Kernel density estimation-based real-time prediction for respiratory motion

IOP PUBLISHING PHYSICS IN MEDICINE AND BIOLOGY

Phys. Med. Biol. **55** (2010) 1311–1326 doi:10.1088/0031-9155/55/5/004

Kernel density estimation-based real-time prediction for respiratory motion

Dan Ruan

Department of Radiation Oncology, Stanford University, Stanford, CA, USA

E-mail: druan@stanford.edu

Received 16 April 2009, in final form 15 November 2009
Published 4 February 2010
Online at stacks.iop.org/PMB/55/1311

Abstract

Effective delivery of adaptive radiotherapy requires locating the target with high precision in real time. System latency caused by data acquisition, streaming, processing and delivery control necessitates prediction. Prediction is particularly challenging for highly mobile targets such as thoracic and abdominal tumors undergoing respiration-induced motion. The complexity of the respiratory motion makes it difficult to build and justify explicit models. In this study, we honor the intrinsic uncertainties in respiratory motion and propose a statistical treatment of the prediction problem. Instead of asking for a deterministic covariate–response map and a unique estimate value for future target position, we aim to obtain a distribution of the future target position (response variable) conditioned on the observed historical sample values (covariate variable). The key idea is to estimate the joint probability distribution (pdf) of the covariate and response variables using an efficient kernel density estimation method. Then, the problem of identifying the distribution of the future target position reduces to identifying the section in the joint pdf based on the observed covariate. Subsequently, estimators are derived based on this estimated conditional distribution. This probabilistic perspective has some distinctive advantages over existing deterministic schemes: (1) it is compatible with potentially inconsistent training samples, i.e., when close covariate variables correspond to dramatically different response values; (2) it is not restricted by any prior structural assumption on the map between the covariate and the response; (3) the two-stage setup allows much freedom in choosing statistical estimates and provides a full nonparametric description of the uncertainty for the resulting estimate. We evaluated the prediction performance on ten patient RPM traces, using the root mean squared difference between the prediction and the observed value normalized by the standard deviation of the observed data as the error metric. Furthermore, we compared the proposed method with two benchmark methods: most recent sample and an adaptive linear filter. The kernel density estimation-based prediction results demonstrate universally significant improvement over the alternatives and are especially valuable for long lookahead time, when the alternative methods fail to produce useful predictions.

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Modern radiotherapy systems are capable of delivering the prescribed dose to a specified target position with high precision. However, target motion, if not properly accounted for, compromises the delivery accuracy. Intrafractional target motion during treatment is managed with passive gating (Keall *et al* 2002) or adaptive tracking (Nuyttens *et al* 2006). Furthermore, intrinsic system latency exists due to hardware limitation, software processing time and data communication. These considerations have motivated a large body of studies on prediction algorithms during treatment, especially for highly mobile targets, such as thoracic and abdominal tumors that are affected by respiratory motion.

The problem of predicting respiratory motion has been intensively studied, and the existing methods can be classified into two categories: (1) those that assumes a specific inference model structure between the covariate, which is usually constructed from a finite sequence of historical samples, and the response (2) those that are 'model free'. The general class of linear filters fall into the first category, even though they may differ in adaptivity and specific formulations. Autoregressive moving average (ARMA) model is a straightforward generalization of the linear model (McCall and Jeraj 2007). Linear models in the statistical setting have given rise to the application of the Kalman filter and multiple models (Putra *et al* 2008, Zimmerman *et al* 2008, McMahon *et al* 2007). By estimating the regression coefficients, these models extrapolate the behavior between the training covariate and training response values, and apply the estimated inference map to the test covariate.

One potential drawback of these fixed-structure models is their incapability to learn the inference pattern from samples that are further away in space/time, but more similar in behavior, e.g., training samples that are obtained at similar breathing phase. Changing the representation space, such as the sinusoidal (Vedam *et al* 2004) or scale space (Ernst *et al* 2007), is one way to 'pull' these samples together. An even more flexible way is to learn the covariate–response directly, as in neural networks (Isaksson *et al* 2005, Kakar *et al* 2005, Murphy and Dieterich 2006). Note that even in neural network setup, a consistent and smooth inference map is assumed, and it will have difficulty if there exist multiple inconsistent training samples whose covariates are identical (or very close), but have very distinct response values.

In this study, we adopt a statistical prospective and consider the probability distribution of the test response upon observing the test covariate. Treating the prediction as a random variable not only honors the inconsistent training samples, but also provides a natural nonparametric description of the uncertainty associated with any prediction estimate. We adopt a kernel density estimator to approximate the joint probability distribution of the covariate and response variable from training samples. The distribution of the test response variable is obtained as the section of the joint distribution picked out by the observed test covariate value.

We present the basic theory of the proposed methodology in section 2. Section 3 describes the test data and the implementation procedure and reports the experimental results. Section 4 provides some structural discussion, and section 5 summarizes the study and discusses future work.

2. Methods

2.1. Basic setup

At current time instant t, we are given a set of discrete samples s_i, $i = 1, 2, \ldots, K$, of breathing trajectory, acquired at preceding times $t_i < t$ prior to t. For simplicity, we discuss the formulation when scalar observations are acquired at uniform time intervals and the look-ahead length is an integer multiple L of the sampling interval. Then, for any $i \leqslant K - L$, we can construct a length p covariate $\boldsymbol{x}_i = [s_{i-(p-1)\Delta}, s_{i-(p-2)\Delta}, \ldots, s_i]$ and response $\boldsymbol{y}_i = s_{i+L}$. The parameter Δ is an integer that indicates the 'lag length' used to augment the covariate. It should be chosen properly to balance the effect of system dynamics and observation noise (Ruan *et al* 2007).

Let $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i) \in \Re^{p+1}$, $i = 1, 2, \ldots, M$, denote the collection of training samples, where $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ are the covariate variable and the response variable, respectively. The goal of prediction is to obtain an estimate of the unknown $\boldsymbol{y}$ given a test covariate $\boldsymbol{x}$.

The key idea of the proposed statistical method is to consider the probability distribution (pdf) of the random vector $Z = (X, Y) \in \Re^n$, and regard each training sample $\boldsymbol{z}_i$ as an realization of Z. The prediction (or more generally inference task) becomes two folds: (1) estimate the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$, for observed covariate $\boldsymbol{x}$ and (2) obtain an estimate $\hat{\boldsymbol{y}}$ from such distribution. Rather than formulating the whole problem in a single optimization setting, we have chosen to present the estimation of conditional pdf and the subsequent estimation of $\boldsymbol{y}$ in a decoupled manner, to honor the fact that each module is self-contained and that once the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ is obtained, the user has the freedom to choose any estimate based on the specific application. Yet the overall setting of constructing the distribution in the joint (X, Y) space, and the logic of obtaining an estimate based on the conditional probability holds regardless.

We present the framework for kernel-based prediction in section 2.2, which provides a pdf of the response variable. Section 2.3 discusses some natural estimates derived from the resulting distribution. Section 2.4 provides a specific example for the proposed scheme and section 2.5 explains in depth certain design considerations, and describes some variations that could potentially improve the prediction performance.

2.2. Estimating the distribution of response variable with kernel density approximation

We consider the samples $\boldsymbol{z}_i = (\boldsymbol{x}_i, \boldsymbol{y}_i)$, $i = 1, 2 \ldots, M$, as independent samples of the random vector Z and obtain the kernel density approximation (Duda *et al* 2001) of the pdf $p(\boldsymbol{z})$ as the superposition of (equally weighted) local kernel densities centered about each sample $\boldsymbol{z}_i$:

$$p(\boldsymbol{z}) = \frac{1}{M} \sum_{i=1}^{M} \kappa(\boldsymbol{z}|\boldsymbol{z}_i), \tag{1}$$

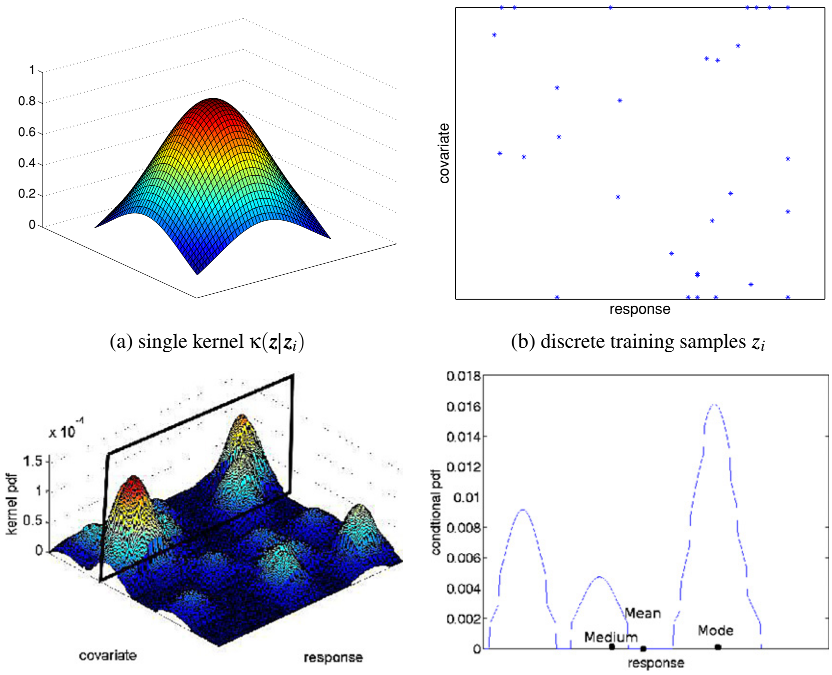
where $\kappa(\boldsymbol{z}|\boldsymbol{z}_i)$ is a local density kernel.

The distribution of the response variable conditioned on the covariate $p(\boldsymbol{y}|\boldsymbol{x})$ can be written as

$$p(\boldsymbol{y}|\boldsymbol{x}) = p((\boldsymbol{x}, \boldsymbol{y})|\boldsymbol{x}) = p(\boldsymbol{z}|\boldsymbol{x}) = p(\boldsymbol{z})/p(\boldsymbol{x}), \tag{2}$$

where $p(\boldsymbol{x})$ is the marginal distribution $p(\boldsymbol{x}) = \int p((\boldsymbol{x}, \boldsymbol{y}))\,\mathrm{d}\boldsymbol{y}$. The conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ is the (normalized) section of $p(\boldsymbol{z})$ at $X = \boldsymbol{x}$.

Equations (1) and (2) provide the principle for estimating the distribution of the response variable conditioned on the observed test covariate, by approximating the joint distribution



(a) single kernel $\kappa(\boldsymbol{z}|\boldsymbol{z}_i)$

(b) discrete training samples z_i

(c) kernel density estimated joint distribution $p(z)$

(d) conditional distribution $p(y|x)$.

Figure 1. Schematic of the kernel density estimation-based prediction method: a single kernel density (a) is placed at each discrete training sample (b) to construct the joint pdf (c). Based on the observed test covariate, the corresponding section of the joint pdf is selected and renormalized to obtain the conditional distribution of the test response (d), where certain statistics could be used as the prediction, such as mean, medium and mode, as illustrated in (d).

of covariate–response with kernel densities. Figure 1 illustrates the major component in this procedure.

A common choice of kernel density is the Gaussian kernel:

$$\kappa(z|z_i, \Sigma_i) = \mathcal{N}(z - z_i), \qquad \Sigma_i = \frac{1}{(2\pi)^{n/2}|\Sigma_i|^{1/2}} \exp[-(z - z_i)^T \Sigma_i^{-1}(z - z_i)].$$

In the prediction application, the various coordinates in z correspond to different states in the covariate or response variable. Therefore, it is feasible to assume the covariance matrix to be block separable:

$$\Sigma_i = \begin{bmatrix} \Sigma_{x,i} & 0 \\ 0 & \sigma_{y,i}^2 \end{bmatrix}. \tag{3}$$

The covariance of the local Gaussian kernel about each sample z_i reflects contribution of the sample to the local curvature of the overall probability distribution. If all training samples are obtained under similar environment, then it is reasonable to assume $\Sigma_{x,i} = \Sigma_x$ and $\sigma_{y,i} = \sigma_y$ for all i. The block diagonal form of the covariance indicates a separable Gaussian kernel, which enables further simplification of (2) as follows:

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{x}) &= \frac{1}{M}\sum_i \kappa((\boldsymbol{x},\boldsymbol{y})|\boldsymbol{z}_i) \\ &= \frac{1}{C}\sum_i \exp\bigl[-(\boldsymbol{x}-\boldsymbol{x}_i)^T\Sigma_x^{-1}(\boldsymbol{x}-\boldsymbol{x}_i) - \|\boldsymbol{y}-\boldsymbol{y}_i\|/\sigma_y^2\bigr] \\ &= \frac{1}{C}\sum_i \exp\bigl[-(\boldsymbol{x}-\boldsymbol{x}_i)^T\Sigma_x^{-1}(\boldsymbol{x}-\boldsymbol{x}_i)\bigr]\exp\bigl[-\|\boldsymbol{y}-\boldsymbol{y}_i\|/\sigma_y^2\bigr]. \end{aligned} \tag{4}$$

The normalization parameters C is independent of both i and $\boldsymbol{y}$, so one could delay the normalization until the last step by scaling the final conditional pdf to unity integral. Define the sample weights w_i as

$$w_i = \exp\bigl[-(\boldsymbol{x}-\boldsymbol{x}_i)^T\Sigma_x^{-1}(\boldsymbol{x}-\boldsymbol{x}_i)\bigr], \tag{5}$$

and (4) reduces to

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{\tilde{C}}\sum_i w_i \exp\bigl[-\|\boldsymbol{y}-\boldsymbol{y}_i\|/\sigma_y^2\bigr], \tag{6}$$

with normalization parameter $\tilde{C}$. This indicates that conditioned on the test covariate $\boldsymbol{x}$, the distribution of the response variable $\boldsymbol{y}$ is a Gaussian mixture, with each Gaussian component centered at the sampled $\boldsymbol{y}_i$'s. The weights in the mixture are determined by how 'close' the test covariate $\boldsymbol{x}$ is to the covariate $\boldsymbol{x}_i$ in the training samples.

2.3. Natural estimates

Given the distribution of the response variable Y conditioned on the observed covariate state $\boldsymbol{x}$, one could devise estimates of the response variable accordingly.

2.3.1. The mean estimate. The mean of a random variable minimizes the expected squared error:

$$\hat{\boldsymbol{y}}_{\text{mean}} = \arg\min_{\boldsymbol{y}} E[(\boldsymbol{y}-Y)^2|\boldsymbol{x}] = E[Y|\boldsymbol{x}]. \tag{7}$$

Given the Gaussian mixture conditional distribution (6), the mean estimate reads

$$\hat{\boldsymbol{y}}_{\text{mean}} = E[Y|\boldsymbol{x}] = \frac{1}{\tilde{C}}\int\sum_i \boldsymbol{y}w_i\exp\bigl[-\|\boldsymbol{y}-\boldsymbol{y}_i\|\big/\sigma_y^2\bigr]\,\mathrm{d}\boldsymbol{y} = \frac{1}{\sum_i w_i}\sum_i w_i\boldsymbol{y}_i. \tag{8}$$

It turns out that the mean estimate is the weighted sum of the training response values, where the weights are determined by the 'closeness' between the testing and training covariates. Also note that explicit normalization is no longer necessary due to cancellation.

2.3.2. The mode estimate. The mode can be considered as a maximum *a posteriori* probability (MAP) estimate, as it seeks a response value that maximizes the conditional distribution:

$$\hat{\boldsymbol{y}}_{\text{mode}} = \arg\max_{\boldsymbol{y}} p(\boldsymbol{y}|\boldsymbol{x}). \tag{9}$$

2.3.3. The medium estimate. From the point of view of order statistics, one could also adopt the medium estimate, which is defined as

$$\hat{\boldsymbol{y}}_{\text{medium}} = \{\boldsymbol{y} : p(Y \leqslant \boldsymbol{y}) = 1/2\}\,. \tag{10}$$

Note that for mixture Gaussian distribution in (6), the cumulative distribution can be obtained as

$$\begin{aligned} p(Y \leqslant \boldsymbol{y}|\boldsymbol{x}) &= \int_{-\infty}^{\boldsymbol{y}} \sum_i \tilde{y} w_i \exp\left[-\|\tilde{y} - y_i\|/\sigma_y^2\right] \mathrm{d}\tilde{y} \\ &= \frac{1}{2} \sum w_i \left[1 + \operatorname{erf}\left(\frac{y - y_i}{\sqrt{2}\sigma_y}\right)\right], \end{aligned} \tag{11}$$

where the Gauss error function is defined as

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \mathrm{e}^{-x^2}\, \mathrm{d}x.$$

2.4. An exemplary scheme for respiratory motion prediction

As an example, we provide the algorithmic flow chart using Gaussian kernel and the mean estimate.

Algorithm 1 Predict $\hat{\boldsymbol{y}}$ from $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ with Gaussian kernel and mean estimate.

```
Determine covariance $\Sigma_x$ and $\sigma_y$ for covariate and response variables.
Compute the weighting according to (5).
Compute the mean estimate $\hat{y}_{\text{mean}} = \frac{w_i y_i}{\sum_i w_i}$, equivalent to (8).
```

Note that it is unnecessary to compute the conditional distribution explicitly when the mean estimate is used, as the order of taking the expectation to obtain the mean and computing the weighted sum for the conditional pdf can be interchanged. This is not true in general, as in the case of mode and medium estimates.

2.5. Design considerations and potential variations

We describe some design considerations and variations that could potentially improve the prediction performance.

2.5.1. Data-driven covariance/bandwidth selection. In introducing the principle of the kernel method, we assumed the covariance of the covariate and the response variables *a priori*. In practice, one needs to determine these values. Setting the covariance for a general kernel is a challenging problem and can be identified with the bandwidth selection problem studied in nonparametric density estimators. To this end, statistical methods in data-driven bandwidth selection (Sheather and Jones 1991, Botev 2006) may be applied.

In our application, the end goal is to obtain an estimate of the test response rather than estimating the pdf itself and we expect the prediction performance to be less sensitive to the choice of covariance than in general fitting problem. With the assumed separable Gaussian kernel and approximate spatial invariance (cf (3)), we estimate the covariate covariance Σ_x and response σ_y from the training population as follows:

$$\bar{\boldsymbol{x}} \triangleq \frac{1}{N_{\text{train}}} \sum_{i \in \text{training set}} \boldsymbol{x}_i, \qquad \bar{\boldsymbol{y}} \triangleq \frac{1}{N_{\text{train}}} \sum_{i \in \text{training set}} \boldsymbol{y}_i;$$

$$\Sigma_{\boldsymbol{x}} = \frac{1}{N_{\text{train}} - 1} \sum_{i \in \text{training set}} (\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{x}_i - \bar{\boldsymbol{x}})', \qquad \sigma_{\boldsymbol{y}}^2 = \frac{1}{N_{\text{train}} - 1} \sum_{i \in \text{training set}} (\boldsymbol{y}_i - \bar{\boldsymbol{y}})^2. \tag{12}$$

2.5.2. Inhomogeneous Gaussian kernel with varying covariance. When the training samples differ in their uncertainty, which may be caused by varying local noise level, one could assign different covariances to different samples. In particular, samples with higher uncertainty should use a kernel with higher covariance, corresponding to a 'more flat' local kernel. With Σ_i being the covariance for the local kernel around sample $\boldsymbol{z}_i$, the conditional distribution in (4) becomes

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{x}) &= \frac{1}{M} \sum_i \kappa((\boldsymbol{x}, \boldsymbol{y})|\boldsymbol{z}_i) \\ &= \frac{1}{M} \sum_i \frac{1}{(2\pi)^{n/2} |\Sigma_{x,i}|^{1/2} \sigma_{y,i}} \exp\left[-(\boldsymbol{x} - \boldsymbol{x}_i)^T \Sigma_{x,i}^{-1} (\boldsymbol{x} - \boldsymbol{x}_i)\right] \\ &\quad \times \exp\left[-\|\boldsymbol{y} - \boldsymbol{y}_i\| \big/ \sigma_y^2\right]. \end{aligned} \tag{13}$$

We modify the definition of sample weights w_i as

$$w_i = \frac{1}{|\Sigma_{x,i}|^{1/2} \sigma_{y,i}} \exp\left[-(\boldsymbol{x} - \boldsymbol{x}_i)^T \Sigma_{x,i}^{-1} (\boldsymbol{x} - \boldsymbol{x}_i)\right], \tag{14}$$

and the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ maintains the Gaussian mixture form as in (6). Subsequently, the mean estimate $\hat{\boldsymbol{y}}_{\text{mean}}$ is again a weighted sum of the sample response $\boldsymbol{y}_i$. The only difference incurred by allowing the local Gaussian kernels to have different covariances is that the local confidence level modifies the weights: less reliable training samples (with higher $|\Sigma_{x,i}|^{1/2}\sigma_{y,i}$) receive less weight. In data-driven approaches, neighborhoods with few samples have higher uncertainty, and it is desirable to 'flatten' out the local kernel. This approach is similar in spirit to the variable kernel method in Silverman (1986).

2.5.3. Robustness to outliers in training samples. Training samples during abrupt (and non-repetitive) changes, such as patient coughing, are less representative of the general covariate–response behavior and may be regarded as outliers. To increase the robustness of the kernel density estimator to such outliers, one should decrease the contribution of a training sample if its behavior differs significantly from other samples. We follow a similar philosophy of iterative weight assignment for robust local weighted regression (Cleveland 1979, Ruan *et al* 2007) and adjust the weight of a sample in the kernel density estimation as follows.

Let $B(\cdot)$ be a scalar function that satisfies

- $B(x) > 0$ for $|x| < 1$ and $B(x) = 0$ for $|x| \geqslant 1$,
- $B(x) = B(-x)$,
- $B(x)$ is non-increasing for $x \geqslant 0$.

Let $I = \{1, 2, \ldots, M\}$ be the complete index set. Denote the kernel approximation of $p(\boldsymbol{z})$ with samples $\boldsymbol{z}_j$, for $j \in I \setminus \{i\}$, termed *leave-one-out* density, as

$$p_i(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{M-1} \sum_{j \in I, j \neq i} \kappa((\boldsymbol{x}, \boldsymbol{y})|\boldsymbol{z}_j).$$

Conditioned on $X = \boldsymbol{x}_i$, one obtains a distribution of $p_i(\boldsymbol{y}|\boldsymbol{x}_i)$, and subsequently an estimate $\hat{\boldsymbol{y}}_i$ using any chosen estimate in section 2.3. Let $e_i = \boldsymbol{y}_i - \hat{\boldsymbol{y}}_i$ be the residual of the observed sample response from the estimated response with the leave-one-out density. Let ρ be a scale parameter, which could be set as the medium of $|e_i|$'s for $i = 1, 2, \ldots, M$. We define the robust weight by

$$\delta_i = B(\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i/\rho). \tag{15}$$

The original kernel density approximation (1) can then be modified by

$$p(z) = \frac{1}{\sum_{i=1}^{M} \delta_i} \sum_{i=1}^{M} \delta_i \kappa(\boldsymbol{z}|\boldsymbol{z}_i) \tag{16}$$

to incorporate the robust weighting.

2.5.4. Modified kernel weight to account for temporal correlation. By the same token as in section 2.5.3, one could incorporate the temporal correlation between the training samples and the test sample by modifying (16) further with

$$p(z) = \sum_{i=1}^{M} \delta_i \eta_i \kappa(\boldsymbol{z}|\boldsymbol{z}_i), \tag{17}$$

where η_i is a monotonically decreasing function of the temporal distance between sample $\boldsymbol{z}_i$ and $\boldsymbol{z}$. Fading memory is often modeled with exponential discounting or windowed training.

- For exponential discounting,

$$\eta_i = \exp(-\alpha|t_i - t|), \tag{18}$$

  where t_i and t are the time tags associated with $\boldsymbol{z}_i$ and $\boldsymbol{z}$, respectively. The positive constant α determines the decay rate of influence of one sample on another as their temporal distance increases.
- For moving window:

$$\eta_i = \begin{cases} 1 & |t - t_i| < \Gamma, \\ 0 & \text{else}, \end{cases} \tag{19}$$

  where Γ is the window size. Here, only training samples close enough in time to the test sample are used to estimate the pdf. The window size Γ should be chosen large enough to ensure a reasonable kernel approximation of the pdf.

2.6. Benchmark methods for comparison

Sampling rate and lookahead length are the two major prediction parameters in adaptive image-guided radiotherapy. We desire prediction algorithms that can handle low sampling rates and large lookahead lengths: a low sampling rate means less imaging dose and a large lookahead length allows more time for observation acquisition, signal processing and delivery. We will study the performance of the proposed method, with varying sampling rates and lookahead lengths and compare the outcome with the following benchmarks.

- Most recent sample (no prediction)

$$\hat{\boldsymbol{y}}_k = s_k. \tag{20}$$

- Adaptive linear predictor

$$\hat{\boldsymbol{y}}_k = \beta_k^T \boldsymbol{x}_k + \gamma_k, \tag{21}$$

where the linear coefficients β_k and γ_k are obtained by solving the least-squares problem for the observed covariate–response pairs in a dynamically updated training set. More specifically, at each instant k, a training set of covariate–response pairs is constructed with the most recent observed samples, and the prediction coefficients are determined by

$$(\hat{\beta}_k, \hat{\gamma}_k) = \arg\min \sum_{i \in \text{training set for time } k} (y_i - \beta_k \boldsymbol{x}_i - \gamma_k)^2,$$

which can be solved in closed form.

3. Materials and results

3.1. Material

We used the real position management system (RPM system, Varian Medical, Palo Alto, CA) to obtain 1D traces of fiducial markers placed on the patient's chest wall. The RPM traces are believed to be highly correlated with respiratory motion and sufficiently capture the temporal behavior of respiration. Moreover, the performance of respiratory prediction algorithms depends on the fundamental variation pattern rather than the amplitude, so the RPM traces are reasonable test subjects for algorithmic development.

To rid the adverse impact of the arbitrary scaling in RPM amplitude, we adopt the normalized root mean squared error (nRMSE) as the performance measure for each trace, defined by the usual RMSE divided by the standard deviation of the observed sample values:

$$\text{nRMSE}\big(\{\hat{\boldsymbol{y}}\}_{i=1}^N\big) = \frac{\text{RMSE}}{\text{std}_{\boldsymbol{y}}} = \frac{\sqrt{E((\boldsymbol{y} - \hat{\boldsymbol{y}})^2)}}{\sqrt{E((\boldsymbol{y} - \bar{\boldsymbol{y}})^2)}}. \tag{22}$$

Population nRMSE (across traces) is computed by taking the L_2 average of the trace-wise nRMSE, i.e.,

$$\text{nRMSE} = \frac{1}{\text{number of traces}} \sum_{i:\text{trace id}} \text{nRMSE}_i^2.$$

We report the RPM data characteristics in table 1 and illustrate some traces in figure 2.

Table 1. RPM dataset information.

Subject ID	1	2	3	4	5	6	7	8	9	10
STD	0.49	0.50	0.30	0.20	0.32	0.59	0.07	0.23	0.28	0.11
P-P	2.54	2.36	1.27	1.12	1.87	0.97	0.29	0.88	1.22	0.43
P-P/STD	5.11	4.74	4.21	5.66	5.92	5.61	4.59	3.88	4.44	4.05
Duration (s)	140	79	113	165	165	117	150	165	160	162

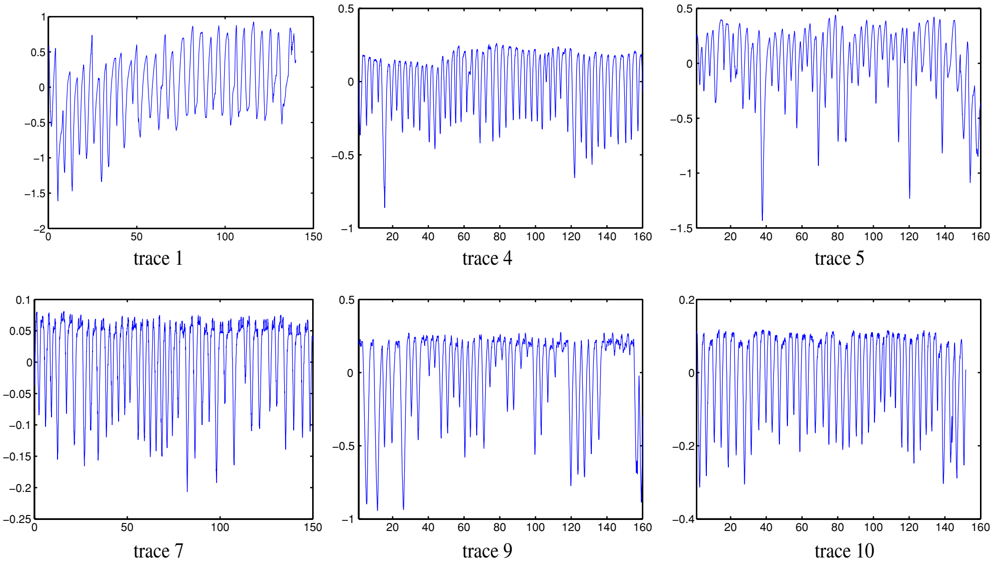


Figure 2. Typical RPM traces used in this study.

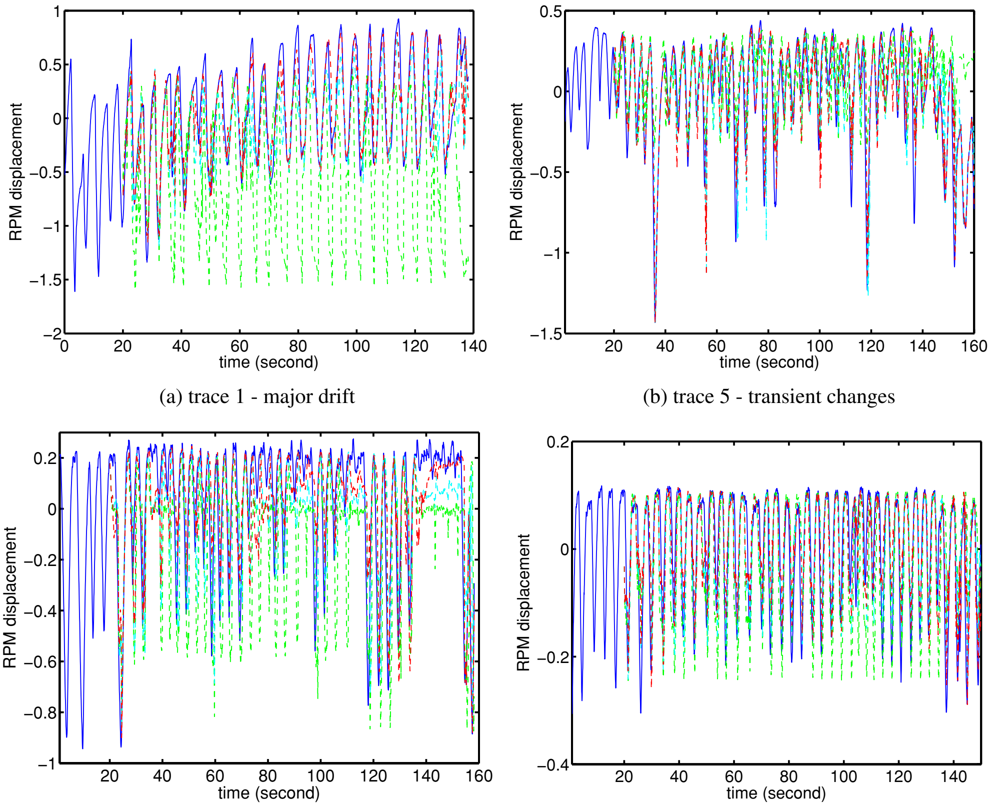
3.2. Experiment detail and results

We have chosen to use a three-dimensional augmented covariate variable, so that $\boldsymbol{x}_i = [s_{i-2\Delta}, s_{i-\Delta}, s_i]$. This low-dimensional decision was based on the consideration that we would like to obtain a reasonable kernel density estimation from limited samples and avoid the 'curse of dimensionality' in density estimation. This setup is also designed to ensure fairness in performance comparison with the benchmark methods.

To properly capture the motion dynamics, we choose Δ to correspond to 0.4 s delay between consecutive coordinates of the covariate $\boldsymbol{x}$. In the baseline setup, we used a sampling rate of 30 Hz, with $\Delta = 12$. As mentioned in section 2.5.1, we are most interested in a predictor that could perform well after a short training stage under low sampling conditions, so we set the covariance in (3) with the estimated population covariance from the training samples as in (12): this covariance setup corresponds to an overly broad (flat) local kernel, but it provides numerical stability in most cases, an issue discussed in section 4 . We investigated a lookahead length $L = 30$ corresponding to a 1 s prediction, which has been reported to be challenging for a wide spectrum of common prediction techniques (Vedam *et al* 2003, Sharp *et al* 2004, Murphy and Dieterich 2006).

For a fair comparison, the adaptive linear filter uses the same covariate variables, and the observed covariate–response pairs in the most recent 20 s are used to estimate the linear regression coefficients $\beta_k \in \Re^3$ and $\gamma_k \in \Re$ at each instant k.

3.2.1. Training samples in kernel density estimation. We studied three different schemes in choosing the training samples used in kernel density estimation. In the static scheme, the first 20 s of the trajectory was used to generate the training sample collection, and the estimated pdf was kept still thereafter for all predictions. In the expansive scheme, the training set is enriched as new covariate–response pairs were observed. This corresponds to a special case of fading memory (cf section 2.5.4) where all previous samples contribute equivocally to the kernel approximation ($M = k - L$, $\alpha = 0$ in (17) and (18)). In the moving window update scheme, only training samples within the most recent 20 s temporal window were used to



(a) trace 1 - major drift

(b) trace 5 - transient changes

(c) trace 9 - irregular

(d) trace 10 - quasi regular

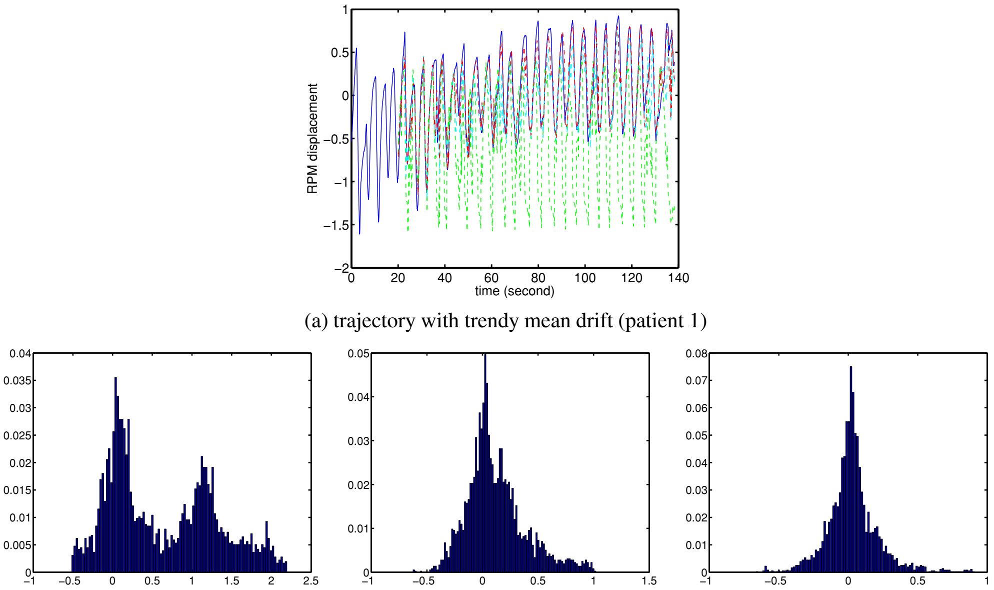
Figure 3. Comparison among 1 s lookahead prediction results with different training schemes in kernel density estimation. Actual signal trajectory (solid blue line), prediction with the static training scheme (dashed green line), prediction from the expansive training scheme (dashed cyan line), prediction from the moving window training scheme (dashed red line).

Table 2. Comparison of prediction performance among static training, expansive training and moving window training.

Subject ID	1	2	3	4	5	6	7	8	9	10	Average
					Root mean squared error (RMSE)						
Static	1.85	0.81	0.73	0.91	0.99	1.03	0.83	0.54	1.05	0.85	1.02
Expansive	0.58	0.49	0.48	0.64	0.56	0.54	0.72	0.41	0.72	0.63	0.59
Moving window	0.37	0.37	0.39	0.58	0.45	0.42	0.69	0.35	0.60	0.56	0.49

construct the kernel pdf, as formulated in (19). Table 2 reports the prediction performance of these three training schemes and figure 3 illustrates some typical prediction trajectories with these training schemes.

It is quite obvious that expanding the training samples, or even better, using a moving window to select the training sample collection, improves the prediction performance. This is no surprise, as the updated pdf would drive the final prediction estimate to resemble the training samples that both behave similarly and are close in time. This is particularly true for trajectories with drifting, as shown in figure 4.



(a) trajectory with trendy mean drift (patient 1)

(b) static training

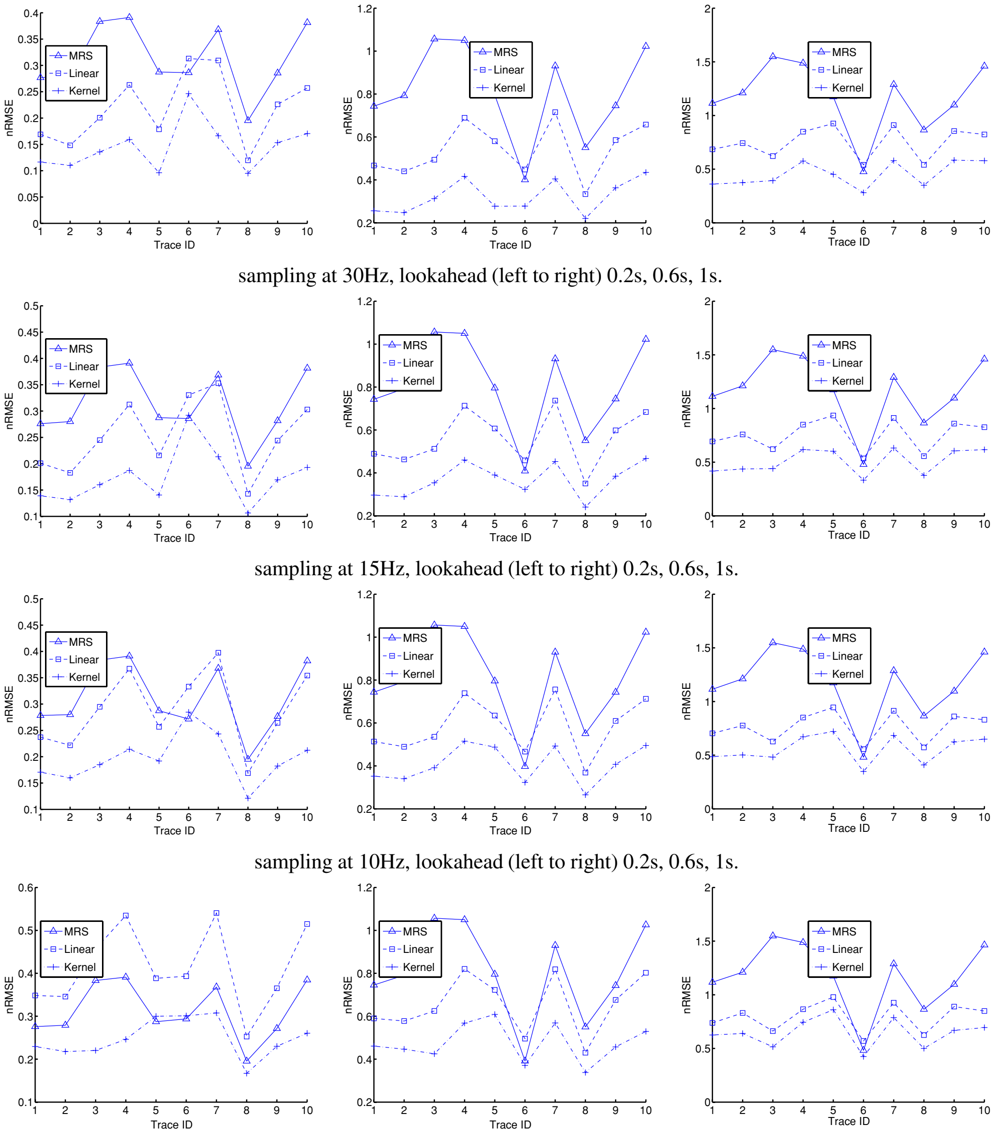
(c) expansive training

(d) moving-window training

Figure 4. Comparison among 1 s lookahead prediction results with different training samples in kernel density estimation for drifting trajectory. Top row: (a) time series of actual signal trajectory (solid blue line), prediction with the static training scheme (dashed green line), prediction from the expansive training scheme (dashed cyan line), prediction from the moving window training scheme (dashed red line). Bottom row: histogram of prediction residual $y_k - \hat{y}_k$ for (b) static training; (c) expansive training ; (d) moving window training.

3.2.2. The effect of sampling rate and lookahead length. Since sampling rate and lookahead length are two of the most critical parameters in a prediction system, we compared the performance of the proposed method with the most recent sample and the adaptive linear filter prediction method as described in section 2.6, for various sampling rates and lookahead lengths. In particular, we tested prediction performances for lookahead lengths 0.2 s, 0.6 s and 1 s when samples are acquired at 5 Hz, 10 Hz, 15 Hz and 30 Hz respectively. Figure 5 reports the nRMSE for the three methods (MRS, Linear, Kernel) under different combinations of prediction parameters. In figure 5, the upper-left corner corresponds to the relatively easy case of short prediction (0.2 s) with dense samples (30 Hz). The sampling rate decreases as we move down the rows and the prediction length increases as we move to the columns on the right—both changes increases the difficulty of prediction.

In most cases, the adaptive linear filter yields significant accuracy gain compared to the MRS, usually reducing the nRMSE by about a half. This performance is comparable to what is reported by current development of respiratory predictors, justifying our previous argument that the adaptive linear predictor is a fair benchmark for this study. The only exception occurs when 5 Hz samples are used to predict 0.2 s ahead. This can be explained from two aspects: (1) 200 ms delay is relatively short and does not incur too high an nRMSE even using MRS; (2) the low sampling rate basically renders a down-sampled path, causing relative slow response of the adaptive linear filter to dynamic changes. Meanwhile, the proposed kernel-based method universally dominates the alternatives on a trace-to-trace basis, always yielding the lowest nRMSE value. Moreover, its improvement upon RMS and adaptive linear predictor is quite significant.



sampling at 30Hz, lookahead (left to right) 0.2s, 0.6s, 1s.

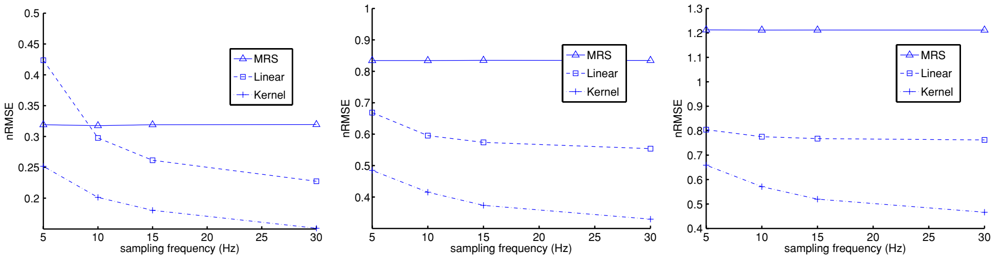
sampling at 15Hz, lookahead (left to right) 0.2s, 0.6s, 1s.

sampling at 10Hz, lookahead (left to right) 0.2s, 0.6s, 1s.

sampling at 5Hz, lookahead (left to right) 0.2s, 0.6s, 1s.

Figure 5. Comparison of prediction performance in terms of nRMSE under various prediction parameters. Column-wise (left → right): lookahead = 0.2 s, 0.6 s, 1 s. Row-wise (up → down): sampling rate = 30 Hz, 15 Hz, 10 Hz, 5Hz.

Figure 6 reports the nRMSE across all traces with varying sampling rates for lookahead lengths 0.2 s, 0.6 s and 1 s respectively, and figure 7 reports the performance subject to varying lookahead lengths, for each given sampling rate. The advantage of the proposed method is universal across all prediction lengths and all sampling rates and is most dramatic for long lookahead length when the alternative methods yield essentially useless prediction (nRMSE $\geqslant 1$). The nonparametric nature and data-driven learning makes the kernel-based prediction robust toward the changes in the prediction parameters.

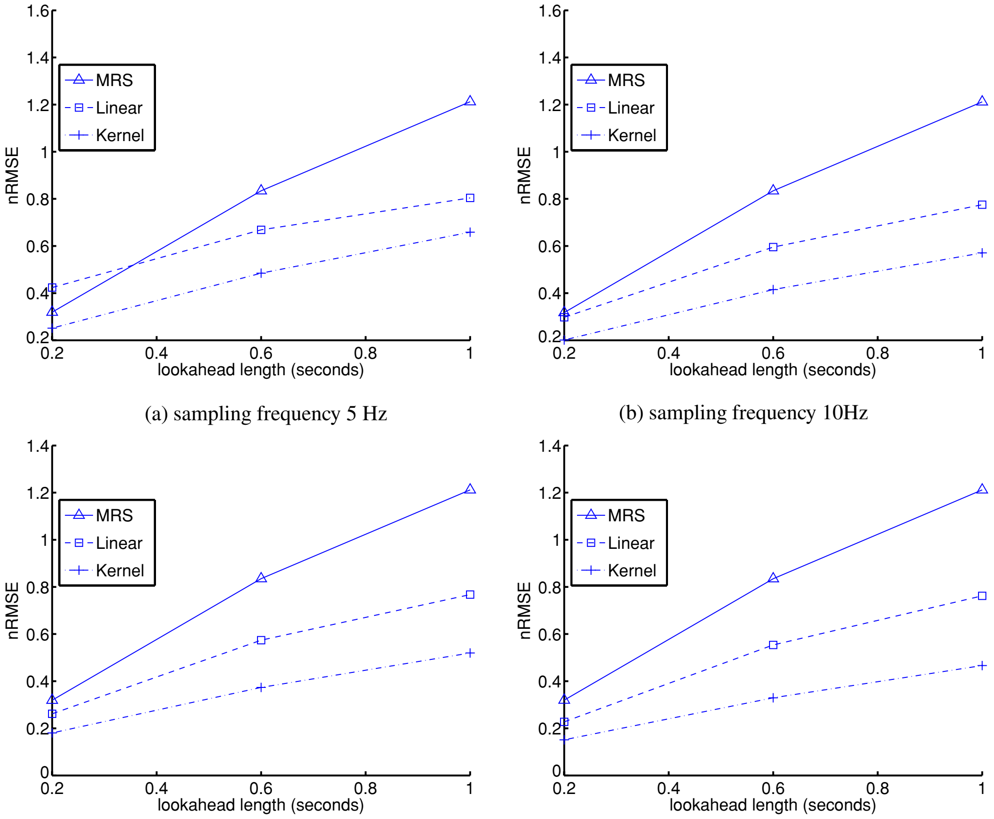


(a) lookahead 0.2 s

(b) lookahead 0.6 s

(c) lookahead 1 s

Figure 6. nRMSE versus sampling frequency for 0.2, 0.6 and 1 s lookahead prediction across all traces.



(a) sampling frequency 5 Hz

(b) sampling frequency 10Hz

(c) sampling frequency 15 Hz

(d) sampling frequency 30 Hz

Figure 7. nRMSE versus lookahead length with various sampling rates across all traces.

4. Discussions

- The Gaussian mixture structure in (6) offers an efficient form to store and process the pdf. It suffices to record the weights w_i, the Gaussian centers y_i and the variance σ_y to fully recover the probability distribution. Furthermore, the Gaussian kernel with separable covariance (with respect to covariate and response variables), together with the linearity in the mean estimate allows us to obtain the prediction $\hat{\boldsymbol{y}}_{\text{mean}}$ as a simple weighted sum of sample response values (8), where the weight of each training sample is determined by how close its covariate value is to the covariate value of the test sample.

- When there are only a few training samples (with respect to the covariate–response space under study) available, numerical instability may occur if each of the weights w_i in (5) is small relative to the machine precision. A 'zero divided by zero' fault may arise from normalization. Geometrically, uniformly small weights indicate that the test sample is far from all training samples. One could utilize this observation to detect 'unseen' changes in the trajectory. From the perspective of obtaining a stable estimate, one could adjust the covariance Σ_x in (5) to 'flatten' the local kernel and increasing its bandwidth, so the support of the kernel density approximation from the same training samples covers a larger portion of the whole space, preventing uniformly vanishing $p(\boldsymbol{y}|\boldsymbol{x})$. As an alternative, one could use a finitely supported density kernel and determine a test-sample-dependent bandwidth by requesting the test sample to fall inside the kernel support of a certain number of training samples, as in Cleveland (1979), Silverman (1986), Ruan *et al* (2007).

5. Conclusion and future work

This paper reports a kernel density-based method to estimate the probability distribution of the future target position. We provide a general framework for estimating the conditional probability distribution and several options in constructing estimates based on the conditional distribution. In particular, we have discussed the use of the Gaussian density kernel and the mean estimate, leading to a simple yet intuitive estimate that is the weighted sum of the training response values. Our proposed method compares favorably with alternative benchmark methods, yielding a reduction of two- to threefolds in RMSE. Improvement of the proposed method is most noticeable in the case of low sampling rate and/or long lookahead length prediction.

We have discussed how numerical instability could be indicative of an 'unseen' scenario, and may be used for change detection. More generally, the marginal distribution of the test covariate $p(\boldsymbol{x})$ indicates the chances of such event and the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ provides all the information for characterizing the quality of the resulting estimate. This gives rise to an instantaneous quantification of the prediction, which is an interesting alternative to the conventional collective performance measure. We will study the performance quantification issue as a natural extension of this work.

As we described in our methodology and observed in our experiment, the choice of the covariance for the kernel density is not particularly critical for the purpose of prediction. However, a good estimation of the joint distribution of the covariate–response is valuable for its own sake. Related to the previous comment, change detection relies on a good estimate of the marginal distribution. We will study automatic schemes to 'optimally' choose the kernel covariance.

We have formulated the mode and medium estimates in section 2.3, but have focused on the mean estimate due to its simplicity and the resulting intuitive weighted superposition form for the predictor. However, the mode estimate corresponds to the maximum *a posteriori* (MAP) estimate and the medium estimate has its advantage toward outlier samples, and have their own significance. In the future, we will analyze these estimates and investigate their applicability in various settings.

Acknowledgments

The author thanks Dr Per Poulsen and Dr Byung-Chul Cho for motivating this work and Dr Paul Keall for his support. She is grateful to the reviewers for their insightful comments that greatly improved the quality of this work.

References

Botev Z I 2006 A novel nonparametric density estimator *Postgraduate Seminar Series* Department of Mathematics, The University of Queensland

Cleveland W S 1979 Robust locally weighted regression and smoothing scatterplots *J. Am. Stat. Assoc.* **74** 829–36

Duda R O, Hart P E and Stork D G 2001 *Pattern Classification* (New York: Wiley)

Ernst F, Schlaefer A and Schweikard A 2007 Prediction of respiratory motion with wavelet-based multiscale autoregression *Med. Image Comput. Comput. Assist. Intervention* **10** 668–75

Isaksson M, Jalden J and Murphy M J 2005 On using an adaptive neural network to predict lung tumor motion during respiration for radiotherapy applications *Med. Phys.* **32** 3801–9

Kakar M, Nystrom H, Aarup L R, Nottrup T J and Olsen D R 2005 Respiratory motion prediction by using the adaptive neuro fuzzy inference system (ANFIS) *Phys. Med. Biol.* **50** 4721–8

Keall P J, Kini V R, Vedam S S and Mohan R 2002 Potential radiotherapy improvements with respiratory gating *Australas Phys. Eng. Sci. Med.* **25** 1–6

McCall K C and Jeraj R 2007 Dual-component model of respiratory motion based on the periodic autoregressive moving average (periodic ARMA) method *Phys. Med. Biol.* **52** 3455–66

McMahon R, Papiez L and Sandison G 2007 Addressing relative motion of tumors and normal tissue during dynamic MLC tracking delivery *Australas Phys. Eng. Sci. Med.* **30** 331–6

Murphy M J and Dieterich S 2006 Comparative performance of linear and nonlinear neural networks to predict irregular breathing *Phys. Med. Biol.* **51** 5903–14

Nuyttens J J, Prevost J B, Praag J, Hoogeman M, Van Klaveren R J, Levendag P C and Pattynama P M 2006 Lung tumor tracking during stereotactic radiotherapy treatment with the cyberknife: marker placement and early results *Acta Oncol.* **45** 961–5

Putra D, Haas O C, Mills J A and Burnham K J 2008 A multiple model approach to respiratory motion prediction for real-time IGRT *Phys. Med. Biol.* **53** 1651–63

Ruan D, Fessler J A and Balter J M 2007 Real-time prediction of respiratory motion based on nonparametric local regression methods *Phys. Med. Biol.* **52** 7137–52

Sharp G C, Jiang S B, Shimizu S and Shirato H 2004 Prediction of respiratory tumour motion for real-time image-guided radiotherapy *Phys. Med. Biol.* **49** 425–40

Sheather S J and Jones M C 1991 A reliable data-based bandwidth selection method for kernel density estimation *J. R. Stat. Soc.* B **53** 683–90

Silverman B W 1986 *Density Estimation for Statistics and Data Analysis* (New York: Chapman and Hall)

Vedam S S, Kini V R, Keall P J, Ramakrishnan V, Mostafavi H and Mohan R 2003 Quantifying the predictability of diaphragm motion during respiration with a noninvasive external marker *Med. Phys.* **30** 505–13

Vedam S S, Keall P J, Docef A, Todor D A, Kini V R and Mohan R 2004 Predicting respiratory motion for four-dimensional radiotherapy *Med. Phys.* **31** 2274–83

Zimmerman J, Korreman S, Persson G, Cattell H, Svatos M, Sawant A, Venkat R, Carlson D and Keall P 2008 DMLC motion tracking of moving targets for intensity modulated arc therapy treatment—a feasibility study *Acta Oncol.* 1–6