Big Data Meets Medical Physics Dosimetery
Computer Integrated Surgery II
Seminar for Project IX:
Machine Learning Approach

Hilary Paisley

Team Members: Fumbeya Marungo, John Rhee
Mentors: Dr. Todd McNutt, Dr. Scott Robertson

1 Introduction

The goal of Project IX is to find a correlation between radiation treatment and specific sub-regions of the parotid glands. With Oncospace data, which includes dosages, voxel locations and toxicity grades, Project IX hopes to use machine-learning techniques to determine the relationship between locational treatment and toxicity.

Therefore, it is necessary to research the different approaches to machine-learning tasks and the appropriate data-mining algorithm. The first paper discusses a common approach to machine-learning tasks, explaining the nine-step KDD process. The second paper focuses on a specific data-mining algorithm that seems appropriate for Project IX's task.

2 KDD Process

2.1 Introduction

Project IX focuses on a machine learning approach to solving a Medical Physics Dosimetry problem. The knowledge discovery in databases (KDD) process is a key method for a successful machine learning algorithm and approach.

"The KDD process is one of mapping low-level data into other forms that might be more compact, more abstract or more useful" (Fayyad, et al). In this case, Project IX is most interested in transforming the data into useful information for medical doctors to assess new patients in their radiation treatment.

2.2 Steps in the Process

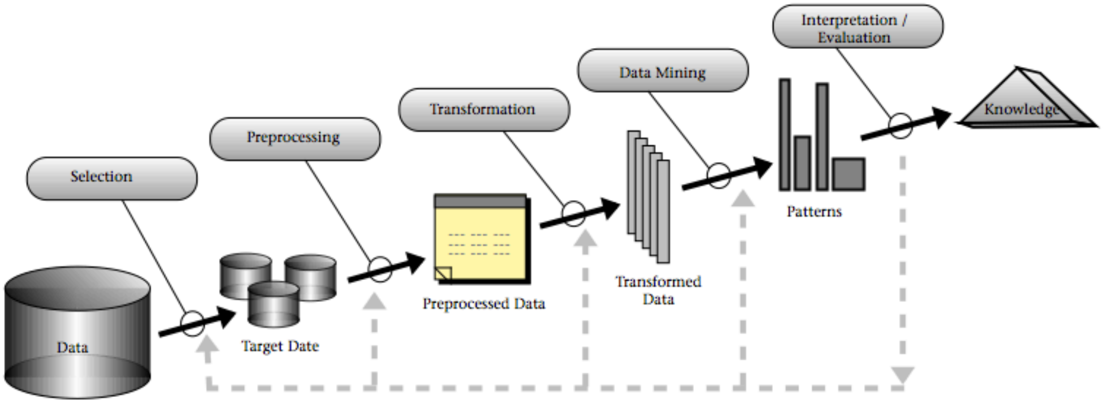

Figure 1: KDD Process (Fayyad, et. al)

1. Understand Application Domain – identifying the goal of the process and obtaining relevant background information to meet that goal
2. Create Target Data Set – selecting data set or determine a focus on specific variables which will be used for data mining techniques
3. Data Cleaning and Preprocessing – removing noise, accounting for missing data fields and finding strategies to handle such problems in the data
4. Data Reduction and Projection – finding useful features based on the goal of the process
5. Find a Data Mining Method – making sure to match the goals with a specific data mining method
6. Exploratory Analysis – assess whether the data mining method found will accurately assess the data for the goal established
7. Data Mining – implementing the algorithm
8. Interpretation – assess the results and if any of the previous steps need to be repeated
9. Implementation – using the knowledge discovered or integrating the data into different areas of interest

2.3 Application to Project IX

In order to obtain success with Project IX, it is imperative that the KDD process or a similar process is followed. The data-mining step may seem like the most important, but if a correct algorithm cannot be found or the data is not in a ready form, there is no way for the data-mining step to be successful.

3 Random Forests

3.1 Introduction

"A random forest is a classifier consisting of a collection of tree-structures classifiers $\{h(\boldsymbol{x}, \Theta_k), k = 1, \dots\}$ where the $\{\Theta_k\}$ are independent identically distributed random vectors and each tree casts a unit for the most popular class at input $\mathbf{x}$" (Breiman, L). The algorithm will construct a multitude of different decision trees when training. Decision trees are predictive models that make observations about an instance to make a conclusion about the target value. Random forests are an extension on tree bagging.

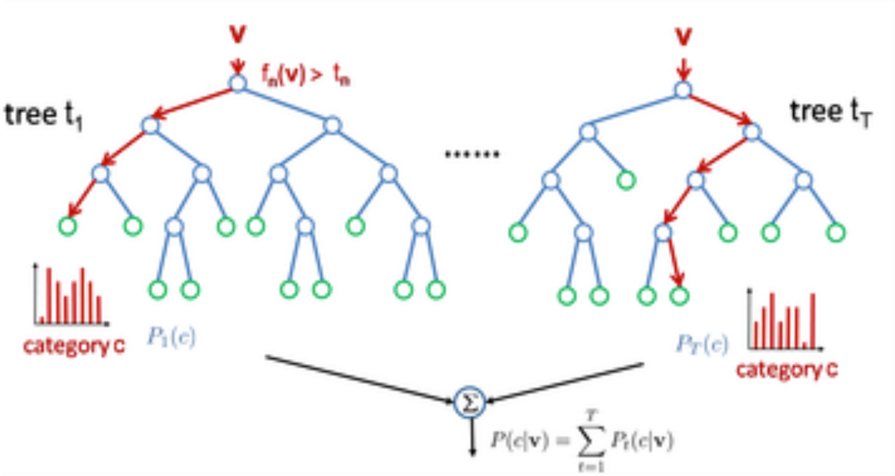

Figure 2: Random Forest
(http://www.iis.ee.ic.ac.uk)

3.2 Tree Bagging

Given a training set, X, and output values, Y, tree bagging will continually select a bootstrap sample of the training set and fit trees to these samples. For each sample, get n training examples and train a decision tree, f, on the samples. After training, predict for unlabeled examples by averaging the predictions from the individual decision trees (take the majority vote of the decision trees).

The above paragraph describes the original tree bagging method. Random forests instead when training select a random subset of features instead of picking strong features. This leads to less correlation, which is better for testing, leading to less bias.

3.3 Convergence

Theorem: As the number of trees increases, for almost surely all sequences $\Theta_1, \ldots PE^*$ converges to $P_{\boldsymbol{X},Y}(P_\Theta(h(\mathbf{X}, \Theta) = Y) - \max P_\Theta\ (h(\mathbf{X}, \Theta) = j) < 0)$ where the generalization error, $PE^* = P_{\boldsymbol{X},Y}(mg(\boldsymbol{X}, Y) < 0)$ (Breiman, L).

Proof: Consider a fixed training set and a fixed Θ. Let, for a set of $\mathbf{x}$, $h(\Theta, \boldsymbol{x}) = j$ be a union of hyper-rectangles, which are generalizations of rectangles for higher dimensions. There exist only a finite number of unions between these hyper-rectangles, denoted $S_1, \ldots, S_K$. Now define $\varphi(\Theta) = k$ if $\{\boldsymbol{x} : h(\Theta, \boldsymbol{x}) = j\} = S_k$ and let N_k be the number of times $\varphi(\Theta) = k$.

Then $\frac{1}{N}\sum_{n=1}^{N} I(h(\Theta_n, \boldsymbol{x}) = j) = \frac{1}{N}\sum_k N_k I(\boldsymbol{x} \in S_k)$.

By Law of Large Numbers, $\frac{1}{N}\sum_{n=1}^{N} N_k I(\varphi(\Theta_n) = k)$ converges.

So in conclusion, $\frac{1}{N}\sum_{n=1}^{N} I(h(\Theta_n, \boldsymbol{x}) = j) = \sum_k P_\Theta(h(\Theta, \boldsymbol{x}) = j)$.

Because there is convergence of the generalization error, random forests do not over-fit as more trees are added.

3.4 Correlation

The generalization error is dependent on the strength of the individual trees and the overall tree correlation. Less correlation leads to a lower generalization error. When choosing random split vectors, the correlation is much lower than in previous tree bagging applications.

An upper bound for the generalization error is calculated as $PE^* \leq \frac{\rho(1-s^2)}{s^2}$. The strength of the set of classifiers is calculated as $s = E_{\boldsymbol{X},Y} mr(\boldsymbol{X}, Y)$, where $mr(\boldsymbol{X}, Y) = P_\Theta(h(\boldsymbol{X}, \Theta) = Y) - \max P_\Theta(h(\boldsymbol{X}, \Theta) = j)$ and is called the margin function. The correlation term can be calculated from the equation $var(mr) = \rho(E_\Theta sd(\Theta))^2$, where sd stands for standard deviation.

3.5 Application to Medical Physics Dosimetry

Project IX has decided to try using a random forest mainly because of its computational efficacy. Using one decision tree would not be as time efficient as using the random vector splitting of a random forest method. Also, for further application after the conclusion of this course, it is easy to determine from the random forest which features are key to the organ's functioning.

4 Possible Pitfalls

4.1 Over-fitting

Over-fitting is when a training data set will have close to 100% accuracy and the testing data set will have much lower accuracy. This occurs when there is a limited set of data or when the training data set is not an appropriate representation of the data as a whole.

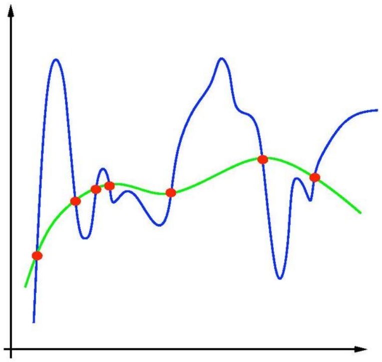

Figure 3: Over-fitting
(en.wikipedia.org)

A straightforward solution would be to obtain more data for training and testing, but this is very difficult to do, especially when dealing with medical data. An easier solution is cross-validation, which is running different data sets as training and testing data, or regularization, where additional information is obtained and the training and testing data is retested. Another solution would be to add more trees to the random forest algorithm because, as proved previously, random forests will not over-fit with more trees.

4.2 Constantly Changing Data

When data is changing or more data is constantly being added to the data set, the machine-learning algorithm needs to be able to account for such changes. With medical data, the patient's health is constantly changing and more patients are diagnosed and treated every day.

Therefore, the machine-learning algorithm must have an appropriate way to update based on the new information and shift the outcome if necessary.

4.3 Noisy Data

Noisy data includes data entries that may be outliers, have missing data fields or have incorrect data fields. Medical data is quite noisy and this needs to be accounted for when determining how to set up the machine-learning algorithm.

A solution would be to find the hidden variables and reassess the data to make sure it is accurate. This can be challenging and it may be necessary to account for possible noise in the data with an error term.

5 Conclusion

The papers described above will provide Project IX with a good starting point for this project. The first paper lays out a common approach to machine-learning tasks in general, which Project IX has used to establish the calendar and checkpoints. The second paper describes a possibly successful algorithm to handle the data in an accurate manner and potentially lead to interesting results.

- Fayyad, et. al. From Data Mining to Knowledge Discovery in Databases. American Association for Artificial Intelligence, 1996.
- Breiman, Leo. Random Forests. Machine Learning, 45, 5-32, 2001.