MEDINFO 2021: One World, One Health – Global Partnership for Digital Innovation
P. Otero et al. (Eds.)

doi:10.3233/SHTI220118

Feature Engineering for Interpretable Machine Learning for Quality Assurance in Radiation Oncology

Malvika Pillai[a], Karthik Adapa[a,b], John W Shumway[b], John Dooley[b], Shiva K Das[b], Bhishamjit S Chera[b], Lukasz Mazur[b]

[a]Carolina Health Informatics Program, University of North Carolina, Chapel Hill, North Carolina
[b]Department of Radiation Oncology, University of North Carolina, Chapel Hill, North Carolina

Abstract

Chart checking is a time intensive process with high cognitive workload for physicists. Previous studies have partially automated and standardized chart checking, but limited studies implement data-driven approaches to reduce cognitive workload for quality assurance processes. This study aims to evaluate feature selection methods to improve the interpretability and transparency of machine learning models in predicting the degree of difficulty for a pretreatment physics chart check. We compare chi-square, mutual information, feature importance thresholding, and greedy feature selection for four different classifiers. Random forest has the highest performance with SMOTE oversampling using mutual information for feature selection (accuracy 84.0%, AUC 87.0%, precision 80.0%, recall 80.0%). This study demonstrates that feature selection methods can improve model interpretability and transparency.

Keywords:
Machine learning, radiation oncology, quality assurance

Introduction

Prior to radiation therapy, quality assurance (QA) is conducted after treatment planning to ensure quality of treatment and patient safety. Treatment planning is typically done by a team of dosimetrists, physicists, and physicians, where QA is done by dosimetrists, physicists, and physicians. Studies show that the majority of errors that occur in the radiation therapy workflow happen in the pretreatment process [5; 9]. The pretreatment QA process is broken into a series of dosimetry checks and physics checks. After a dosimetry check, physicists verify the dosimetry check and perform their own chart review. The nationally regulated approach to QA consists of a physicist manually checking a series of metrics prior to radiation therapy [11]. This process is necessary to catch errors and ensure patient safety, and although physics QA checks are very effective, this can lead to physicists having a high cognitive workload, which can impact how sensitive they are to errors in treatment plans [5; 6]. Recent efforts have been applied in the QA process to reduce workload and optimize accuracy [4; 7; 8; 12]. Machine learning has also been implemented in this space to assist with automation and strengthening effectiveness of physics chart checks [10]. However, there are very few studies that implement data-driven approaches for process QA [1].

The aim of this study is to evaluate feature selection methods to improve the interpretability and transparency of machine learning models for physicists by predicting the degree of difficulty to check a plan. After prediction, the objective is for physicists to be able to differentiate between plans that require more or less scrutiny prior to starting the chart check process. While the study is not directly aimed at error prediction, reducing cognitive workload of physicists can improve their effectiveness and thereby have downstream effects on improving patient safety.

Methods

Data Collection

Data were retrieved from physics pre-treatment and weekly chart checks for treatment plans of each patient from July 2018 to October 2020, encompassing all cancer sites at UNC Department of Radiation Oncology. The outcome variable, the degree of difficulty of treatment plans, was collected from physics pre-treatment chart checks from sixteen physicists as a subjective rating on a scale of 1-10. 778 patient plans were used as a training set, divided into 622 patient plans to train and 156 patient plans to test (test set).

Attribute Selection

An iterative data selection process was conducted by one clinician, one physicist, and a software development team to select attributes along the radiation therapy workflow, which consists of four key stages: shared decision making, CT simulation, treatment planning, and quality assurance [10]. The attributes selected were based on clinical relevance, contribution to plan complexity, and quality assurance metrics and taken throughout the workflow. Clinically relevant features include patient age (mean 62 years, SD 15 years), patient sex (52.73% male, 47.27% female) and site name. All fourteen sites found in plans were included in the analysis. Plan complexity features included numbers of isocenters (mean 1, SD 1), fractions (mean 16, SD 11), beam sets (mean 2, SD 1), and images (mean 5, SD 6) as well as organ count (mean 33, SD 24), dose per monitor unit (MU) (mean 0.64, SD 0.33), having a pacemaker (99.9% without, 0.01% with), being pregnant (99.2% not pregnant, .8% pregnant), and having had previous treatment (86.0% without, 14.0% with). Quality assurance features included numbers of physicians (mean 1, SD 1) and dosimetrists (mean 1, SD 1) on a plan, whether the plan was on an accelerated schedule (72.3% not accelerated, 27.2% accelerated, 0.5% missing data), and the physicist on the plan. The physicist on the plan was incorporated to account for differences in experience levels and perceptions of degree of difficulty.

Data preparation

The outcome variable for the analysis was the degree of difficulty for each treatment plan. The degrees of difficulty were binned into two classes to enable a binary classification. The plans that were rated 1-5 were considered not difficult, and plans that were rated 6-10 were considered difficult. Synthetic Minority Over-sampling Technique (SMOTE)[2] with a minority sampling strategy was used to oversample the difficult class and improve performance (Table 1).

Table 1– Dataset Class Distributions

Train (n=622)	**Count (% of Set)**
Not Difficult	440
Difficult	182

Train (Oversampled) (n=740)	**Count (% of Set)**
Not Difficult	440
Difficult	300

Test (n=156)	**Count (% of Set)**
Not Difficult	108
Difficult	48

Feature Selection and Classification

Four different feature selection methods were compared for four different classifiers. The feature selection methods used include: mutual information, chi-square test (filtering methods), feature importance thresholding (wrapper method), and greedy forward selection (embedded method). Mutual information is the measurement of mutual dependence between two random variables, where the larger the measurement, the more dependent one variable is on another. Features with a mutual information score > 0.05 were included in the analysis. The mutual information score threshold was selected by training classifiers with 10-fold cross-validation and selecting the score related to the highest accuracy. Chi-square feature selection consists of testing the independence between variables, and we aim to select the features that are significantly dependent on the outcome variable. Features with p-value < 0.05 were included in the analysis. Feature importance thresholding is where scores are assigned to input features to estimate the relative importance of a feature when making a prediction and scores above a set threshold are selected for the model. Greedy forward selection is the process of iteratively adding a single feature to identify the feature set that maximizes performance. Both feature importance thresholding and greedy forward selection use 10-fold cross-validation to produce different feature sets for each classifier, where features are iteratively added in every fold and optimal feature sets are selected based on performance on each cross-validation subset. Mutual information and chi-square test are applied to the dataset, producing a single feature set each that is inputted into all classifiers.

Different classifiers were selected to compare performance. Random forest, support vector machine (SVM), an adaptive boosting classifier (adaboost), and logistic regression were used for prediction. As SVM does not have a feature importance parameter, feature importance thresholding was not conducted for SVM. All other feature selection methods were compared across the algorithms.

Results

The five feature selection methods were applied to predict degree of difficulty with four classifiers. Accuracy, area under the Receiver Operating Characteristic curve (AUC), precision and recall were used to evaluate performance on the test set with and without SMOTE oversampling. Table 2 shows accuracies of classifiers using and not using feature selction methods with the highest accuracy for each feature selection method in bold and the highest accuracy for each algorithm highlighted. Without oversampling, not using feature selection resulted in better accuracy for all algorithms. However, SVM and logistic regression showed equal accuracy without feature selection and using greedy feature selection. With oversampling, not using feature selection resulted in the highest accuracy for the random forest classifier. For SVM and adaboost, using greedy feature selection resulted in the highest accuracies. When comparing algorithm accuracies on the test set between using and not using oversampling, random forest and logistic regression had higher accuracies after oversampling. Random forest achieved the highest accuracy with oversampling and without feature selection (i.e., using all features) (0.84). Across all feature selection methods, random forest also consistently outperformed all other classifiers in terms of accuracy. The highest accuracy for logistic regression was with oversampling and using chi-square for feature selection (0.77). The highest accuracies for SVM remained the same before and after oversampling. Without oversampling, SVM achieved 0.71 accuracy without feature selection, and with oversampling, SVM achieved 0.71 with greedy feature selection. Adaboost accuracy was highest without oversampling and without feature selection (0.76). Thus, in all classifiers, using all features resulted in highest accuracies.

Table 2– Accuracy for Classifiers with and without Feature Selection Methods on Test Set

Without SMOTE Oversampling					
Algorithm	**All features used**	**Chi-Square**	**Mutual Information**	**Feature Importance Thresholding**	**Greedy Feature Selection**
Random Forest	**0.83**	**0.81**	**0.80**	**0.80**	**0.79**
SVM	0.71	0.71	0.71	N/A	0.71
Adaboost	0.76	0.75	0.73	0.72	0.73
Logistic Regression	0.75	0.74	0.73	0.70	0.75
With SMOTE Oversampling					
Algorithm	**All features used**	**Chi-Square**	**Mutual Information**	**Feature Importance Thresholding**	**Greedy Feature Selection**
Random Forest	**0.84**	**0.78**	**0.80**	**0.82**	**0.78**
SVM	0.70	0.70	0.70	N/A	0.71

Adaboost	0.72	0.74	0.72	0.72	0.75
Logistic Regression	0.76	0.77	0.73	0.72	0.74

After evaluating the algorithms on accuracy, ROC curves were created to evaluate all classifiers without feature selection and with greedy feature selection before and after oversampling. Greedy feature selection and not using feature selection were selected because they had the highest performance. Random forest had the highest AUC out of all classifiers without oversampling and using all features (0.87) as well as with oversampling, using all features (0.87) and using mutual information (0.87). SVM had the highest AUC with oversampling and greedy feature selection (0.79). Adaboost had the highest AUC without oversampling using chi-square feature selection (0.78) and feature importance thresholding (0.78) as well as with oversampling and greedy feature selection (0.78). Logistic regression had the highest AUC with oversampling and mutual information (0.77) (Table 2).

Algorithm performance was analyzed with precision and recall due to the class imbalance (Table 2). Without oversampling, random forest had the highest precisions (0.87) and recalls (0.79) compared to all other classifiers. With oversampling, random forest also had the highest precisions and recalls. For precision, random forest performed equally with mutual information and with feature importance thresholding (0.80). For recall, random forest performed best with mutual information (0.80).

Without oversampling, using random forest achieved its best performance in terms of precision and recall after using all features. SVM achieved equal recalls (0.58) after using all features, chi-square, and mutual information feature selection. It achieved highest precision using mutual information (0.65). Adaboost achieved the highest precision and recall after using all features, like random forest. Logistic regression using all features and using greedy feature selection had a recall of 0.65. After greedy feature selection, it also achieved its highest precision (0.72).

With oversampling, algorithm performance in terms of precision and recall varied more. Random forest performed the best across all feature selection methods except for chi-square feature selection. Using chi-square, logistic regression had the highest precision (0.73) and recall (0.73). Greedy feature selection allowed SVM and adaboost to perform their best with SVM having 0.68 precision and 0.71 recall and adaboost having 0.71 precision and 0.71 recall. SVM also achieved 0.68 precision using mutual information.

When compared with accuracy, using AUC, precision and recall metrics provides a more detailed evaluation of algorithm performance across feature selection methods. While using all features produced the best accuracies, it did not produce the best AUC, precision, and recall after oversampling. Notably, the highest recall overall was from random forest using mutual information (0.80). The highest precision overall was from random forest using all features (0.81).

Table 2– ROC AUC, precision (P), and recall (R) values with and without SMOTE oversampling

Without SMOTE Oversampling

Algorithm	**Using all features**			**Chi-Square**			**Mutual Information**			**Feature Importance Thresholding**			**Greedy Feature Selection**		
	AUC	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**
Random Forest	**0.87**	**0.81**	**0.79**	**0.81**	**0.73**	**0.68**	**0.86**	**0.79**	**0.72**	**0.86**	**0.79**	**0.76**	**0.84**	**0.76**	**0.74**
SVM	0.73	0.64	0.58	0.72	0.64	0.58	0.69	0.65	0.58	N/A			0.74	0.64	0.57
Adaboost	0.77	0.72	0.68	0.78	0.71	0.67	0.77	0.68	0.65	0.78	0.66	0.63	0.77	0.68	0.66
Logistic Regression	0.74	0.71	0.65	0.74	0.70	0.64	0.75	0.68	0.63	0.67	0.62	0.57	0.76	0.72	0.65

With SMOTE Oversampling

Algorithm	**Using all features**			**Chi-Square**			**Mutual Information**			**Feature Importance Thresholding**			**Greedy Feature Selection**		
	AUC	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**	**AUC**	**P**	**R**
Random Forest	**0.87**	**0.79**	**0.77**	0.72	0.70	0.69	**0.87**	**0.80**	**0.80**	**0.85**	**0.80**	**0.78**	**0.84**	**0.75**	**0.73**
SVM	0.78	0.66	0.68	**0.78**	0.66	0.68	0.78	0.68	0.70	N/A			0.79	0.68	0.71
Adaboost	0.76	0.67	0.66	0.77	0.70	0.69	0.76	0.68	0.68	0.76	0.67	0.67	0.78	0.71	0.71
Logistic Regression	0.75	0.71	0.71	0.75	**0.73**	**0.73**	0.77	0.69	0.69	0.67	0.66	0.65	0.76	0.69	0.70

Discussion

Machine learning methods have increasingly been used in radiation oncology to improve effectiveness of radiation treatment planning QA. While efforts to automate the QA check process are ongoing, manual checks are still the primary method for chart review. The average physics QA check has approximately 170 items and can take multiple hours to complete [3]. The overall goal for this work is to create a data-driven aid to help physicists understand what kind of plan they are going to work on so they can budget their time more effectively and be more vigilant with plans that are deemed difficult. Currently, physicists depend on prior experience to assess the difficulty of a plan and approximate how long a chart check will take. The task for the current study was to predict degree of difficulty of pre-treatment chart checks, but more specifically, it was to survey feature selection methods to ensure that the final models are parsimonious, interpretable and transparent. Any future aid created for physicists will need to fit into their current workflow, and for a machine learning tool to be useful to them, it must provide reasoning behind predictions. By using feature selection methods, we can identify the most important features for prediction and thereby demonstrate why a plan was classified as difficult or not difficult.

This study analyzed performance using four different metrics in a classification with an imbalanced dataset. Oversampling was used to reduce the class imbalance in the dataset and improve performance. Results showed that on the test set, oversampling did not increase accuracy for all algorithms, and for algorithms with higher accuracies after oversampling, the difference was marginal. However, when comparing with AUC, precision and recall, using oversampling resulted in higher performance in many cases. For an imbalanced classification, precision and recall can be more useful to evaluate performance because accuracy values can be skewed by the majority class, as shown in this study. The limitation of oversampling is that it can increase the likelihood of overfitting and reduce generalizability on the test set, which was likely the case in this study. However, some algorithms could suffer more when faced with class imbalance, which could be why SVM performance is much higher with oversampling compared to without. Classifiers performed the best without using feature selection and using greedy feature selection.

A similar study was conducted by Brown et al.[1], where an undersampling framework was used to support imbalanced data classification with SVM variants. The study used AUC and false positive rate at two thresholds to evaluate performance. However, they did not evaluate their approach on a test set and only provided metrics for cross-validation performance. We used cross-validation for hyperparameter selection but reported metrics on the validation set in this work.

A key limitation of this work was the limited dataset size. Also, all data were collected in a single academic institution, and the outcome variable, degree of difficulty, was a subjective rating by physicists with different experience levels. The attributes selected by the team were double the number included in the study due to data accessibility issues. Therefore, adding more attributes that could contribute to plan check difficulty would make the analysis more robust.

For future work, we plan to optimize prediction of more difficult cases and better the imbalanced data classification by using a voting schema to classify cases based on agreement across multiple algorithms. We also intend to implement the machine learning algorithms in the clinic, allowing physicists to rank treatment plans based on the predicted degree of difficulty.

Conclusions

The findings from this study demonstrate that feature selection methods improve the transparency and interpretability methods of ML algorithms. With respect to interpretability, being able to identify the most important features will improve physicist adoption of these classifiers in a future clinical implementation. When physicists make decisions on how to budget their time, seeing the factors contributing to a classification can be less disruptive and potentially reduce cognitive workload and errors, which will in turn improve patient safety.

Acknowledgements

The first author is funded by the NLM T15 training grant #T15-LM012500.

References

[1] W.E. Brown, K. Sung, D.M. Aleman, E. Moreno-Centeno, T.G. Purdie, and C.J. McIntosh, Guided undersampling classification for automated radiation therapy quality assurance of prostate cancer treatment, *Medical Physics* **45** (2018), 1306-1316.

[2] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer, SMOTE: synthetic minority oversampling technique, *Journal of artificial intelligence research* **16** (2002), 321-357.

[3] E.L. Clouser, Q. Chen, and Y. Rong, Computer automation for physics chart check should be adopted in clinic to replace manual chart checking for radiotherapy, *Journal of applied clinical medical physics* **22** (2021), 4.

[4] E.L. Covington, X. Chen, K.C. Younge, C. Lee, M.M. Matuszak, M.L. Kessler, W. Keranen, E. Acosta, A.M. Dougherty, S.E. Filpansick, and J.M. Moran, Improving treatment plan evaluation with automation, *Journal of applied clinical medical physics / American College of Medical Physics* **17** (2016), 16-31.

[5] E. Ford, L. Conroy, L. Dong, L.F. de Los Santos, A. Greener, G. Gwe-Ya Kim, J. Johnson, P. Johnson, J.G. Mechalakos, and B. Napolitano, Strategies for effective physics plan and chart review in radiation therapy: report of AAPM Task Group 275, *Medical Physics* **47** (2020), e236-e272.

[6] E.C. Ford, S. Terezakis, A. Souranis, K. Harris, H. Gay, and S. Mutic, Quality control quantification (QCQ): a tool to measure the value of quality control checks in radiation oncology, *International Journal of Radiation Oncology, Biology, Physics* **84** (2012), e263-269.

[7] C. Holdsworth, J. Kukluk, C. Molodowitch, M. Czerminska, C. Hancox, R.A. Cormack, K. Beaudette, and J.H. Killoran, Computerized system for safety verification of external beam radiation therapy planning, *International Journal of Radiation Oncology, Biology, Physics* **98** (2017), 691-698.

[8] D. Lack, J. Liang, L. Benedetti, C. Knill, and D. Yan, Early detection of potential errors during patient treatment planning, *Journal of applied clinical medical physics / American College of Medical Physics* **19** (2018), 724-732.

[9] A. Novak, M.J. Nyflot, R.P. Ermoian, L.E. Jordan, P.A. Sponseller, G.M. Kane, E.C. Ford, and J. Zeng, Targeting safety improvements through identification of incident origination and detection in a near-miss incident learning system, *Medical Physics* **43** (2016), 2053-2062.

[10] M. Pillai, K. Adapa, S.K. Das, L. Mazur, J. Dooley, L.B. Marks, R.F. Thompson, and B.S. Chera, Using artificial intelligence to improve the quality and safety of radiation therapy, *Journal of the American College of Radiology* **16** (2019), 1267-1272.

[11] G.S. Tracton, L.M. Mazur, P. Mosaly, L.B. Marks, and S. Das, Developing and assessing electronic checklists for safety mindfulness, workload, and performance, *Practical radiation oncology* **8** (2018), 458-467.

[12] K.C. Younge, K.W. Naheedy, J. Wilkinson, J. Dekmak, E. Covington, B. Durbin, E. Nelson, S. Filpansick, and J.M. Moran, Improving patient safety and workflow efficiency with standardized pretreatment radiation therapist chart reviews, *Practical radiation oncology* **7** (2017), 339-345.

Address for correspondence

Lukasz Mazur

Email: lukasz_mazur@med.unc.edu