbf{p}^i = \arg\min_{\mathbf{p}} \sum_{j=1}^{N^i} \left(\sum_{l=0}^{d} p_l (t_j^i)^l - y_j^i \right)^2, \tag{6.1}$$

with $\mathbf{p} = [p_0, \ldots, p_d]^T$. We consider only $d = 1$ since a substantial proportion of subjects have no more than five observations, not enough for a reliable fitting of a polynomial with $d \geq 2$. We normalize the entries in $\mathbf{p}^i$ by the standard deviation and then use them as input to a support vector machine,

$$\hat{p}_l^i = \frac{p_l^i}{\sqrt{\frac{1}{k-1} \sum_j (p_l^j - \overline{p}_l)^2}}, \quad l = 0, \ldots, d, \tag{6.2}$$

with $\overline{p}_l = \frac{1}{k} \sum_{i=1}^{k} p_l^i$. We use an RBF kernel based on the squared Euclidean distance,

$$\mathbf{K}_{ij} = \exp\left(-\frac{||\hat{p}^i - \hat{p}^j||_2^2}{2s^2} \right), \tag{6.3}$$

where $\hat{\mathbf{p}}^i = [\hat{p}_0^i, \ldots, \hat{p}_d^i]^T$, and the radius s is chosen using leave-one-out cross-validation. In the remainder of the letter, we refer to the matrix $\mathbf{K}$ defined in equation 6.3 as the least squares (LSQ) kernel. (Note that the feature defined in equations 6.1 and 6.2 be used in other classifiers such as multilayer perceptron; Bishop, 1995.)

6.2 Nonparametric Feature Extraction. We can extend the feature extraction described in section 6.1 to a nonparametric form as follows. The model is based on gaussian process regression (Rasmussen & Williams, 2006) and the reproducing kernel Hilbert space (RKHS). We assume that the observations for each individual are generated from an independent gaussian process indexed by age. The nth observation for subject i is modeled as

$$y_n^i = f^i(t_n^i) + \epsilon_n^i, \quad n = 1, 2, \ldots, N_i,$$

where f^i is a gaussian process and ϵ_n^i is a white observation noise with standard deviation σ. We further assume that the f^i all have the same covariance function, denoted C. We can then summarize each individual by standard kernel regression (Rasmussen & Williams, 2006),

$$\hat{f}^i(t) = E(f(t)|\mathbf{y}^i; \mathbf{t}^i, C, \sigma) = C(t, \mathbf{t}^i)(C(\mathbf{t}^i, \mathbf{t}^i) + \sigma^2\mathbb{I})^{-1}\mathbf{y}^i,$$

where $C(\mathbf{t}^i, \mathbf{t}^i)$ is the matrix with element (n, m) set to $C(t_n^i, t_m^i)$.

The difference between two individuals can be measured by the distance between the two summarizing curves in a Hilbert space,

$$d_{ij} = ||\hat{f}^i - \hat{f}^j||_{\mathcal{H}}^2, \tag{6.4}$$

where $||\cdot||_{\mathcal{H}}$ is the norm in Hilbert space $\mathcal{H}$. [12] When $\mathcal{H}$ is chosen to be the RKHS induced by the covariance function C, this distance measure is

$$\begin{aligned} d(i, j) &= ||\hat{f}^i - \hat{f}^j||_{\mathcal{H}}^2 \\ &= ||C(t, \mathbf{t}^i)(C(\mathbf{t}^i, \mathbf{t}^i) + \sigma^2\mathbb{I})^{-1}\mathbf{y}^i - C(t, \mathbf{t}^j)(C(\mathbf{t}^j, \mathbf{t}^j) + \sigma^2\mathbb{I})^{-1}\mathbf{y}^j||_{\mathcal{H}}^2 \\ &= < C(t, \mathbf{t}^i)\mathbf{v}^i - C(t, \mathbf{t}^j)\mathbf{v}^j, C(t, \mathbf{t}^i)\mathbf{v}^i - C(t, \mathbf{t}^j)\mathbf{v}^j >_{\mathcal{H}}, \end{aligned} \tag{6.5}$$

where $\mathbf{v}^i = (C(\mathbf{t}^i, \mathbf{t}^i) + \sigma^2\mathbb{I})^{-1}\mathbf{y}^i$. Since

$$< C(t_n, t), C(t_m, t) >_{\mathcal{H}} = C(t_n, t_m),$$

the distance measurement can be simplified to

$$d_{ij} = (\mathbf{v}^i)^T C(\mathbf{t}^i, \mathbf{t}^i)\mathbf{v}^i + (\mathbf{v}^j)^T C(\mathbf{t}^j, \mathbf{t}^j)\mathbf{v}^j - 2(\mathbf{v}^i)^T C(\mathbf{t}^i, \mathbf{t}^j)\mathbf{v}^j. \tag{6.6}$$

[12] One might want to use $E(||f^i - f^j||_{\mathcal{H}}^2 | t^i, y^i, t^j, y^j)$ as the measure of distance. Unfortunately, this expectation goes to infinity as any random sample f from a gaussian process with C as the covariance function will have $||f||_{\mathcal{H}} = \infty$ with probability 1 (Seeger, 2004).

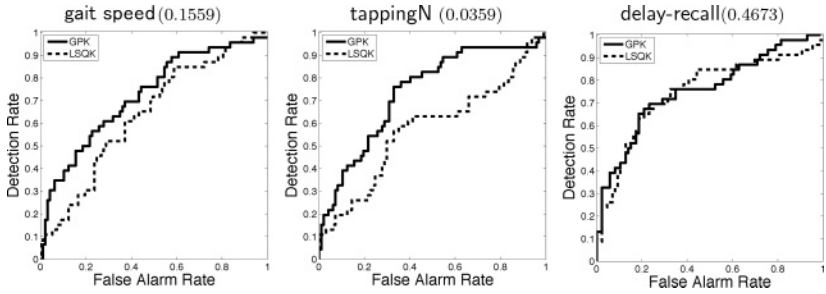

Figure 6: Gaussian process kernel versus LSQ kernel. The number in the parentheses is the p-value of the Z-test for the null hypothesis: the AUC of the gaussian process kernel is the same as the AUC of the LSQ kernel.

The distance d_{ij} can be interpreted as the Bregman divergence on f with $||f||^2_{\mathcal{H}}$ as the seed functional (Frigyik, Srivastava, & Gupta, 2006).

Based on this distance, we use the kernel

$$\mathbf{K}_{ij} = \exp\left(-\frac{d_{ij}}{2s^2}\right), \tag{6.7}$$

where the kernel width s is chosen using leave-one-out cross-validation. The matrix $\mathbf{K}$ defined in equation 6.6 is a Mercer kernel, simply because it can be rewritten as an RBF kernel after we embed the distance d_{ij} into an N-dimensional Euclidean space with $N = \sum_{i=1}^{N_i}$. In the remainder of the letter, we refer to this kernel as the gaussian process kernel. To get such a kernel, we need to specify the covariance function C and width s used in equation 6.6. In this letter, we use a gaussian covariance function $C(t, t') = \exp(-\frac{(t-t')^2}{2s_c^2})$, where s_c is chosen to be the average time interval between two adjacent observations, averaged over all subjects.

6.3 Comparison Between Parametric and Nonparametric Feature Extraction. We compare classifiers built on the parametric and nonparametric kernels in Figure 6. For each design, we set the radius s to maximize the classifier performance at the operating point using leave-one-out cross-validation. The ROC curves are in Figure 6. The gaussian process kernel (GPK) yields a slightly larger AUC than the LSQ kernel (LSQK), but the difference is significant (at $p < 0.05$) only for the tappingN. On delayed recall, the two are comparable, and there is a slight advantage (at $p = 0.16$) on gait speed.

7 Comparing Generative and Discriminative Models

Finally, we compare the best performer in each of the three categories of methods:

- For the likelihood ratio tests (sections 4.1–4.3), we choose the one based on the mixed-effect model, since it is slightly better than the one based on the simplified model that assumes no random effect.
- For the discriminative models based on the extensions to the Fisher kernel (see section 5), we pick design C. It is consistently better than designs A and B on all six measurements, although the difference is not statistically significant.
- For the feature extraction models independent of the mixed-effect model (section 6), we pick the gaussian process kernels since it performs better than the LSQ kernel on all six measurements, although the difference is not statistically significant (except on tappingN).

In Figure 7 we compare the three best performers for all six measures. It shows that both discriminative models are better than the likelihood ratio test on the four motor behaviors. On the two psychometric tests, the likelihood ratio test and the kernel-based classifiers have similar performance. Among the discriminative models applied to the motor behaviors, the one based on our Fisher kernel extension (design C) outperforms the likelihood ratio classifiers at $p < 0.08$. The superiority of the Gaussian process kernel relative to the likelihood ratio classifiers is not as strong ($p \geq 0.16$), except on tappingN, for which the differences are significant at $p < 0.05$. The AUCs of the best performers on all six measures are also listed in Table 2. It shows that the Fisher kernel extension is overall the most reliable since it is the best on four of the six measurements and achieves performance comparable to the best on the rest two measurements.

8 Discussion and Conclusion

We presented several models for predicting cognitive decline based on clinical recordings of motor and psychometric tests. We view these classification studies as an initial step toward a more sophisticated system that can estimate the risk of the onset of MCI at a given age.

We developed and compared two categories of methods: likelihood-ratio tests based on generative models and discriminative models. We adopt the mixed-effect model for its ability to model both the population behavior and the individual's deviation from the population. In section 3 we discussed the models in detail and showed that these models capture the differences between the normal and impaired groups. In section 4, we explored the

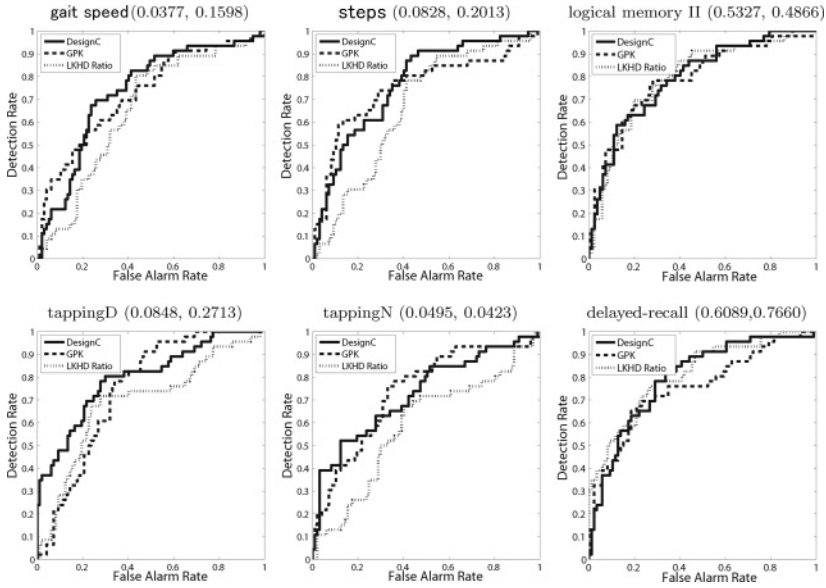

Figure 7: The comparison between Fisher kernel extension (design C), gaussian process kernel, and likelihood ratio classifiers. (mixed-effect model). For each measurement, the first number in the parentheses is the p-value (Z-test) for the null hypothesis "the AUC of design C is same as the AUC of likelihood ratio classifier," and the second number is the p-value (Z-test) for the null hypothesis "the AUC of gaussian process kernel is the same as the AUC of likelihood ratio classifier."

Table 2: AUCs of the Model with the Best Performance on Six Measurements.

Steps	Gait speed	Logical memory II
0.7333 (design C)	0.7674 (design C)	0.7945 (design C)
TappingD	TappingN	Delayed Recall
0.7990 (design C)	0.7360 (GPK)	0.8124 (LKHD)

discriminative capability of the mixed-effect model, building a likelihood ratio classifier from the mixed-effect models. This yields reasonable classification results on four motor behaviors and has excellent performance on two neuropsychological tests.

For use in SVM classifiers, we developed two types of kernels based on the longitudinal data. The first type exploits the latent structure of mixed-effect models to extend the Fisher kernel construction so it deals properly with time series of unequal length and variable sampling intervals.

The key development is to write a kernel in the latent variables (which are of the same dimension for all observed sequences) using the proper metric and then project this latent-variable kernel into the space of observations. The construction allows direct computation of the Fisher information. This is an improvement over the usual ad hoc replacement of the Fisher information with an identity matrix for sequences with varying length. Using the Fisher information as the metric provides proper scaling of the Fisher scores and retains the invariance of the kernel under reparameterization of the generative model present in the formal development. The second type of kernel uses feature extraction routines not based in the mixed-effect models. We constructed both parametric and nonparametric (gaussian process) regression curves for each individual time series. The nonparametric approach gives a new distance measure potentially useful for a wide range of time-series applications. Our experiments show that the discriminative methods yield significantly better classification performance than likelihood ratio tests on motor behaviors, and are comparable to them on the neuropsychological tests.

Several problems remain unsolved. First, our classification algorithms do not give an estimation of the risk of decline at different ages. One possible remedy is to combine traditional survival analysis (Klein & Moeschberger, 2003) with the classification techniques we developed in this letter. Second, we have not found an effective way to fuse the information from the different time series. Obviously we can do so with a multivariate output mixed-effect model that describes several types of observations with one model. Unfortunately, our preliminary results show that this does not capture enough correlation between different types of observations to improve the classification. Another direction is to build a separate kernel for each type of observation and combine them in a discriminator. Methods like kernel-target alignment (Cristianini, Shawe-Taylor, Elisseeff, & Kandola, 2002) or kernel extrapolation (Vishwanathan, Borgwardt, Guttman, & Smola, 2006) provide interesting choices. We will explore these ideas in our future research.

Appendix: Two Kernels in Design B

The Fisher score is the gradient of the log likelihood

$$\begin{aligned}\phi_{\widetilde{\Theta}}(z^i, \gamma^i) &\equiv \nabla_{\widetilde{\Theta}} \log p(z^i, \gamma^i; \widetilde{\Theta}) \\ &= \left[\frac{\partial \log p}{\partial \pi_0}; \frac{\partial \log p}{\partial \alpha_0}; \frac{\partial \log p}{\partial \mathbf{D}_0}; \frac{\partial \log}{\partial \pi_1}; \frac{\partial \log p}{\partial \alpha_1}; \frac{\partial \log p}{\partial \mathbf{D}_1}\right]^T,\end{aligned}$$

and the Fisher kernel for the joint variable (γ^i, z^i) is defined as

$$K((z^i, \gamma^i), (z^j, \gamma^j)) = (\nabla_{\widetilde{\Theta}} \log p(\gamma^i|\widetilde{\Theta}))^T (\mathbf{I}^{z,\gamma})^{-1} \nabla_{\widetilde{\Theta}} \log p(\gamma^i|\widetilde{\Theta}), \quad \text{(A.1)}$$

where $\mathbf{I}^{z,\gamma}$ is the Fisher information matrix. In equation A.1, we have for $m = 0, 1$

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \pi_m} = \delta(z^i, m)\frac{1}{\pi_m},$$

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \alpha_m} = \delta(z^i, m)\mathbf{D}_m^{-1}(\alpha_m - \gamma^i),$$

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \mathbf{D}_m} = \delta(z^i, m)\left\{-\frac{1}{2}\mathbf{D}_m^{-1} + \frac{1}{2}\mathbf{D}_m^{-1}(\alpha_m - \gamma^i)(\alpha_m - \gamma^i)^T\mathbf{D}_m^{-1}\right\}.$$

Note that

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \alpha_m} = \delta(z^i, m)\frac{\partial \log p(\gamma^i; \Theta_m)}{\partial \alpha_m}, \tag{A.2}$$

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \mathbf{D}_m} = \delta(z^i, m)\frac{\partial \log p(\gamma^i; \Theta_m)}{\partial \mathbf{D}_m}. \tag{A.3}$$

Denoting $\Theta_m = \{\alpha_m, \mathbf{D}_m\}$ $m = 0, 1$, equations A.2 and A.3 can be summarized as

$$\frac{\partial \log p(z^i, \gamma^i; \widetilde{\Theta})}{\partial \Theta_m} = \delta(z^i, m)\frac{\partial \log p(\gamma^i; \Theta_m)}{\partial \Theta_m}. \tag{A.4}$$

The Fisher information matrix $\mathbf{I}^{z,\gamma}$ is defined as

$$\mathbf{I}^{z,\gamma} = E_{z^i,\gamma^i}(\phi_{\widetilde{\Theta}}^T(z^i, \gamma^i)\phi_{\widetilde{\Theta}}(z^i, \gamma^i)|\widetilde{\Theta}) \tag{A.5}$$

$$= \begin{bmatrix} E\left(\frac{\partial \log p}{\partial \pi_0}\left(\frac{\partial \log p}{\partial \pi_0}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \pi_0}\left(\frac{\partial \log p}{\partial \Theta_0}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \Theta_0}\left(\frac{\partial \log p}{\partial \pi_0}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \Theta_0}\left(\frac{\partial \log p}{\partial \Theta_0}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \pi_1}\left(\frac{\partial \log p}{\partial \pi_0}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \pi_1}\left(\frac{\partial \log p}{\partial \Theta_0}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \Theta_1}\left(\frac{\partial \log p}{\partial \pi_0}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \Theta_1}\left(\frac{\partial \log p}{\partial \Theta_0}\right)^T|\widetilde{\Theta}\right) \end{bmatrix}$$

$$\begin{bmatrix} E\left(\frac{\partial \log p}{\partial \pi_0}\left(\frac{\partial \log p}{\partial \pi_1}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \pi_0}\left(\frac{\partial \log p}{\partial \Theta_1}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \Theta_0}\left(\frac{\partial \log p}{\partial \pi_1}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \Theta_0}\left(\frac{\partial \log p}{\partial \Theta_1}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \pi_1}\left(\frac{\partial \log p}{\partial \pi_1}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \pi_1}\left(\frac{\partial \log p}{\partial \Theta_1}\right)^T|\widetilde{\Theta}\right) \\ E\left(\frac{\partial \log p}{\partial \Theta_1}\left(\frac{\partial \log p}{\partial \pi_1}\right)^T|\widetilde{\Theta}\right) & E\left(\frac{\partial \log p}{\partial \Theta_1}\left(\frac{\partial \log p}{\partial \Theta_1}\right)^T|\widetilde{\Theta}\right) \end{bmatrix}. \tag{A.6}$$

It is straightforward to verify that for $m = 0, 1$

$$E_{z^i,\gamma^i}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_m}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_m}\right)^T \middle| \widetilde{\Theta}\right) = \frac{1}{\pi_m},$$

$$E_{z^i,\gamma^i}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_m}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \Theta_m}\right)^T \middle| \widetilde{\Theta}\right) = 0,$$

$$E_{z^i,\gamma^i}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_m}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_{1-m}}\right)^T \middle| \widetilde{\Theta}\right) = 0,$$

$$E_{z^i,\gamma^i}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \pi_m}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \Theta_{1-m}}\right)^T \middle| \widetilde{\Theta}\right) = 0,$$

$$E_{z^i,\gamma^i}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \Theta_m}\left(\frac{\partial \log p(z^i,\gamma^i;\widetilde{\Theta})}{\partial \Theta_m}\right)^T \middle| \widetilde{\Theta}\right)$$
$$= \pi_m E_{z^i,\gamma^i} \times \left(\frac{\partial \log p(\gamma^i;\Theta_m)}{\partial \Theta_m}\left(\frac{\partial \log p(\gamma^i;\Theta_m)}{\partial \Theta_m}\right)^T \middle| \Theta_m\right),$$

from which equation A.6 can be simplified as

$$\mathbf{I}^{z,\gamma} = \begin{bmatrix} \frac{1}{\pi_0} & 0 & 0 & 0 \\ 0 & \pi_0 E\left(\frac{\partial \log p}{\partial \Theta_0}\left(\frac{\partial \log p}{\partial \Theta_0}\right)^T | \Theta_0\right) & 0 & 0 \\ 0 & 0 & \frac{1}{\pi_1} & 0 \\ 0 & 0 & 0 & \pi_1 E\left(\frac{\partial \log p}{\partial \Theta_1}\left(\frac{\partial \log p}{\partial \Theta_1}\right)^T | \Theta_1\right) \end{bmatrix}. \tag{A.7}$$

It is not hard to see from here that

$$\begin{aligned} K((z^i,\gamma^i),(z^j,\gamma^j)) &= (\nabla_{\widetilde{\Theta}} \log p(\gamma^i|\widetilde{\Theta}))^T (\mathbf{I}^{z,\gamma})^{-1} \nabla_{\widetilde{\Theta}} \log p(\gamma^i|\widetilde{\Theta}) \\ &= \frac{1}{\pi_{z^i}} \delta(z^i,z^j)(1 + K_{z^i}(\gamma^i,\gamma^j)) \\ &= \frac{1}{\pi_{z^i}} \widetilde{K}((z^i,\gamma^i),(z^j,\gamma^j)) + \frac{1}{\pi_{z^i}} \delta(z^i,z^j). \end{aligned}$$

Acknowledgments

This work was supported by Intel Corp. under the OHSU BAIC award, by NSF under grant IIS-0812687, by the National Institutes of Health, National Institute of Aging grants P30-AG008017, P30-AG024978, and the Department of Veterans Affairs. We thank Milar Moore and Robin Guariglia of the Layton Aging and Alzheimer's Disease Center for help with data from the Oregon Brain Aging Study. We thank Misha Pavel, Tamara Hayes, and Nichole Carlson for helpful discussion. We thank the anonymous reviewer for comments helping to clarify the text.

References

Bishop, C. M. (1995). *Neural networks for pattern recognition*. New York: Oxford University Press.

Boise, L., Morgan, D., Kaye, J., & Camicioli, R. (1999). Delays in the diagnosis of dementia: Perspectives of family caregivers. *American Journal of Alzheimer's Disease and Other Dementias, 14*, 20–26.

Camicioli, R., Howieson, D., Oken, B., Sexton, G., & Kaye, J. (1998). Motor slowing precedes cognitive impairment in the oldest old. *Neurology, 50*, 1496–1498.

Chen, M.-H., Shao, Q.-M., & Ibrahim, J. G. (2000). *Monte Carlo methods in Bayesian computation*. New York: Springer.

Chen, P., Ratcliff, G., Phil, D., Belle, S., Cauley, J., Kosky, S. D., et al. (2000). Cognitive tests that best discriminate between presymptomatic and those who remain nondemented. *Neurology, 55*, 1847–1853.

Cristianini, N., Shawe-Taylor, J., Elisseeff, A., & Kandola, J. (2002). On kernel-target alignment. In T. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems, 14* (pp. 367–373). Cambridge, MA: MIT Press.

Demidenko, E. (2004). *Mixed models, theory and applications*. Hoboken, NJ: Wiley.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc., B39*, 1–39.

Druck, G., Pal, C., Zhu, X., & McCallum, A. (2007). Semi-supervised classification with hybrid generative/discriminative methods. In *Conference on Knowledge Discovery and Data Mining* (pp. 280–289). New York: ACM.

Frigyik, B., Srivastava, S., & Gupta, M. (2006). *Functional Bregman divergence and Bayesian estimation of distributions*. arXiv:cs/0611123.

Green, M., Kaye, J., & Ball, M. (2000). The Oregon Brain Aging Study: Neuropathology accompanying healthy aging in the oldest old. *Neurology, 54*(1), 105–113.

Guo, W. (2002). Functional mixed effects models. *Biometrics, 58*, 121–128.

Gwyther, L. (2000). Family issues in dementia: Finding a new normal. *Neurologic Clinics, 18*, 993–1010.

Holub, A., Welling, M., & Perona, P. (2005). Combining generative models and Fisher kernels for object recognition. In *International Conference on Computer Vision* (pp. 136–143). Piscataway, NJ: IEEE.

Howieson, D., Dame, A., Camicioli, R., Sexton, G., Payami, H., & Kaye, J. (1997). Cognitive markers preceding Alzheimer's dementia in the healthy oldest old. *J. Am. Geriatr. Soc., 45*, 584–589.

Jaakkola, T., Diekhaus, M., & Haussler, D. (1999). Using the Fisher kernel method to detect remote protein homologies. In *Proc. 7th Intell. Sys. Mol. Biol.* (pp. 149–158). Cambridge, MA: AAAI Press.

Jaakkola, T., & Haussler, D. (1998). *Exploiting generative models in discriminative classifiers* (Tech. Rep.). Santa Cruz: Department of Computer Science, University of California, Santa Cruz.

Jaakkola, T., & Haussler, D. (1999). Probabilistic kernel regression models. In *Proceedings of the AISTATS 1999*. Society of Artificial Intelligence and Statistics.

Jaakkola, T., Meila, M., & Jebara, T. (1999). *Maximum entropy discrimination* (Tech. Rep. AITR-1668). Cambridge, MA: MIT, Artificial Intelligence Laboratory.

Keogh, E., & Pazzani, M. (1998). An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In *Knowledge Discovery in Data 1998* (pp. 239–241). New York: ACM Press.

Klein, J., & Moeschberger, M. (2003). *Survival analysis*. New York: Springer.

Laird, N., Lange, N., & Stram, D. (1987). Random-effects models for longitudinal data. *Journal of the American Statistical Association, 82*(397), 97–105.

Laird, N. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics, 38*(4), 963–974.

Marquis, S., Moore, M., Howieson, D. B., Sexton, G., Payami, H., Kaye, J. A. et al. (2002). Independent predictors of cognitive decline in healthy elderly persons. *Arch. Neurol., 59*, 601–606.

Moreno, P., & Rifkin, R. (2000). Using the fisher kernel method for Web audio classification. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing* (pp. 2417–2420). Piscataway, NJ: IEEE.

Ng, A., & Jordan, M. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In T. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems, 14* (pp. 287–296). Cambridge, MA: MIT Press.

Pepe, M. S. (2003). *The statistical evaluation of medical tests for classification and prediction*. New York: Oxford University Press.

Rasmussen, C., & Williams, C. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Richards, M., Stern, Y., & Mayeux, R. (1993). Subtle extrapyramidal signs can predict the development of dementia in elderly individuals. *Neurology, 43*, 2184–2188.

Riefler, V., & Larson, E. (1988). Excess disability in demented elderly outpatients: The rule of halves. *Journal of the American Geriatrics Society, 47*, 1065–1072.

Seeger, M. (2004). Gaussian process for machine learning. *International Journal of Neural System, 14*(2), 69–106.

Seymore, K., McCallum, A., & Rosenfeld, R. (1999). Learning hidden Markov model structure for information extraction. In *AAAI 99 Workshop on Machine Learning for Information Extraction*. Cambridge, MA: AAAI Press.

Tsuda, K., Kin, T., & Asai, K. (2002). Marginalized kernels for biological sequences. *Bioinformatics, 1*(1), 1–8.

Vapnik, V. (1998). *Statistical learning theory*. Hoboken, NJ: Wiley.

Verghese, J., Lipton, R., Hall, C., Kuslansky, G., Katz, M., & Buschke, H. (2002). Abnormality of gait as a predictor of non-Alzheimer's dementia. *N. Engl. J. Med., 347*(22), 1761–1768.

Vishwanathan, S., Borgwardt, K. M., Guttman, O., & Smola, A. (2006). Kernel extrapolation. *Neurocomputing, 69*, 721–729.

Wilson, R. S., Schneider, J. A., Bienias, J. L., Evans, D. A., & Bennett, D. A. (2003). Parkinsonianlike signs and risk of incident Alzheimer disease in older persons. *Arch. Neurol., 60*, 539–544.

Received July 6, 2010; accepted January 29, 2011.