IOP PUBLISHING PHYSICS IN MEDICINE AND BIOLOGY

Phys. Med. Biol. **52** (2007) 6511–6524 doi:10.1088/0031-9155/52/21/012

Towards the development of an error checker for radiotherapy treatment plans: a preliminary study

Fatemeh Azmandian[1], David Kaeli[1], Jennifer G Dy[1], Elizabeth Hutchinson[2], Marek Ancukiewicz[2], Andrzej Niemierko[2] and Steve B Jiang[2,3]

[1] Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA
[2] Department of Radiation Oncology, Massachusetts General Hospital and Harvard Medical School, Boston, MA 02114, USA
[3] Department of Radiation Oncology, University of California San Diego, La Jolla, CA 92093, USA

E-mail: sbjiang@ucsd.edu

Received 19 July 2007, in final form 25 September 2007
Published 16 October 2007
Online at stacks.iop.org/PMB/52/6511

Abstract

Major accidents can happen during radiotherapy, with an extremely severe consequence to both patients and clinical professionals. We propose to use machine learning and data mining techniques to help detect large human errors in a radiotherapy treatment plan, as a complement to human inspection. One such technique is computer clustering. The basic idea of using clustering algorithms for outlier detection is to first cluster (based on the treatment parameters) a large number of patient treatment plans. Then, when checking a new treatment plan, the parameters of the plan will be tested to see whether or not they belong to the established clusters. If not, they will be considered as 'outliers' and therefore highlighted to catch the attention of the human chart checkers. As a preliminary study, we applied the *K*-means clustering algorithm to a simple patient model, i.e., 'four-field' box prostate treatment. One thousand plans were used to build the clusters while another 650 plans were used to test the proposed method. It was found that there are eight distinct clusters. At the error levels of ±100% of the original values of the monitor unit, the detection rate is about 100%. At ±50% error level, the detection rate is about 80%. The false positive rate is about 10%. When purposely changing the beam energy to a value different from that in the treatment plan, the detection rate is 100% for posterior, right-lateral and left-lateral fields, and about 77% for the anterior field. This preliminary work has shown promise for developing the proposed automatic outlier detection software, although more efforts will still be required.

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

0031-9155/07/216511+14$30.00

1. Introduction

Radiotherapy is unique from the point of view of radiation safety, since it is the only application of radiation sources in which very high doses are given on purpose to a part of a human body. Therefore, extreme caution is often exercised during radiotherapy (ICRP 1973, 1991, ICRU 1993). However, incidents that compromise patient safety still happen occasionally, caused by either human errors or equipment malfunction (Calandrino *et al* 1993 1997, Ostrom *et al* 1996, Yeung *et al* 2005, Ekaette *et al* 2006).

Complete avoidance of human errors is difficult, if not impossible, due to the fact that radiotherapy is a very complex process and there are a large number of steps from the prescription of the treatment to the delivery of the dose (Leunens *et al* 1992). Many records and communications are involved in those steps, between different professionals and even with the patient. There is a combination of very different activities from the very manual, to very sophisticated computer-assisted techniques and high technology equipments. The typical consequences of human errors include (1) treatment delivered to wrong patients, (2) treatment delivered to wrong treatment sites and (3) treatment delivered with a wrong dose.

When the treatment is delivered with a dose significantly different from that prescribed by clinicians, the treatment outcome can be seriously compromised. For example, if the delivered dose is too low, the tumor control probability is reduced, and if the dose is too high, acute and late complications can occur. The worst scenario is that, when the delivered dose is much higher than the prescribed dose, patients may die of the treatment. One example is the Panama incident (Akashi *et al* 2001), where 16 patients were severely overexposed (approximately twice the prescribed dose) in late 2000 and early 2001, resulting in eight treatment-related death. Another more recent incident happened in early 2006, involving the death of a young female patient who received a radiation dose of 58% higher than that intended while undergoing a course of radiotherapy at the Beatson Oncology Center (BOC) in Glasgow (Johnston 2006).

Many lessons can be learned from those incidents (Ostrom *et al* 1996, IAEA 2007). Although the probability of major incidents is very low, the consequence is extremely severe to both patients and clinical professionals. Therefore, the level of awareness of the potential for accidental radiation exposures should be raised and effective plans for the prevention of accidental exposures should be developed.

Widely adopted practices in the radiotherapy community for eliminating such incidents include independent calculation of monitor units (MU) and a manual check of the data reported in the treatment chart. These approaches have been proven to be effective in greatly reducing the occurrence of systematic errors before treatment delivery (Kutcher *et al* 1994, Calandrino *et al* 1997, Fraass *et al* 1998). Other effective approaches include the use of portal imaging or *in vivo* dosimetry before or during the treatment to detect errors (Essers and Mijnheer 1999, Yeung *et al* 2005). Recently, some efforts have been made to develop systematic approaches for collecting, processing and reporting incidents (Yeung *et al* 2005, Ekaette *et al* 2006, Dunscombe *et al* 2007).

While the above-mentioned methods have been shown effective in clinical practice, there is still room for improvement. Manual chart checking can be more effective and efficient if aided with tools that can automatically highlight treatment parameters of suspicion. Independent MU calculation is not totally independent since it uses the same treatment setup parameters (such as the source to skin distance and the calculation depth) and patient geometry data as the planning system. *In vivo* and portal dosimetry also rely on the planning system to provide the reference dose values for the given treatment setup. Therefore, a method that is *completely independent* of the planning system would be complementary to those methods.

Such a method would potentially further prevent erroneous treatment delivery from happening when working together with existing methods, since it looks at the problem from a totally different angle.

In this paper, we propose such a new method. The proposed method, based on all the previous patient treatments, can detect potential catastrophic errors in a treatment plan and highlight them to get the clinical professional's attention. The main idea of this method can be summarized as follows:

(1) A piece of software can reside in a record and verification (R&V) system, checking all the treatment parameters in the background after a new plan is uploaded into the R&V system. Therefore, to clinical professionals, this outlier detection method is effortless.
(2) The outlier treatment parameters are detected using data mining and machine learning techniques. The method is totally independent of the dose calculation system, and is not another monitor unit calculation algorithm.
(3) When an outlier treatment parameter is detected, the software will highlight the parameter to assist the radiotherapy professionals to quickly spot errors. The software is designed to improve the accuracy and efficiency of human chart checkers. It is not intended to replace the human experts for checking treatment plans, nor to replace any monitor unit verification algorithms.
(4) The method is aimed at catastrophic errors, not small errors.

While various data mining and machine learning techniques might be suitable for the proposed method, in this paper we present a preliminary study based on a computer-clustering algorithm. The goal of this preliminary study is to show the proof of principle for the proposed concept, using a simple patient model: prostate patients treated with a 'four-field box' technique.

2. Methods and materials

The basic idea of the proposed computer-clustering-based treatment plan error checker is to first cluster the treatment parameters for a large number of patients having been treated previously. Then, when checking a new treatment plan, the parameters of the plan will be tested to see whether or not they belong to the established clusters. If not, they will be considered as 'outliers' and therefore highlighted to catch the attention of the human experts. In the following sections, we will first describe the patient data and the clustering method used in this preliminary work, and then we will use intentionally added errors to test the proposed outlier detection method.

2.1. Patient data

To show the proof of principle for the concept for a treatment plan error checker, only a simple treatment technique was studied in this work. Data for 1650 prostate cancer patients treated from 1995 to 2005 at Massachusetts General Hospital (MGH) in Boston with the 'four-field box' technique were used. Both primary and boost treatments were included. A computer code was written to extract all the entries of a patient treatment plan from the IMPAC record and verification system (IMPAC Medical Systems, Inc., Sunnyvale, CA). In this study, to simplify the model, we only consider the most significant eight entries, which are the beam energies and monitor units (MUs) for the four radiation fields for each patient. Those eight entries are the so-called 'features' of a treatment plan in the terminology of computer clustering. They are referred to as E_{AP}, E_{PA}, E_{RL}, E_{LL}, $\mathrm{MU}_{\mathrm{AP}}$, $\mathrm{MU}_{\mathrm{PA}}$, $\mathrm{MU}_{\mathrm{RL}}$ and $\mathrm{MU}_{\mathrm{LL}}$ for beam energies and

MUs for anterior–posterior (AP), posterior–anterior (PA), right-lateral (RL) and left-lateral (LL) fields, respectively. The beam energies of the linear accelerators used for treating those patients are 6, 10, 18 and 23 MV. The monitor units ranged from about 50 to over 200. To provide equal weight for all the features, we normalize them before clustering. All the features are normalized to have zero mean and unit standard deviation.

2.2. Data clustering

Clustering is a data-mining and machine-learning technique that is used to extract valuable information from a set of unlabeled data (Fayyad 1996, Jain *et al* 1999). It is one of the most important data-mining methods applied to discover patterns and relations in complex medical datasets (Greene *et al* 2004). The goal of clustering is to separate data into groups, called clusters, such that objects in the same cluster are similar to each other and dissimilar to objects in other clusters.

The clustering method we use is *K*-means clustering (Forgy 1965, MacQueen 1967) (also known as Lloyd's algorithm). *K*-means strives to minimize the sum-squared-error (SSE) criterion, which is the sum of the squared distance of each data point to its closest cluster center:

$$\mathrm{SSE} = \sum_{j=1}^{k} \sum_{x \in C_j} [D(x, \mu_j)]^2, \tag{1}$$

where C_j is the jth cluster, μ_j is the center of the jth cluster, and D stands for the distance between the two points. Each data point, x, is a vector described by the eight features (four beam energy values and four monitor units). The distance measure we use is the Euclidean distance.

The main steps in the *K*-means algorithm are provided below:

(1) Initialization: choose the initial centers for *k* clusters.
(2) Assignment step: assign each data point to its nearest cluster based on its distance to the cluster center.
(3) Estimate cluster means: recalculate the cluster centers by taking the average of the data points within each cluster.
(4) Repeat steps 2 and 3 until some convergence criterion is met (usually that the assignment has not changed or that the difference in SSE from one assignment to the next is less than some predetermined threshold).

In terms of implementation, *K*-means has a couple of issues: (1) how to automatically determine the number of clusters, and (2) *K*-means is a greedy algorithm and as such only guarantees to reach a local optimum depending on the initialization (initial selection of the *k* cluster centers).

2.2.1. Determining the number of clusters. To select the number of clusters, we utilized the Bayesian information criterion (BIC) (Schwarz 1978). Note that as we increase the number of clusters, *k*, the sum-squared-error will also decrease. If we use SSE as the criterion for choosing *k*, it will end up with the trivial result of *k* equal to the number of data points, with each data point as a cluster center. Some form of penalty should be subtracted for more complex (large *k*) models. BIC is commonly used to penalize for model complexity in maximum likelihood estimation for model selection (such as, selecting *k*) to avoid over-fitting.

X-means (Pelleg and Moore 2000) utilized BIC for model selection in K-means. We also apply BIC for model selection as follows:

$$\text{BIC score} = \ln p(X|\theta) - \frac{M}{2}\ln N, \tag{2}$$

where $p(X|\theta)$ is the likelihood of the data $X = \{x_1, \ldots, x_N\}$ given the model parameters θ, M is the number of free parameters, $N$is the number of data points and ln is the natural logarithm. To compute the likelihood, we assume a finite mixture of Gaussian distributions for X, $p(x_i) = \sum_{j=1}^{k} \pi_j p(x_i|\theta_j)$, with π_j as the proportion of data points belonging to cluster j and θ_j as the parameters (mean and covariance) for cluster j. We set the means to be the k centroids in K-means. We estimate the covariance matrix (Σ) of each cluster using the sample covariance matrix of the cluster found by K-means.

$$\hat{\Sigma} = \frac{1}{N-1}\sum_{x\in C_j}(x-\mu_j)(x-\mu_j)^T, \tag{3}$$

where T is the matrix transpose operation. For the mixture weights, π_j, we use the relative size of the cluster (the number of data points in the cluster divided by the total number of data points).

The formula for the log-likelihood is provided below:

$$L(\theta) = \ln p(X|\theta) = \sum_{i=1}^{N}\ln p(x_i) = \sum_{i=1}^{N}\ln\left(\sum_{j=1}^{k}\pi_j p(x_i|\theta_j)\right). \tag{4}$$

With our mixture of Gaussian assumptions, the point probability of each data point is calculated as

$$p(x_i|\theta_j) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_j|^{\frac{1}{2}}}\,\mathrm{e}^{-\frac{1}{2}(x_i-\mu_j)^T\Sigma_j(x_i-\mu_j)}, \tag{5}$$

where d is the dimension (number of features) of the data. The number of free parameters, M, in our model is $(k-1) + kd + k\frac{d(d+1)}{2}$ where $k-1$ is the number of mixture weights, kd is the number of parameters for the mean vectors, and $k\frac{d(d+1)}{2}$ is the number of parameters for the covariance matrices.

Figure 1 shows the log-likelihood and the BIC score with respect to the number of clusters k. Note that as k increases, the log-likelihood increases, but the maximum BIC score falls at the knee of the log-likelihood curve demonstrating a good trade-off between model fit to the data (log-likelihood) and model complexity (size of k). As can be seen from the figure, the knee of the log-likelihood curve and the best value of the BIC score occur at $k = 8$. Therefore, eight clusters are sufficient to represent the data well.

Our log-likelihood computation is slightly different from that in X-means. X-means utilized a constrained covariance mixture model, where the covariances for each cluster were constrained to be the same and spherical. In addition, we do not use the same implementation as X-means. We applied regular K-means for different possible k from 2 to k-max and pick the k which provides us with the best BIC value. X-means extends K-means not only to select the number of clusters automatically, but also to make K-means computationally tractable by utilizing cached statistics and kd-trees. X-means would be useful for large volumes of data.

2.2.2. Initializing K-means. K-means clustering may be trapped into a local optimum. The quality of the final clustering solution depends on the initial selection of the k cluster centers. To avoid local minima, random restart is typically employed. However, random restart is not repeatable and one needs to run (restart) K-means several times to find a good solution. After

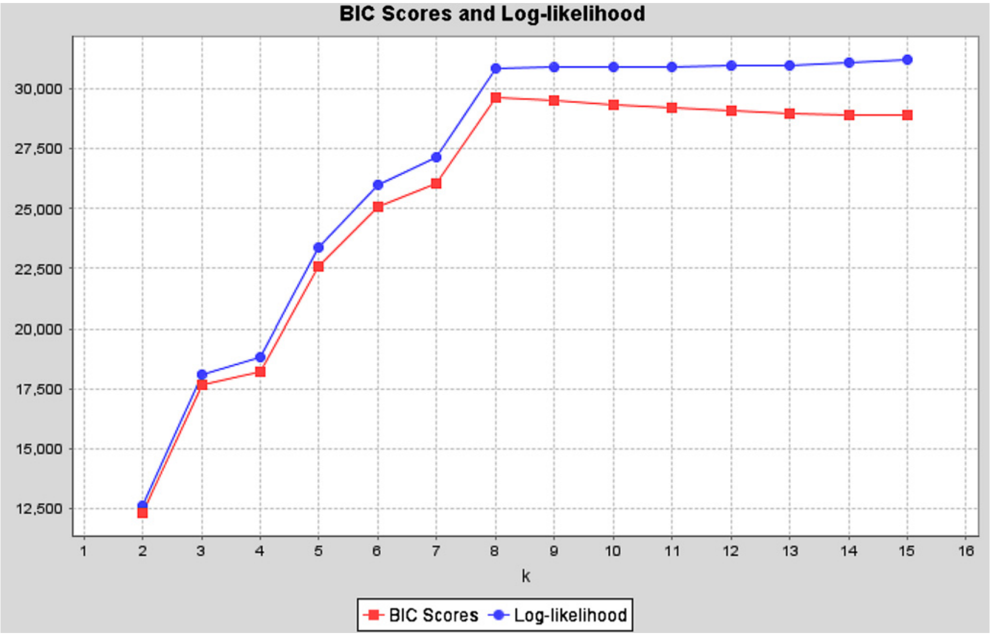

Figure 1. The BIC scores and log-likelihood of the data for different values of k.

exploring several techniques to initialize the K-means algorithm, we found a modified version of PCA-Part (Su and Dy 2004) to provide us with the best results for our data. PCA-Part is a deterministic initialization method that has been shown by Su and Dy to lead K-means to solutions that are close to optimum. Details about this algorithm can be found in Su and Dy (2004).

Principal component analysis (PCA) is a popular technique for dimensionality reduction (Jolliffe 2002). PCA finds a linear transformation, $Y = A^T X$, that projects the original high-dimensional data X with dimensions d to a lower dimensional data Y with q dimensions where $q < d$, such that the mean-squared error between X and Y is as small as possible. X, here is $d \times n$, where n is the number of data points, and A is a $d \times q$ matrix. The solution is the transformation matrix A whose columns correspond to the q eigenvectors with the q largest eigenvalues of the data covariance. It also projects the high-dimensional dataset to the lower dimensional subspace in which the original dataset has the largest variance (i.e., restricts attention to those directions along which the scatter of the data points is greatest).

PCA-Part is an initialization algorithm that hierarchically splits the data into two in the direction of the largest eigenvector (first principal axis) at each step until k clusters are obtained. K-means clustering aims to minimize the sum-squared-error criterion. The largest eigenvector with the largest eigenvalue is the direction which contributes to the largest sum-squared error. Hence, a good candidate direction to project a cluster for splitting is the direction of the cluster's largest eigenvector, which is the basis for PCA-Part initialization. Our modified algorithm integrates the K-means clustering algorithm into the PCA-Part initialization algorithm, in order to refine the result at each iteration. This modification helps the partitions converge to final clusters with smaller SSE values. The main steps in our modified algorithm are provided below:

(1) Given a value for k (the number of clusters).
(2) Start with all the data as comprising a single cluster, C.

(3) Find the first principal axis of cluster C and project the data points of C onto the first principal axis (Jolliffe 2002).

(4) Divide cluster C into two groups by separating data points in C based on whether its projected value is less than or greater than the mean of the projected data.

(5) Run the original K-means algorithm (as described above) on all the data, with the number of clusters equal to the number of partitions in the whole dataset and the initial cluster centers equal to the means of the partitions.

(6) Use the resulting clusters as the new partitions for the following steps.

(7) Find the SSE as defined in equation (1) for each cluster. Divide the SSE by the number of points in the cluster to find the mean-squared error (MSE) of each cluster.

(8) If there are k clusters, end clustering. Otherwise set C equal to the cluster with the largest MSE and go to step 3.

We modified the original PCA-Part in Su and Dy (2004) by adding step 5, and utilizing MSE in step 7 instead of SSE. We applied this modified PCA-Part to initialize K-means clustering.

2.3. Outlier detection

In this study, we have 1650 prostate cancer patient data. To avoid statistical bias, we randomly select 1000 patients as our training set and use the other 650 patients as the testing set. The training set was used to build clusters while the test set was used to test the model's outlier detection capability.

We assume that our training set comprises of normal ('correct') treatments. We apply K-means clustering to extract similarity groups from these data. In K-means clustering, we assume that each cluster comes from a Gaussian distribution. Since our training data are examples of normal ('correct') treatment, we test a new treatment instance as correct or an outlier by testing whether it belongs to any of our Gaussian clusters. We assign a rule of classifying a test treatment instance as an outlier if its Euclidean distance from the closest cluster center is greater than a threshold. Because we have a probability distribution model for each cluster, we can set the threshold to assure us of the probability of making a type I error or false positive (i.e., of deciding a point as an outlier when in fact it is normal) is smaller than α. In our experiments, we set the threshold to be 2 sigma (where sigma is the standard deviation), which assures us of the probability of a type I error to be less than 5%.

A summary of our outlier detection algorithm is as follows:

(1) For each cluster built based on the training set, calculate the mean and standard deviation of all the features (in this study we have eight features for each treatment plan).

(2) To check whether or not a new data point is an outlier, we first find a cluster whose center is the closest to the data point using the Euclidean distance, and then calculate the difference for each feature between the data point and the cluster center.

(3) If the difference is within a pre-set tolerance (e.g., two standard deviations of that cluster) for all the features, this data point is considered to belong to the cluster. Otherwise, classify the data point as an outlier.

To measure the quality of the clustering results and how well they could be used to identify outliers, we purposely introduce errors to the test set and use the outlier detection algorithm described above to compute the outlier detection rate. The outlier detection rate is defined as the ratio of the number of data points that are detected as outliers to the total number of data points tested. The outlier detection rates are computed at various error levels for MUs. The error level is defined as the deviation from the original value of a MU feature. For example, 10–20% error level means that the introduced errors are 10–20% of the original MU value.

We first introduce errors to MUs while keeping the energies untouched. The way we calculate the outlier detection rate is outlined below:

(1) Randomly select one patient data from the test set of 650 patient data.
(2) For the selected patient, randomly select one of the four MU features ($\mathrm{MU}_{\mathrm{AP}}$, $\mathrm{MU}_{\mathrm{PA}}$, $\mathrm{MU}_{\mathrm{RL}}$, or $\mathrm{MU}_{\mathrm{LL}}$).
(3) For the selected MU feature, randomly generate an error ranging from -100% to 100% of the original value of the MU feature.
(4) Add the error to the original value of the MU feature. Now the modified value of the MU feature ranges from 0 to twice the original value.
(5) Run the outlier detection algorithm as outlined above on the modified test data point to see whether or not this point will be detected as an outlier.
(6) Repeat steps 1 to 5 for 100 000 times, and calculate the outlier detection rate for every error level.
(7) Repeat steps 1 to 6 for 100 times, and calculate the mean and standard deviation of the outlier detection rate for every error level.

We also introduce errors to the energy features. Unlike the MU features, the possible values of an energy feature are discrete. They can only be one of 6, 10, 18 and 23 MV. Similar to computing the outlier detection rate for MU features, we first randomly select a patient from the test set and select a feature out of four energy features (E_{AP}, E_{PA}, E_{RL}, E_{LL}). We then randomly set the value of the energy feature to one of the three values that are different from the original value. The outlier detection procedure is performed to compute the outlier detection rate for each of the four energy features and for all features combined.

3. Results and discussion

We check our clustering results by visualizing our data. One way to visualize data in dimensions greater than three is to project it to two dimensions, and plot the data in that two-dimensional space. We apply PCA described in section 2.2 to reduce the dimensionality. To be able to visualize the clustering results, we projected the data set onto its first three principal components, as shown in figures 2(a)–(c). By looking at all the three figures, the separation between the clusters becomes quite clear. For example, from figure 2(a), clusters 4 and 7 projected onto principal components 1 and 2 seem to be very close. However, it does not mean that they are actually one cluster. When we look from figures 2(b) and (c), where the data are projected onto principal components 1 and 3 and principal components 2 and 3, respectively, clusters 4 and 7 are clearly separated. That means they are two separated clusters. Similar situations exist for clusters 5 and 8, and clusters 1 and 6. Figure 2 also shows that indeed our clustering results (the different colors represent different clusters) make sense and the cluster means (marked in black dots) are correct.

To find the optimal number of clusters, we ran *K*-means clustering on the training set for values of k between 2 and 25, scoring each result using the Bayesian information criterion. We found $k = 8$ to produce the best final clustering results. This means, for this group of prostate cancer patients treated using the 'four-field box' technique, the beam energies and monitor units belong to eight distinct clusters. Again, figure 2 reveals that indeed there are eight clusters.

The mean and standard deviation of each feature of each cluster are given in table 1. It can be seen that all the energy features essentially have zero standard deviation within each

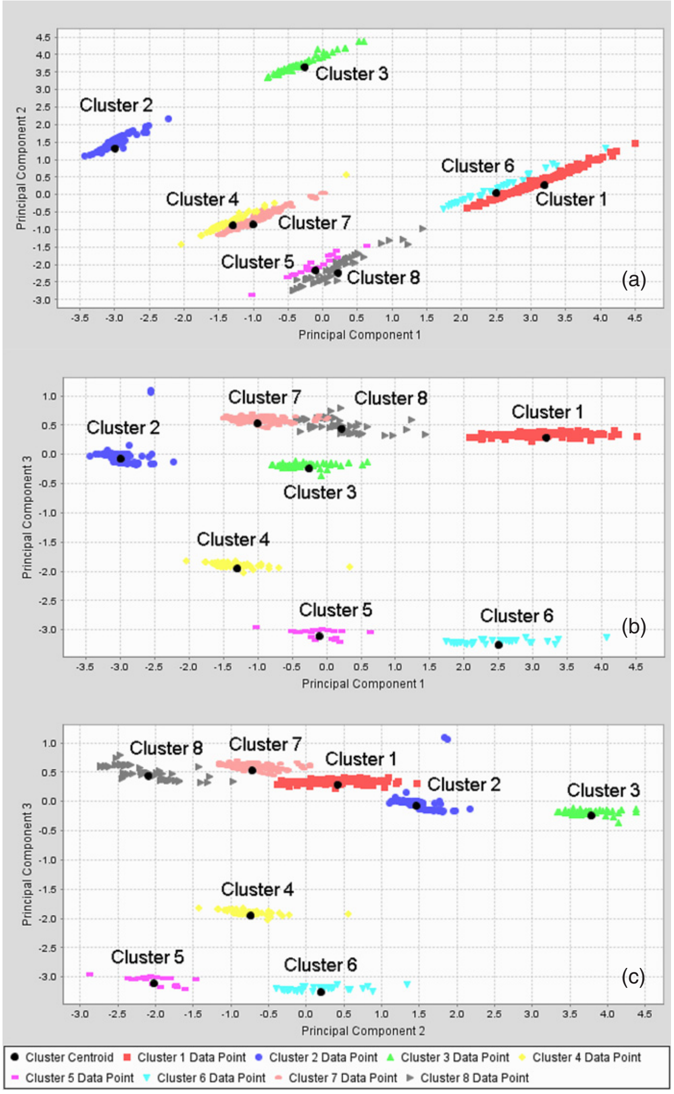

Figure 2. The training data projected onto two-dimensional planes formed by two of its first three principal components: (a) principal components 1 and 2; (b) principal components 1 and 3; (c) principal components 2 and 3.

cluster, which means the clusters are primarily separated based on the beam energies. For clusters 1 and 8, 2 and 3, 5 and 6, which have the same beam energies, they are then separated by two different levels of monitor units. The lower MU level corresponds to primary fields while the higher level corresponds to boost fields.

One noteworthy observation in this table is that the arithmetic mean of E_{AP} in cluster two is 9.9 MV (rather than 10 MV). This is because in cluster two, there is one point whose E_{AP} value is not 10 MV. On the other hand, the rest of its features are similar to those of cluster two, and so it was assigned to that cluster. It can be argued that this point might be an outlier when compared to the other points in the cluster.

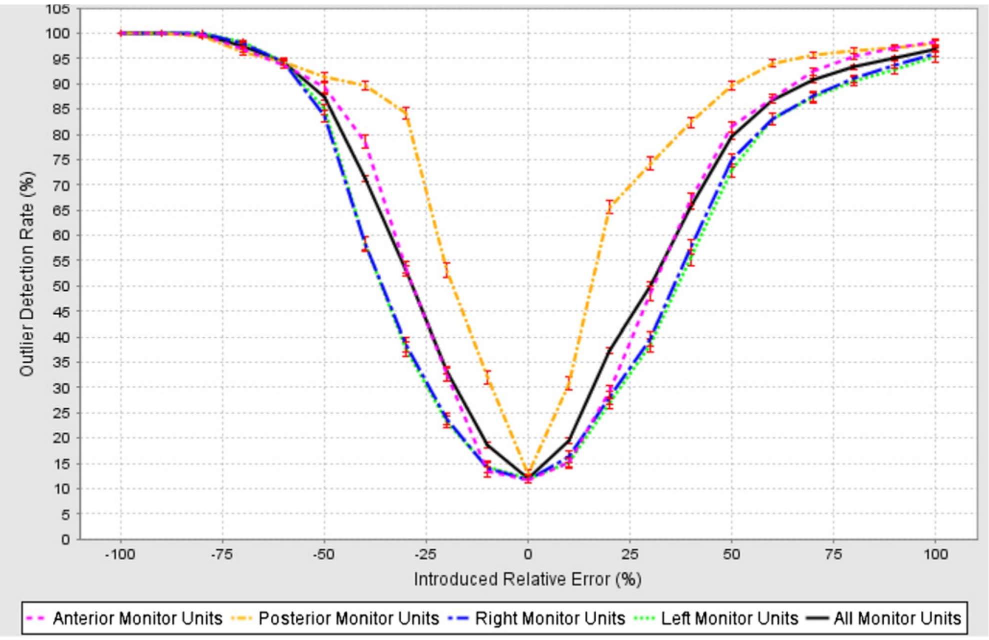

Figure 3. Outlier detection rate as a function of error level for each of the MU features and all MU features combined. The error bar shows one standard deviation.

Table 1. The mean and standard deviation (in parentheses) for each feature of each cluster.

	Features							
Clusters	E_{AP} (MV)	E_{PA} (MV)	E_{RL} (MV)	E_{LL} (MV)	$\mathrm{MU}_{\mathrm{AP}}$	$\mathrm{MU}_{\mathrm{PA}}$	$\mathrm{MU}_{\mathrm{RL}}$	$\mathrm{MU}_{\mathrm{LL}}$
1	6 (0)	23 (0)	23 (0)	23 (0)	111.5 (10.6)	103.9 (4.3)	183.0 (21.2)	182.1 (21.6)
2	9.9 (0.7)	10 (0)	10 (0)	10 (0)	57.8 (9.8)	59.2 (9.7)	62.5 (13.6)	62.8 (13.2)
3	10 (0)	10 (0)	10 (0)	10 (0)	108.5 (7.9)	109.2 (6.3)	154.1 (19.4)	151.0 (23.7)
4	18 (0)	18 (0)	18 (0)	18 (0)	62.2 (12.2)	56.3 (6.2)	58.9 (13.1)	59.0 (14.4)
5	23 (0)	23 (0)	23 (0)	23 (0)	59.3 (14.2)	57.6 (10.9)	68.2 (16.9)	67.6 (16.9)
6	23 (0)	23 (0)	23 (0)	23 (0)	108.5 (12.3)	105.2 (4.2)	152.7 (30.7)	152.1 (29.3)
7	6 (0)	18 (0)	18 (0)	18 (0)	66.5 (11.9)	54.5 (5.7)	65.3 (13.8)	64.8 (13.6)
8	6 (0)	23 (0)	23 (0)	23 (0)	56.2 (18.6)	51.4 (17.7)	83.5 (26.9)	83.0 (26.4)

The testing results are shown in figure 3. A few observations can be made from figure 3: (1) even at the 0% error level, the detection rate is still about 10%. This is basically the false positive rate; (2) the outlier detection rate for the anterior field is about the same as that for all fields together. The detection rates for the right and left fields are very similar. The performance of the proposed algorithm decreases from the posterior field, to the anterior field, and to the right and left fields. This is likely due to the fact that the distribution of the monitor units for the posterior field is narrower around peaks than other fields, as shown in figure 4. Also, in all of the clusters, the standard deviation of the posterior MUs is smaller than that of the other MU features, as can be seen from table 1; (3) the outlier detection rate as a function of the error level is asymmetric, with higher values for negative errors than positive errors. To understand the better performance with negative errors, consider

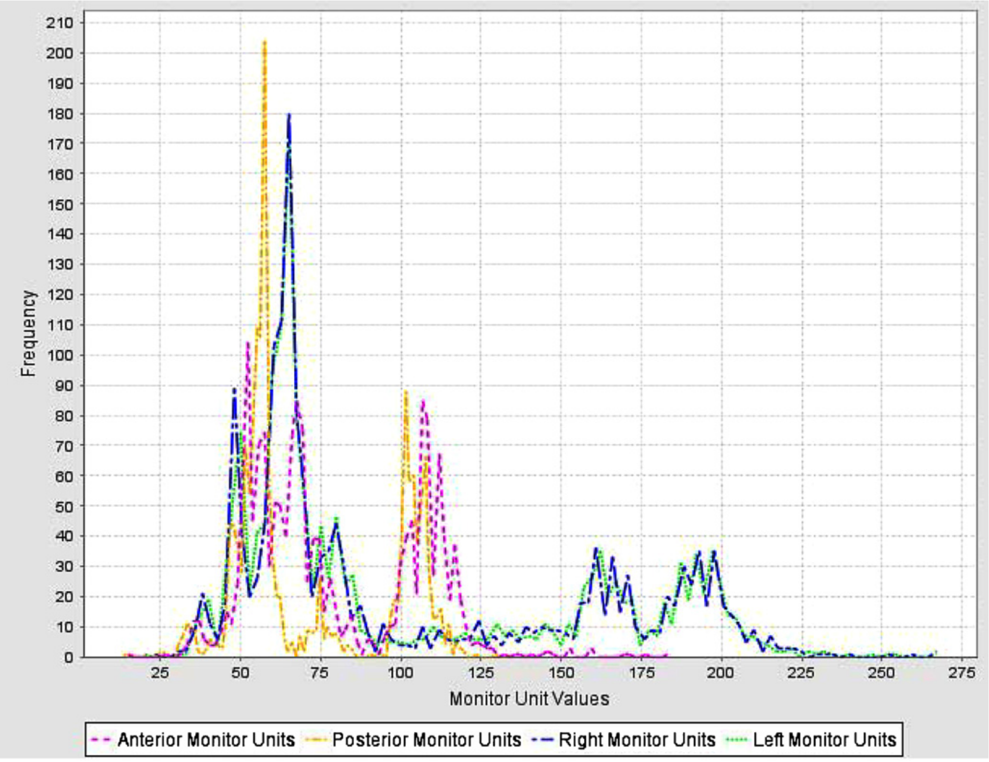

Figure 4. Histogram of monitor units for all four fields.

cluster 5 as a simple example. Referring to table 1, adding +100% errors to MUs in cluster 5 (and in essence, doubling their value) will cause the data points to appear to belong to cluster 6 and so they will not be detected as outliers. On the other hand, adding −100% errors to MUs in clusters 5 (making their values equal to 0) will cause the points to be detected as outliers. As for cluster 6, adding either +100% or −100% errors will cause the points to be detected as outliers. Therefore, there is an overall higher outlier detection rate with negative errors.

As shown in figure 3, the outlier detection rate changes with the error level as one might expect. At ±100% error level, the detection rate is about 100%. At ±50% error level, the detection rate is about 80%. It seems that the proposed algorithm has a good chance to detect errors at the levels of the Glasgow and Panama incidents (Akashi *et al* 2001, Johnston 2006). However, we should keep in mind that this preliminary study only utilized a very simple patient model, i.e., 'four-field box' prostate treatment.

For energy features, the outlier detection rate is 100% for E_{PA}, E_{RL} and E_{LL}, while for E_{AP} it is only 76.9%, resulting in an overall outlier detection rate of 94.2% for all four energy features combined. This can be understood by looking at table 1. For the same cluster, E_{PA}, E_{RL} and E_{LL} always have the same value and therefore changing one will cause the data point to be detected as an outlier, while for E_{AP}, this is not true.

The performance of the computer-clustering-based outlier detection method depends on how clustered the data points are. In the future, the performance of the proposed algorithm will be tested under more realistic situations where patients with various tumor sites and treated using various techniques are considered. For most of the treatment techniques, especially for intensity-modulated radiation therapy (IMRT), the treatment parameters are more scattered than those of the 'four-field box' technique. It is expected that a lot more effects will be needed before any error checker software is deployed in clinical practice. In this paper, we focused on a simplified treatment model in order to lay the foundation for an outlier detection method which can be built upon and extended for more complex cases. The results are encouraging and merit further research in developing our general outlier detection method to suit the needs of other treatment models.

We also plan to test more sophisticated data mining/machine learning algorithms for the proposed method, such as the local correlation integral (LOCI) for outlier detection (Papadimitriou *et al* 2003). In doing so, we hope to reduce the false positive rate of our current outlier detection method, as well as increase the outlier detection rate for large errors. An advantage of the LOCI algorithm is that it does not consider being an outlier as a binary property, meaning a point either is an outlier or is not. Rather, it offers a measure of the degree of 'outlierness' of the point. In addition, it provides an automatic, data-dictated cut-off point to determine whether or not a point should be classified as an outlier.

Future work will also utilize domain knowledge to help guide the data mining process and produce more accurate results (Stuhlinger *et al* 2000). An example of domain knowledge here is the correlation that exists between the parameters in a radiotherapy treatment plan. The energy features (particularly E_{PA}, E_{RL} and E_{LL}) have a great amount of redundancy. There is strong correlation between $\mathrm{MU}_{\mathrm{AP}}$ and $\mathrm{MU}_{\mathrm{PA}}$. This correlation is even greater for $\mathrm{MU}_{\mathrm{RL}}$ and $\mathrm{MU}_{\mathrm{LL}}$. In a recent preliminary work, we exploit this correlation by modifying the outlier detection program to include the ratio of $\mathrm{MU}_{\mathrm{AP}}$ to $\mathrm{MU}_{\mathrm{PA}}$ and $\mathrm{MU}_{\mathrm{RL}}$to $\mathrm{MU}_{\mathrm{LL}}$ in the search for outliers. The outlier detection results have been improved, especially for smaller errors (at ±25% error, the average outlier detection rate has gone from 45% to 60%, an improvement of 33%). We plan to conduct extensive investigation along this direction.

The treatment parameters included in the error checker will certainly not be limited to beam energy and monitor unit; instead, they should include field size, gantry angle, wedge angle and orientation, etc. The proposed method can also guide extra attention to unusual or new treatment techniques, where data points are sparse in the database and treatment parameters may be highlighted as outliers although they are not real errors.

4. Summary

In this paper, we proposed to use machine-learning and data-mining techniques to help detect large human errors in a radiotherapy treatment plan. Computer clustering has been applied in this work. The basic idea is to first cluster (based on the treatment parameters) a large number of treatment plans for patients who have been treated. Then, when checking a new treatment plan, the parameters of the plan will be tested to see whether or not they belong to the established clusters. If not, they will be considered as 'outliers' and therefore highlighted to catch the attention of the human experts.

As a preliminary study, we applied the *K*-means clustering algorithm to a simple patient model, i.e., 'four-field box' prostate treatment. One thousand data points were used to build the clusters while 650 data points were used to test the proposed method. It was found that there are eight clusters that describe the data well.

At ±100% error level introduced to the MU features, the outlier detection rate is about 100%. At ±50% error level, the detection rate is about 80%. The false positive rate is about 10%. For energy features, the outlier detection rate is 100% for the posterior, right-lateral and left-lateral fields, and 77% for the anterior field. This preliminary work has shown promise for developing the proposed automatic outlier detection software, although more effort will be required before a chart checker can be deployed for clinical use.

Acknowledgments

We would like to thank Greg Sharp, Javed Aslam, Evangelos Kanoulas and Ying Cui for their help on this project.

References

Akashi M, Cosset J-M, Gourmelon P, Konchalovsky M V, Mettler F A Jr, Ortiz L P and Vatnitsky S 2001 *Investigation of an Accidental Exposure of Radiotherapy Patients in Panama* (Vienna: International Atomic Energy Agency)

Calandrino R, Cattaneo G M, Del Vecchio A, Fiorino C, Longobardi B and Signorotto P 1993 Human errors in the calculation of monitor units in clinical radiotherapy practice *Radiother. Oncol.* **28** 86–8

Calandrino R, Cattaneo G M, Fiorino C, Longobardi B, Mangili P and Signorotto P 1997 Detection of systematic errors in external radiotherapy before treatment delivery *Radiother. Oncol.* **45** 271–4

Dunscombe P B, Iftody S, Ploquin N, Ekaette E U and Lee R C 2007 The equivalent uniform dose as a severity metric for radiation treatment incidents *Radiother. Oncol.* **84** 64–6

Ekaette E U, Lee R C, Cooke D L, Kelly K L and Dunscombe P B 2006 Risk analysis in radiation treatment: application of a new taxonomic structure *Radiother. Oncol.* **80** 282–7

Essers M and Mijnheer B J 1999 *In vivo* dosimetry during external photon beam radiotherapy *Int. J. Radiat. Oncol. Biol. Phys.* **43** 245–59

Fayyad U M 1996 Data mining and knowledge discovery: making sense out of data *IEEE Expert: Intell. Syst. Appl.* **11** 20–5

Forgy E 1965 Cluster analysis of multivariate data: efficiency versus interpretability of classifications *Biometrics* **21** 768–80

Fraass B, Doppke K, Hunt M, Kutcher G, Starkschall G, Stern R and Van Dyke J 1998 American association of physicists in medicine radiation therapy committee task group 53: quality assurance for clinical radiotherapy treatment planning *Med. Phys.* **25** 1773–829

Greene D, Tsymbal A, Bolshakova N and Cunningham P 2004 Ensemble clustering in medical diagnostics *CBMS '04: Proc. 17th IEEE Symp. on Computer-Based Medical Systems (CBMS'04)*

IAEA 2007 Lessons learned from accidental exposures in radiotherapy *International Atomic Energy Agency Report 17* (Vienna, Austria: IAEA)

ICRP 1973 Reports of Committee 3 on data for protection against ionising radiation from external sources *ICRP Publication 21* (Oxford: Pergamon)

ICRP 1991 *Recommendations of the International Commission on Radiological Protection* (Oxford: Pergamon)

ICRU 1993 Quantities and units in radiation protection dosimetry *ICRU Report 51*

Jain A K, Murty M N and Flynn P J 1999 Data clustering: a review *ACM Comput. Surv.* **31** 264–323

Johnston A M 2006 Unintended overexposure of patient Lisa Norris during radiotherapy treatment at the Beatson Oncology Centre, Glasgow in January 2006, Report of an investigation by the Inspector appointed by the Scottish Ministers for The Ionising Radiation (Medical Exposures) Regulations 2000 http://www.scotland.gov.uk/Publications/2006/10/27084909/0

Jolliffe I T 2002 *Principal Component Analysis* (New York: Springer)

Kutcher G J *et al* 1994 Comprehensive QA for radiation oncology: report of AAPM radiation therapy committee task group 40 *Med. Phys.* **21** 581–618

Leunens G, Verstraete J, Van den Bogaert W, Van Dam J, Dutreix A and van der Schueren E 1992 Human errors in data transfer during the preparation and delivery of radiation treatment affecting the final result: 'garbage in, garbage out' *Radiother. Oncol.* **23** 217–22

MacQueen J B 1967 Some methods for classification and analysis of multivariate observations *Proc. 5th Berkeley Symp. on Mathematical Statistics and Probability* vol 1 pp 281–97

Ostrom L T, Rathbun P, Cumberlin R, Horton J, Gastorf R and Leahy T J 1996 Lessons learned from investigations of therapy misadministration events *Int. J. Radiat. Oncol. Biol. Phys.* **34** 227–34

Papadimitriou S, Kitagawa H, Gibbons P B and Faloutsos C 2003 LOCI: fast outlier detection using the local correlation integral *Proc. 19th Int. Conf. on Data Engineering* pp 315–26

Pelleg D and Moore A W 2000 X-means: extending K-means with efficient estimation of the number of clusters *Proc. 17th Int. Conf. on Machine Learning* pp 727–34

Schwarz G 1978 Estimating the dimension of a model *Ann. Stat.* **6** 461–4

Stuhlinger W, Hogl O, Stoyan H and Muller M 2000 Intelligent data mining for medical quality management *Proc. 5th Workshop Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-2000), Workshop Notes of the 14th European Conf. Artificial Intelligence (ECAI-2000)* pp 55–67

Su T and Dy J 2004 A deterministic method for initializing K-means clustering *ICTAI '04: Proc. 16th IEEE Int. Conf. on Tools with Artificial Intelligence* pp 784–6

Yeung T K, Bortolotto K, Cosby S, Hoar M and Lederer E 2005 Quality assurance in radiotherapy: evaluation of errors and incidents recorded over a 10 year period *Radiother. Oncol.* **74** 283–91