Novel Methods of Biomarker Discovery and Predictive Modeling using Random Forest

by

Xin Guan

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved June 2017 by the
Graduate Supervisory Committee:

George Runger, Co-Chair
Li Liu, Co-Chair
Valentin Dinu

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

Random forest (RF) is a popular and powerful technique nowadays. It can be used for classification, regression and unsupervised clustering. In its original form introduced by Leo Breiman, RF is used as a predictive model to generate predictions for new observations. Recent researches have proposed several methods based on RF for feature selection and for generating prediction intervals. However, they are limited in their applicability and accuracy. In this dissertation, RF is applied to build a predictive model for a complex dataset, and used as the basis for two novel methods for biomarker discovery and generating prediction interval.

Firstly, a biodosimetry is developed using RF to determine absorbed radiation dose from gene expression measured from blood samples of potentially exposed individuals. To improve the prediction accuracy of the biodosimetry, day-specific models were built to deal with day interaction effect and a technique of nested modeling was proposed. The nested models can fit this complex data of large variability and non-linear relationships.

Secondly, a panel of biomarkers was selected using a data-driven feature selection method as well as handpick, considering prior knowledge and other constraints. To incorporate domain knowledge, a method called Know-GRRF was developed based on guided regularized RF. This method can incorporate domain knowledge as a penalized term to regulate selection of candidate features in RF. It adds more flexibility to data-driven feature selection and can improve the interpretability of models. Know-GRRF showed significant improvement in cross-species prediction when cross-species correlation was used to guide selection of biomarkers. The method can also compete with existing methods using intrinsic data characteristics as alternative of domain knowledge in simulated datasets.

Lastly, a novel non-parametric method, RFerr, was developed to generate prediction interval using RF regression. This method is widely applicable to any predictive models and was shown to have better coverage and precision than existing methods on the real-world radiation dataset, as well as benchmark and simulated datasets.

ACKNOWLEDGMENTS

It is an unforgettable experience to be a PhD student at Arizona State University. I still remember the first day when I walked into my first class. And now here I am, putting the finishing touches on my thesis! My life has changed significantly in past years, but while all of those changes were occurring, I managed to complete some very interesting research. This dissertation would certainly not have come to its successful conclusion without the help, support and trust of colleagues, friends and family.

First and foremost, I would like to sincerely thank my previous advisor Dr. Garrick Wallstrom for his help, guidance and support throughout these years. He offered me great research opportunities, interesting projects, resources, funding, and trust which allowed me to fully explore the research area.

I am grateful to my committee co-chairs Dr. George Runger and Dr. Li Liu, committee member Dr. Valentin Dinu, for their valuable comments and suggestions for both my dissertation and future career. Dr. Runger taught me about data mining and opened the door to my research area. Dr. Liu gave me insightful ideas in continuing my research. Dr. Dinu was the most supportive professor and helped with my manuscript writing. Thank you all for your interest in my work and taking the time to evaluate this dissertation.

I want to take this opportunity to thank Dr. Josh LaBaer’s team members who work on Biodosimeter Development. Without their wonderful work, I cannot finish my dissertation. I would like to extend my gratitude to Paul, Jin, Vel, Mitch, Kris and so on.

I am also grateful to the funding agency, biomedical advanced research and development authority for the research funding. And thanks for ASU Graduate College, I was funded by the dissertation fellowship so that I can complete my work without worrying.

I would like to thank all the people who have helped me at ASU, Maria, Lauren, Patricia and all the faculty members. Special thank goes to my graduate student colleagues. Thanks for the collaboration and friendship!

Last but not the least, I want to thank my parents, Yan and Qingjun, parents-in-law, Xiaohui and Zhi, for their tremendous love and support. Thank my husband, Minglu. I am forever grateful for your help, understanding, love and support. Thank my son, Kevin, for your cutest smile.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

Table Page

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

1.1 Predictive Modeling

In the world of information exposure, machine learning has become a critical part of the scientific methodology, allowing for revealing underlying pattern in data and predictions of a phenomenon based on history observations. In biomedical informatics domain, recent advances in technologies of microarray and next-generation sequencing have made it possible to access to our gigantic gene profiles. An important goal of most gene expression studies is to understand the relationship between gene expression, environmental exposure and disease susceptibility[1]. Traditional biostatistical approach, despite the sound theoretical foundations, do have limitations for detecting non-linear patterns and interactions[2]. Plus, because of the genetic heterogeneity inherent in most diseases and cross-center or cross-cohort heterogeneity in biomedical studies, identifying the true genotype-phenotype relationship is of great complexity. To address the challenges of significant heterogeneity, gene-gene interaction, gene-environment interaction, and non-linear underlying patterns, we need to develop more computational methods.

Efficient and effective algorithms for predictive modeling have been developed to address the challenges of complex gene expression data. Random forest (RF), introduced by Breiman[3], is a very popular and efficient machine-learning tool coming from assembling classification or regression trees. In its original form, RF can handle both categorical and continuous variables and can be widely applicable for classification, regression and clustering problems.

Random forest is popular because of its excellent performance in predictive ability[4]. It has several significant strengths in biomedical domain. One advantage is that RF can handle high-dimensional gene expression data but still quite robust to overfitting[5]. Secondly, the hierarchical tree structure may uncover interactions among genes and/or environmental factors that may not show significant marginal effect[6]. RF is useful for studying gene-gene interaction or gene-environment interaction because it doesn't demand a pre-specified model with all interaction terms. In other words, RF can let the data tell what the model is rather than fit the data into a pre-conceived model. Further, RF can detect non-linear patterns of genotype-phenotype relationships. In addition, tree-based methods are suited to deal with certain types of genetic heterogeneity, since early splits in the tree define separate models for subsets in the data[7]. Last but not the least, there are some high-quality and free implementations of RF. One example is a R package *randomForest*, from Liaw and Wiener, which is based on the original Fortran code from Breiman and Cutler[8].

In Chapter 2, we described an applied project, in which RF was used as the basis of the prediction algorithm of a high-throughput diagnostic system for determining absorbed dose of ionizing radiation from gene expression analysis of whole blood RNA. We demonstrated the challenges of non-linearity, gene-gene interaction, gene-environment interaction and high heterogeneity in this dataset. RF was used to deal with those challenges. However, the primary limitation of RF and other tree-based methods is the nature of greedy algorithm. That is, the algorithm finds the best single variable for the root node before adding additional variables as nodes in the model[1]. To model a general interaction effect between environment and across all genes, we forced the root node of RF models to be the environmental predictor. To further solve the heterogeneity, we proposed a framework of nested modeling to model subsets of samples and then assemble them. These techniques showed promises in improving the prediction accuracy of the prediction algorithm.

1.2 Biomarker Discovery

In the process of modeling the true genotype-phenotype relationships, we are usually facing a challenge of identifying relevant biomarkers. Biomarker discovery for a given disease or condition is important for many reasons[9]. Firstly, measurement of biomarkers requires advanced technology and can be expensive and time consuming in testing. For early diagnosis or detection of a disease, especially in situation where instant decisions need to be made, we prefer to measure a small number of biomarkers only. More importantly, testing on the most relevant biomarkers avoid overfitting and lead to the most accurate decision. Moreover, we can gain a deeper insight into the biological mechanism or underlying processes that generate the data. Therefore, computational method for biomarker discovery is essential for a good predictive model.

Using RF as the learning method, there are some algorithms developed for feature selection, e.g., VSURF[10] and varSelRF[11]. Both two methods consist of multiple iterations by adding or eliminating the features according to their variable importance (VI) scores at each iteration. Starting from ranking VI score from an ordinary RF, they select features to fit RF iteratively, and return the features that lead to the smallest out-of-bag (OOB) error.

In the RF framework, the most widely used VI score for a given variable is the increasing in the mean squared error (MSE) for regression and misclassification rate for classification when the observed values of this variable are randomly permuted in the OOB samples. Both methods rely on the permutation based VI scores, which is believed to be more reliable than the total decrease of node impurity measure[12].

The algorithm of VSURF firstly ranks all explanatory variables by VI score and retains all variables with VI score above a certain threshold. To find important variables for interpretation purpose, it starts from the most important variable, adding one variable at a time sequentially to the model. The algorithm finds the set of variables that leads to the model with the least OOB error. In this way, important variables are found even with some redundancy. To select a smaller set of variables for prediction purpose, a stepwise variable introduction strategy is introduced. Among the variables selected from the interpretation step, each variable is entered into the model sequentially. The variable is kept if the OOB error drops otherwise it is removed. In this way, the algorithm finds a sufficient parsimonious set of important variables for prediction.

Unlike VSURF, the method varSelRF is used for classification data only. In the method, instead of considering one feature at a time, the authors eliminated a fraction of features with the smallest importance score, e.g., 1/5, at each iteration. Thus, varSelRF is less computationally demanding than VSURF, however useful features with small importance scores may be eliminated. The key point of these RF-based methods is that they are fully non-parametric and free from the usual linear assumptions, while they keep all the advantages of RF performance. However, ensembles of RF demand a great amount of computation.

These RF-based feature selection algorithms show the benefits of RF for biomarker discovery. We implemented VSURF to select genes to predict absorbed dose for the radiation biodosimeter. However, beyond the model fit, we need to take into account some constraints and issues based on domain knowledge. One challenge is to incorporate those constraints or domain knowledge into those data-driven feature selection algorithms. Domingos suggests use of domain knowledge as the most promising approach for constraining knowledge discovery and for avoiding overfitting[13]. For example, Jin proposed a new approach of knowledge-integrated biomarker discovery to overcome the obstacles of data noise in Mass Spectrometry analysis[14]. They built up a protein-protein interaction network for cardiovascular disease and put pairs of biomarkers into Support Vector Machines (SVM) for classification. Similarly, Zhou derived gene-gene mutual information and combined with protein-protein interaction networks by a boosted tree regression method to discover disease-associated genes[15]. These methods showed improvement of model performance by the inclusion of knowledge-based information. However, they emphasized more on the development of knowledge network but there was no systematic way to select subsets of biomarkers that lead to the best model fit.

In Chapter 3, we proposed a method of feature selection to incorporate domain knowledge for biomarker discovery, based on guided regularized random forest[16]. We used an embedded way for feature selection and penalized each biomarker by domain knowledge at each splitting node of RF. Using cross-species prediction as an example, we showed how domain knowledge can help to improve the animal model prediction on human data. We also generalized this method to simulated scenarios, using intrinsic data characteristics as alternative of domain knowledge.

1.3 Prediction Interval

Lastly, in bioinformatics and many other domains, researchers develop models to predict the value of a quantity of interest from a number of observable variables. However, in many applications, decision makers need not only an accurate prediction but also the precision of that prediction. This precision is often represented using a prediction interval (PI). While PI can often be readily generated for linear models when using common statistical models, creating PI for complex models is often more challenging and typically requires the use of non-parametric methods. Existing methodologies are few and limited in scope. For example, quantile regression forest (QRF), utilized the full conditional distribution coming from RF to generate PI[17], is overly conservative in its prediction coverage and has limited generalizability. In Chapter 4, we modified QRF and proposed a non-parametric method, RFerr, to create PI. This method is widely applicable and especially useful for complex models. We compared RFerr with QRF on several benchmark datasets and simulated datasets, and a real-word dataset from the biodosimeter. We found this novel approach improved PI with more accurate coverage and better precision.

To conclude, the goal of this thesis is to demonstrate the use of RF to fit complex data of high heterogeneity, non-linearity relationship and interaction. Further, we extend the framework of RF to novel methods for feature selection and generating prediction interval.

CHAPTER 2

ALGORITHM DEVELOPMENT OF A BIODOSIMETRY: APPLICATION OF RANDOM FOREST

2.1 Introduction

2.1.1 Significance of a Biodosimeter

In the event of a large-scale radiation exposure, accurate and quick assessment of absorbed radiation dose would be desired for early triage and individualized medical treatment. In such an event, the vast majority of individuals would receive low and biologically insignificant dose[18]. With the help of dose estimates, we can rapidly triage people for prioritized medical management[19] and to reliably assist in decision support for personalized medicine. According to some clinical guidelines, the treatment of radiologic victims should vary with dose estimates, exposure scenarios and presenting symptoms[18]. It is suggested that a short-term therapy with cytokines is appropriate when the exposure is relatively low, while a prolonged therapy with cytokines, blood transfusion, and even stem-cell transplantation would be more appropriate when exposure dose is high[18]. Therefore, the dose information provided by a biodosimetry is essential to achieve the most effective and efficient treatment, by identifying victims who would benefit the most by a certain medical intervention.

Exposure to radiation generally leads to few immediate visible clinical signs, e.g. vomit. But it can severely cause damage to vital physiological functions and produce long-lasting health consequences among survivors[20]. Conventional biomosimetries integrate physical and clinical measurements to assess dose but have practical limitations.

For example, lymphocyte count can be used as an indicator to radiation dose[21]. However, because of the large variation in lymphocyte counts among normal individuals, this method generally requires repeated measurements over a prolonged period of time. Cytogenetic biodosimetry (CB) is another widely accepted method for dose assessment, which also requires about 4-5 days for measurements[22]. There are two more advanced algorithms utilizing clinical information to provide radiation assessment[23], however, a high-throughput, quantitative assay and a prediction algorithm based on biomarkers is more desired. Currently, the dicentric assay is considered to be the gold standard for radiation biodosimetry. Although new approaches, such as automation of DNA repair and cytogenetic assays[24], protein biomarker[25,26] and metabolomics methods[27] are being developed to improve the assay, it is still time consuming and requires sophisticated equipment and highly trained personnel. Therefore, our goal is to develop a high-throughput biomosimeter that can easily provide rapid and accurate prediction of dose in response to the radiation disaster.

Gene expression changes measured in easily accessible peripheral blood (PB) samples show promise for radiation biodosimetry. Therefore, we developed a high-throughput diagnostic system for determining absorbed dose of ionizing radiation in the range from 0.5 – 10 Gy, based on gene expression analysis from whole blood RNA. The diagnostic system is designed to identify radiation-exposed individuals and assess radiation dose, especially during the first few days after exposure. The system can process patients' blood samples and quantify gene expression through a high-throughput system. A prediction algorithm is then used to predict absorbed dose from gene expression. Medical professional reports concerning dose estimate would be generated for physicians and patients for further treatment.

2.1.2 Experimental Design

Biodosimetries were developed by several groups using various models, including human blood samples irradiated *ex vivo*[28,29,30,31] and blood samples from mice irradiated *in vivo*[32,33,34,35]. None of these models were satisfactory for the use in healthy human[36]. The *ex vivo* model, in which blood samples from healthy human were irradiated outside the body and cultured under lab conditions, is able to recapitulate some acute dose-response seen in patients exposed *in vivo*, but cannot capture the full response of a complete organism. Mice *in vivo* model allows detailed dose-response testing and is more representative to a realistic scenario, but they are phylogenetically removed from humans.

The most reliable biodosimetry would rely on human blood samples irradiated *in vivo*, but it is not ethical or practical to irradiate healthy people. The majority of human subjects who absorb radiation 1 Gy or more are usually accompanied with certain health issues, such as cancer, burns, or broken bones. These conditions could potentially confound their transcriptional profile. Moreover, the dose and post-exposure sampling time of the radiation from treatment is naturally different from a radiation disaster. In treatment, the radiation is usually delivered several times at a small fractional dose, and often targeted on a specific area of the body. While in a mass-exposure event, the radiation is exposed at a single time and uniformly absorbed by the body.

Animal models can be built in a more controlled system than is possible with humans[35]. Due to the close phylogenetic relationship to humans, samples from nonhuman primates (NHP) are more preferred to develop a biodosimetry intended for human. However, there have been few reports using NHP models for radiation biodosimetry development. Therefore, we collected NHP samples irradiated *in vivo* to build a NHP model. Specifically, we randomly assigned NHPs into several groups to receive a certain amount of radiation. Our data mainly came from two labs. NHPs were irradiated by Cobalt 60 irradiator in *Citox* lab at each of the dose level of 0, 2, 4, 6, 7 and 10 Gy (n=12 per dose). PB samples were obtained from these irradiated NHPs 24 hours before irradiation, and again on the 1, 2, 3, 5 and 7 days after irradiation. In parallel, in *ROTR* lab, NHPs were irradiated by LINAC irradiator at each of the dose level of 0, 2, 4, 6 and 8 Gy (n=20 per dose), and were irradiated at each of the dose level of 1 and 10 Gy (n=10 per dose). Besides, we also included some additional datasets, including blind samples, multigeneration samples and fractional dose (FD) samples that were part of other studies in *ROTR* lab. PB samples at *ROTR* lab were obtained from irradiated NHPs three days before irradiation, and on the 1, 3, 5 and 7 days after irradiation. Summary of sample size from different sources is shown in Table 1 and Table 2.

2.1.3 Quantification of Gene Expression

All PB samples were processed by the high-throughput system in the clinical lab network. At the earlier stage of this project, gene expressions were measured by high-throughput microarray and RNA-seq. A large number of genes were measured. The predictability of each gene was examined through their dose-response curve and univariate analysis. The most dose-responsive gene and/or the best-fitted gene were selected. Among them, genes with high baseline variability or strongly confounded by age/gender/disease were excluded. At last, 79 biomarker candidates as well as 9 reference genes were targeted for qPCR assay development. This 88-gene panel was selected by integration of comprehensive datasets and confounder databases. It covered a broad range of radiation-related biological pathways, providing a great robustness of biodosimetry models.

In the modeling process, mRNA expressions of the 88 genes were quantified by reverse-transcription real time PCR (Polymerase chain reaction). PCR is a technique used to amplify a single copy or a few copies of a piece of DNA. At each cycle, it doubles the number of DNA fragments in the sample. To measure gene expression, RNA is converted to cDNA and pre-amplified. Then in qPCR we measure the number of cycles it takes for the abundance of cDNA to exceed a certain cycle threshold (Ct). If there is a large amount of cDNA at the start of the reaction, fewer cycles will be required to accumulate enough products to cross the threshold line. Higher Ct value indicates lower abundance of a given gene.

To reduce the variability from sample to sample, a reference gene is used to normalize the quantifications of gene expression. PPP6R3 is the gene we measured that has the least variability and is used as a housekeeper in our study. Ideally the reference gene should not respond to the external environment. As shown in Figure 1, PPP6R3 is constant across doses on early days. However, Ct value of PPP6R3 increases with dose, especially on day 7. That is because, on day 7, the lymphocyte count decreases in blood samples, so does the RNA abundance of all genes, including PPP6R3. But since the effect of lymphocyte count is beyond the effect of genes, the normalized value (ΔCt) still allows for different samples to be compared. To reduce the sample-to-sample variability and capture the dose-response of the raw Ct, we use ΔCt as the main predictors for modeling. Numeric value of PPP6R3 was categorized to indicate low, medium, and high level of lymphocyte count and was used as a candidate predictor.

$$\Delta Ct_i = Ct_i - Ct_{PPP6R3} \quad \#(2.1)$$

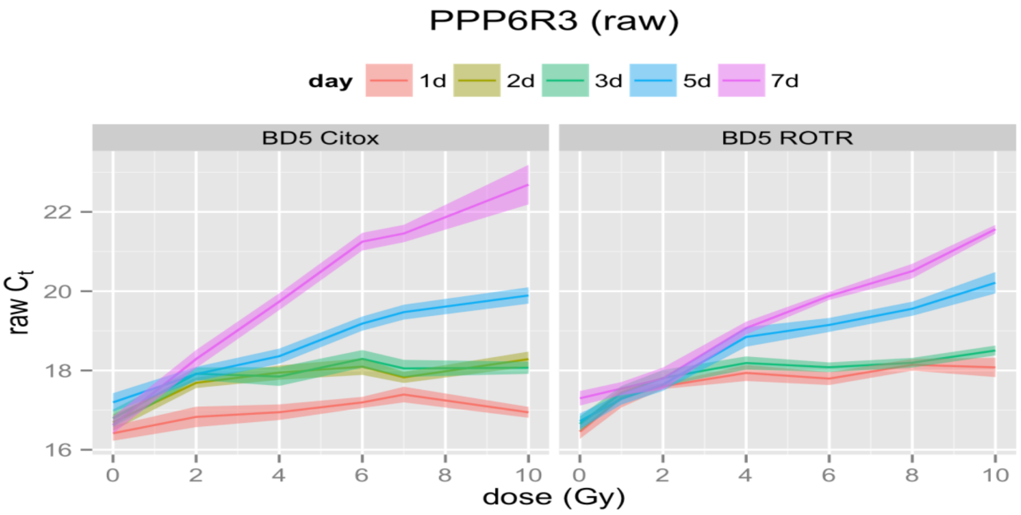

Figure 1 Dose-responsive Relationship of Reference Gene. Two plots are from two labs, Citox and ROTR, respectively. Variability of PPP6R3 was observed between the two labs. Lines of different colors represent dose-responsive curves on different day.

2.1.4 Overview of datasets

Table 1 Number of NHP from Difference Sources by Dose (Outlier Removed)

Datasets	Radiated dosage (Gy)											Total
	0	1	2	3	4	6	7	8	10	12	13.2	
Citox	12		12		12	12	12		12			72
ROTR	20	10	20		20	20		20	9			119
FD										3	4	7
Multigeneration	12				23	22						57
Blind	3			2								5
Total	47	10	32	2	55	54	12	20	21	3	4	260

Table 1 and Table 2 shows the number of NHP evaluated for gene expression at each dose level and the number of observations at each time point from various sources, respectively. The outliers were removed if they had missing or invalid values for the housekeeping gene. Samples with missing data in three genes or more were removed or sent back to measure again. The rest of missing data were replaced by imputed value. Missing data was imputed by random forest and we used *rfImpute* function in R package *randomForest*. The final dataset that included 1357 observations from 260 animals was used to build a predictive model for predicting radiation dose.

Table 2 Number of NHP from Difference Sources by Day (Outlier Removed)

Datasets	Pre-irradiated		Days after exposure					Total
	-3	-1	1	2	3	5	7	
Citox		72	71	68	72	72	70	425
ROTR	119		118		119	118	116	590
FD	7		7	3	4	7	6	34
Multigeneration	57		57		57	56	56	283
Blind	5		5		5	5	5	25
Total	188	72	258	71	257	258	253	1357

2.1.5 Overview of the full algorithm

The ultimate objective for the biodosimeter is to facilitate diagnosis and management of radiation harm to human. But since the prediction algorithm is built from NHP models, additional algorithms are needed prior to applying the device on human.

In the event of a radiation disaster, we would firstly draw blood samples from people who are potentially affected and the gene expression will be quantified by the high-throughput qPCR. Preliminary data cleaning includes quality control check and multiple imputation for missing data. A species-conversion algorithm will be used to map human gene expression to NHP gene expression. For patients who are under GCSF administration, an additional GCSF-conversion algorithm will also be applied. After data cleaning and required conversion, the predictive model built with NHP data can be used to predict radiation dose and provide an interval along with the point estimate. Figure 2 presents an overview of the algorithm framework.

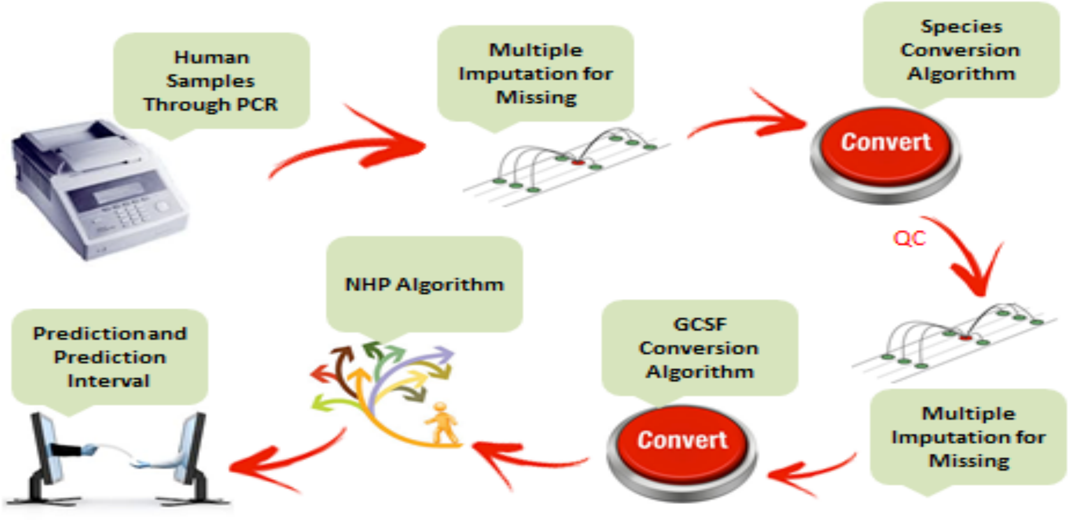
Human Samples Through PCR
Multiple Imputation for Missing
Species Conversion Algorithm
Convert
QC
Multiple Imputation for Missing
GCSF Conversion Algorithm
Convert
NHP Algorithm
Prediction and Prediction Interval

Figure 2 Overview of the prediction algorithm. Quality control criteria, multiple imputation models, species conversion algorithm, and GCSF conversion algorithm are applied before we use the NHP algorithm to make a dose prediction for a human sample.

2.1.6 Challenge with the available experimental data

Using the available animal data, we built predictive models to predict radiation dose from gene expression. However, fitting the specific data has several challenges.

1) The dataset is high dimensional with numerous gene predictors. These genes are to some extent correlated with each other.

2) There is a large variability in this dataset. Animals were irradiated by different radiation sources at different labs. There is a large biological difference among the animals, and there is a large technical variability from the experiments.

3) Dose prediction should be continuous. However due to the experimental constraints, radiation was only received at certain discrete levels, i.e., 0, 1, 2, 4, 6, 7, 8, 10 Gy.

4) We have repeated measures at different time points, so observations within one animal are correlated.

5) There are interaction effects, between day and genes, and between gene and gene. The relationships between genes and dose vary across days. Different sets of genes were dose-responsive on different days.

6) The relationships between genes and dose are not linear. Different sets of genes were dose-responsive at different dose levels.

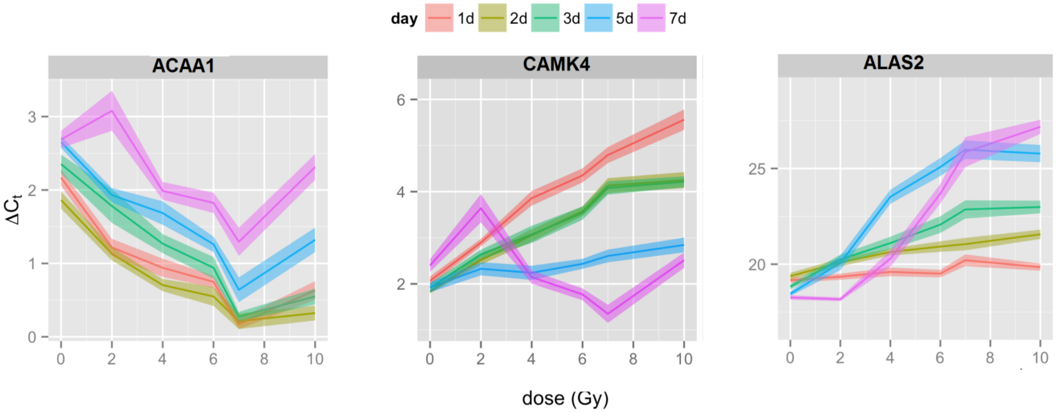

Figure 3 Day Effect on Dose-responsive Relationships. Dose was plotted against ΔCt of some representative gene. Lines with different colors represented the dose-responsive curves on different days. Width of ribbons indicated standard error. For example, ACAA1 was an up-regulated gene on all days but the abundance of ACAA1 decreased on later days. CAMK4 was a down-regulated gene on the first few days but was less dose-responsive on later days, while ALAS2 was on the opposite.

Challenge 1: Day Effect

Although RF is well known for its excellent predictive ability, there were still some challenges that cannot be easily dealt with by RF. One big challenge was the interaction effects between day and gene expression. We observed a day effect on gene expression for all genes. That is, gene expression at the same dose level differed across days. Moreover, there were interaction effects between day and most genes. The relationships between dose and genes depended on day. Exploratory dose-responsive curves were examined before modeling. Figure 3 displayed some representative plots showing the relationships between dose and gene expression across days. For example, a day effect was observed for gene ACAA1. The expression of ACAA1 was consistently lower on later days. The dose-responsive patterns were almost the same on all days though. In contrast, for most of the other genes, the dose-responsive patterns differed across days. Gene sets that were dose-responsive were different from day to day. For example, ALAS2 was not responsive to radiation on early days, but was more dose-responsive on later days. While CAMK4 was more dose-responsive on early days than on later days.

Challenge 2: Non-linear Dose Response

The second challenge of modeling was the non-linear relationships between gene expression and dose. Genes that were responsive to higher dose may not show its importance overall.

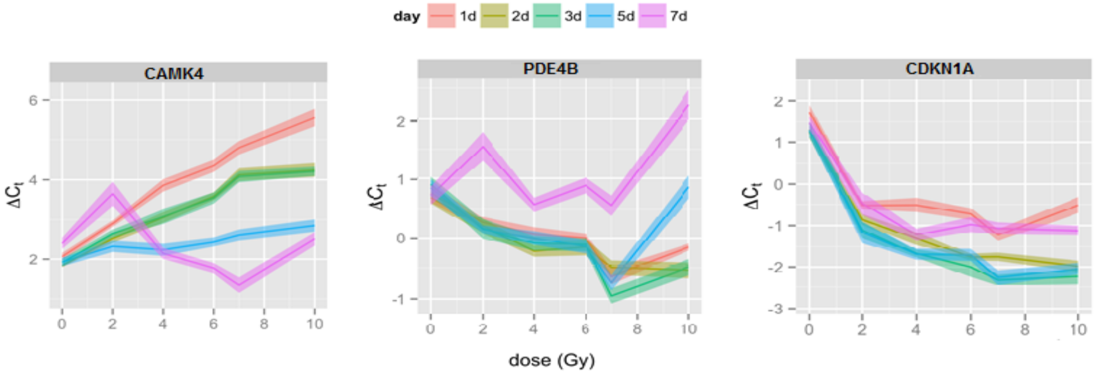

Figure 4 Non-linear Dose-responsive Relationships. ΔCt of some representative genes were plotted against delivered dose. Lines with different colors represented the dose-responsive curves on different days. Width of ribbons indicated standard error.

The non-linear trends were observed in most of genes. For example, as shown in Figure 4, CAMK4 on day 1 had a linear relationship with dose, but not on the other days. Because of the non-linearity, genes tended to respond to radiation only at a certain dose level. For example, PDE4B was more responsive at higher dose range while CDKN1A is more dose-responsive at lower dose range. Technically RF can handle the non-linearity in a nice way, but with a large number of candidate genes, feature selection added more problems to the non-linearity challenge. Different sets of genes need to be considered for different dose ranges.

Challenge 3: Heterogeneity of Data

Another big challenge was the large variability in the data. Variability came from a biological difference among animals, different protocols and radiation sources used in different labs, and technical variability in measurement and experiments. Modeling the heterogeneous data in one model was inferior. To deal with the large variability and handle feature selection at different dose ranges, we need more homogenous subgroups. For example, animals with similar radiation response can be grouped together, or animals from the same labs can be grouped together. Within more homogenous groups, we can build submodels to predict radiation dose. By grouping similar animals together, we would be able to remove some noise and make better predictions within each homogenous group.

2.2 Methods

2.2.1 Early Work

Some early work was done to model the radiation dose using mice microarray data. Because of the non-linear relationship between gene expression and dose, quadratic regression and 5-parameter logistic regression curves were tried to fit the data. However, due to the high-dimensionality of the data and the correlations among genes, feature selection process was extremely difficult, especially with the presence of interaction terms and quadratic terms. The regression models turned out to be too complex and were suspicious of overfitting. To deal with the non-linearity and the presence of interactions in an easier manner, we turn to random forest (RF) for a solution.

2.2.2 Modeling

To deal with interaction effect and capture the day effect, we built RF models respectively for each day, referred to naïve day-specific model. Pre-irradiated samples were combined with irradiated samples on a specific day for training. Furthermore, we tried fine-tuning to further improve the performance of each naïve day-specific model by nested modeling. To generate more homogenous subgroups, we separated samples according to their dose estimate by supervised prediction. We firstly built a primary model using RF and created subgroups with similar dose estimate. We set fixed intervals to separate samples into smaller groups. Table 3 was the grouping schema we used to separate samples. The interval length was set to either 3 Gy or 4 Gy. Using 3 Gy intervals, there were nine subgroups created and thus more submodels were built. Using 4 Gy intervals yielded fewer submodels and thus less complicated. We chose the number/interval of subgroups based on this rational: if the subgroups were too big, there would remain large variability and the nested modeling would not make a big difference. If the subgroups were too small, each subgroup may not have enough distinct response values, which was not good for a regression prediction. Moreover, models with too few training data were not robust and may be in danger of overfitting.

Table 3 Schema of Creating Subgroups Based on Dose Estimates

Grouping interval (Gy)		Subgroup
3 Gy	4 Gy	
0-3	0-4	1
1-4	2-6	2
2-5	4-8	3
3-6	6-10	4
4-7	>8	5
5-8		6
6-9		7
7-10		8
>8		9

Once subgroups were defined, we built submodels individually for each group. Feature selection was done for each submodel. For prediction of a new sample, we firstly generate an initial dose estimate from the naïve day-specific model. If the dose estimate is within the ranges of several subgroups, we make predictions using all corresponding submodels. For example, if a new sample is predicted as 3.6 Gy from the naïve day-specific model and the subgroups have an interval of 3 Gy, since 3.6 Gy is within the range of 1-4 Gy, 2-5 Gy, and 3-6 Gy, we would make predictions using all three corresponding submodels (submodel 2, 3, 4). The final dose estimate of the new sample is the average of the corresponding submodels' predictions.

2.2.3 Description of NHP Algorithm

The proposed algorithm for NHP nested model is summarized as follows:

Model training:

1) Combine pre-irradiated samples and samples obtained on day $i, i \in \{1,2,3,5,7\}$. Build a naïve day-specific model M_i to predict dose using selected genes. Features are selected using a specific feature selection process.

2) Group samples into homogenous subgroups based on predictions $\hat{Y}_{M_i}$from M_i (Grouping criteria refer to Table 3). Note that samples may belong to more than one subgroups.

3) Build a RF submodel $SM_{i,j}$ using samples in subgroup $j, j \in \{1,2,\ldots,9\}$. Feature selection is done for each submodel.

Model testing:

1) Given a test sample x_{new} obtained on day i, run an initial model M_i to get a prediction $\hat{y}_{M_{i,}new}$.

2) Determine applicable submodels $SM_{i,j}$ according to Table 3. Note that there may be more than one applicable submodels.

3) Run each applicable submodel to get a dose estimate. Take an average of dose estimates as the final prediction.

$$\hat{y}_{new} = \overline{\hat{y}_{M_{\iota,J},new}} \#(2.2)$$

To compare with the day-specific naïve models and nested models, we also applied some popular algorithms on the complete dataset, using nearest neighbor (NN), ridge regression (RR) and RF. We combined pre-irradiated samples, which were obtained before irradiation with irradiated samples obtained on all days for training (n=1357). The same set of selected genes (16 genes) and day information were used as predictors to build these models. The selection of gene panel was described later.

NN was implemented using *kknn* function in R packages *kknn*. Parameters tuned for NN included the number of neighbors considered (k). We set k to 12 and kernel function to optimal for the best performance of NN. RR was implemented using *linearRidge* function in R package *ridge* under default setting. No interaction term or higher-order term was specified in the regression model. RF was implemented using *randomForest* function in R package *randomForest*. Because there was an unavoidable randomness in RF algorithm, we repeated the RF modeling 5 times to assess the variability from multiple runs.

2.2.4 Group by Clustering

Another way to create homogenous subgroups is by unsupervised clustering. When dose information was not given for training, an unsupervised clustering algorithm can split samples into a certain number of clusters based on their gene expression. Similar to supervised model, this method generated subgroups based on their radiation response, but not the actual dose delivered.

K-means clustering algorithm was used for unsupervised clustering. K-means clustering is a method commonly used to group samples into different clusters[37]. This algorithm requires firstly determination of the number of clusters (k) in the data, and randomly assigning k numbers of centroids to the dataset. Then it clusters samples to the closest centroid according to their distance. Centroids and distances are calculated iteratively and the algorithm converges when the centroids no longer change.

Once clusters were determined, we used the cluster label as the response to build a predictive model using RF. When a new sample came in, we firstly determined which cluster the new sample belonged to, and then used the corresponding submodel to generate a dose estimate.

Using Day 1 data as an example, we implemented K-means clustering using *kmeans* function in R package *stats*. We at first determined the number of clusters (k) by examining the within-groups sum of squared errors (SSE) with the change of k. The number of clusters was set to 5 and 6 to test the unsupervised model.

To evaluate each model, leave-one-out cross-validation was used for all methods. The evaluation was repeated 5 times to capture the variability of RF. Prediction accuracy is only reported for irradiated samples (n=1098). The performance metrics we used were Mean Squared Error (MSE) and prediction accuracy within 1 Gy. Prediction accuracy at different dose levels may have different clinical implications. Accuracy at a lower dose range (0-3 Gy) is important for early triage. Accuracy at a middle dose range (3-7 Gy) is important to determine which treatment would be applicable according to the exposure dose. On the other hand, patients who exposed to extremely high-level radiation (>7 Gy) will show obvious signs and call for rapid medical intervention and further testing.

Prediction accuracy for high dose samples is less important. Due to the limited samples at higher dose, we grouped samples that are exposed to more than 10 Gy into one group.

2.2.5 Feature Selection

We used *VSURF* function in R package *VSURF* to select features for each day and each submodels separately. To generate a sufficient parsimonious panel of radiation biomarkers, we also considered some constraints and prior knowledge. Although most selection were data driven, we also tried manually excluding, adding or replacing genes.

Constraint 1: Prior Knowledge of Genes

There is an intrinsic difference in human and NHP genes. Although a species conversion algorithm was developed to map human genes to NHP genes, some NHP genes were not well predictable. Therefore, highly human-correlated genes are more preferable. Beyond the predictive capacity, we considered the cross-species correlation in biomarker discovery. If two genes were equally predictive of dose, we preferred the one that was more correlated with human gene.

Similarly, some genes are more likely to be influenced by potential confounders, such as GCSF treatment, disease, gender and age. We also considered the confounding effects. If two genes were equally predictive of dose, we preferred the one that was not confounded by potential confounders.

In practice, to incorporate the prior knowledge of these genes, we manually included or excluded some genes according to their characteristics. If an undesirable gene was selected by *VSURF*, we tried excluding it or replacing it with some other genes that are highly correlated with it. If the error rate didn’t increase significantly, we chose the set of genes without the undesirable genes.

Constraint 2: Continuity Across Models

As observed from exploratory plots, the dose-responsive curves of each gene vary across days and across dose ranges. Feature selection was done for each day and each dose range (submodel) separately. However, for the sake of interpretability, we want to keep the consistency of predictors across days and submodels. That is, same gene is desired to be used for consecutive models in a continuous manner. For example, TEX10 was selected for models on day 1, 2 and 3. ALOX5 was selected for submodel 7, 8 and 9 on day 1.

However, data-driven feature selection would not consider the continuity across models. We manually added or deleted genes in order to enforce that continuity. For example, if a gene is selected by *VSURF* for day 1 and day 3, we added it to day 2 model if it doesn’t hurt the model performance. Or if a gene is selected for submodel 2 and 4 but not for submodel 3, we manually added it to submodel 3.

Constraint 3: Total Number of Biomarkers in the Panel

As discussed before, our ultimate aim for feature selection is to generate a panel of the smallest possible number of biomarkers to predict dose. Therefore, we want to control the total number of genes that are used in the whole model. To do that, we examined how many times a gene was selected by the algorithm and in which models it was used. If a gene is only selected in one submodel on a single day, we excluded it or replaced it by an existing gene.

2.3 Results

Figure 5 shows the change of Mean Square Error (MSE) with the number of neighbors in NN algorithm. Table 4 compares the performance of NN to RR, grand all-day RF model, naïve day-specific RF models and nested day-specific RF models. Genes that were used for modeling were shown in Table 9. The selected gene panel includes 16 genes that were used on different days and different submodels. PPP6R3 was the reference gene and the binned category of PPP6R3 was used on Day 5 and 7 as a predictor. Genes marked as red were used in the naïve models. All 16 genes were used to build NN, RR, and RF on the complete dataset.

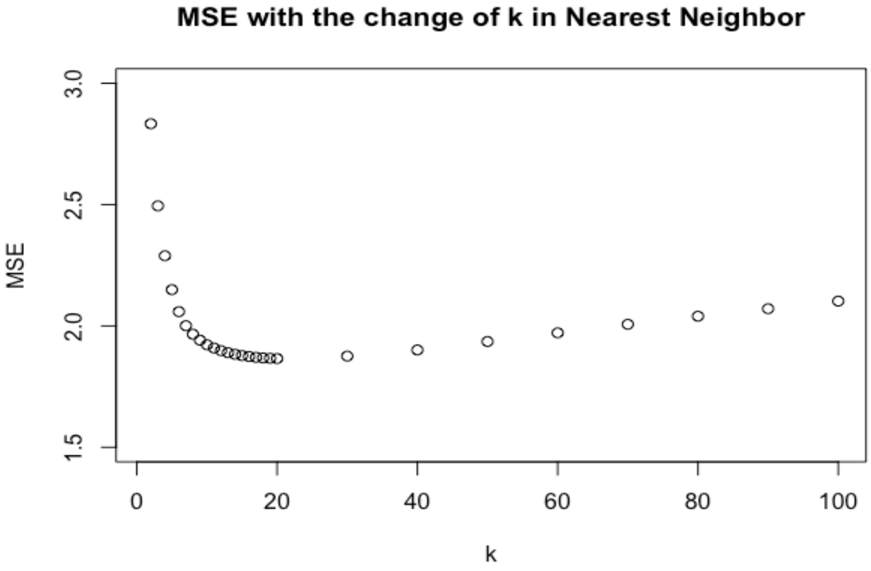

Figure 5 Parameter Tuning for Nearest Neighbor Algorithm. We observed a drop of MSE as k increased from 1 to 20, and then MSE increased with k. It suggested an overfitting when k was too large. We set k to 12 for the best performance of nearest neighbor algorithm.

Table 4 Performance Comparison of RF vs other Algorithms

Method	MSE (SD)	Accuracy <1Gy %									
		0	1	2	3	4	6	7	8	>10	avg
NN	1.90	95	83	82	63	63	65	55	49	26	66
RR	2.12	79	83	55	63	53	62	63	56	27	58
All-day RF	1.85 (0.004)	98 (0.4)	74 (2.2)	72 (0.6)	48 (5.5)	68 (0.8)	63 (1.2)	66 (2.5)	62 (1.1)	27 (1.4)	67 (0.4)
Naive Day-specific RF	1.22 (0.007)	98 (0.3)	93 (2.1)	82 (1.1)	50 (0)	72 (0.9)	75 (0.7)	63 (0.9)	66 (1.9)	38 (2.0)	74 (0.1)
Nested Day-specific RF	1.06 (0.012)	99 (0.3)	93 (1.1)	91 (1.0)	75 (5.5)	72 (0.6)	69 (0.2)	73 (2.3)	69 (1.7)	44 (0.5)	76 (0.3)

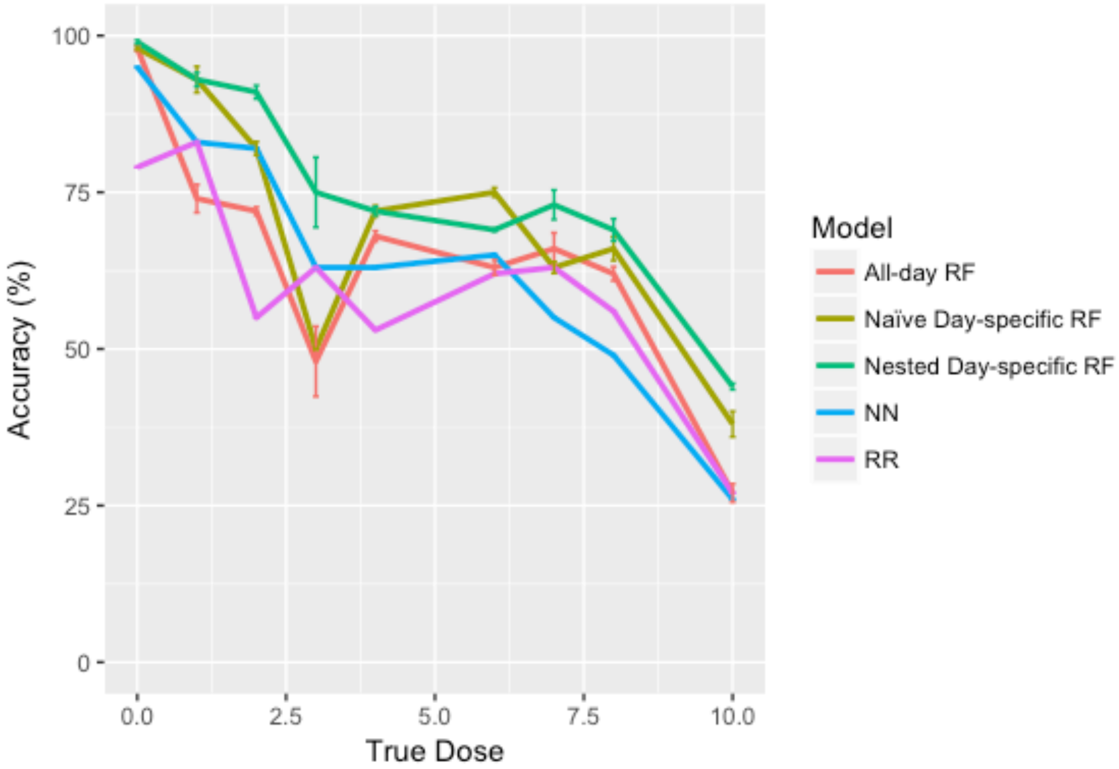

Figure 6 Model Comparison: Prediction Accuracy (%) within 1 Gy. Accuracy of RR and NN were lower than RF. There was no variability associated with RR and NN. RF-based models were evaluated five times and error bar indicated the variability. Nested Day-specific RF models outperformed all the other models.

The performance was the best using nested day-specific models. To determine the best strategy to generate subgroups, we tested several methods to build nested models. Table 5 was a comparison of all nested methods on Day 1. The naïve day-specific RF model yielded a prediction accuracy of 66%. Nested modeling using supervised prediction as the primary model worked the best. Using a fixed interval of 3 Gy improved the overall accuracy to 74%, and using a fixed interval of 4 Gy improved the overall accuracy to 72%. Nested modeling using clustering didn't improve the accuracy significantly. With five clusters, accuracy was improved to 69%. With six clusters, accuracy was not improved. Therefore, we tested the nested modeling on the other days' data using supervised prediction for grouping.

We tested the nesting modeling using 3 Gy interval and 4 Gy interval on other days. It didn't appear one way of subgrouping was always outperforming the other. In fact, the 3 Gy fixed interval method worked the best for the first three days' datasets, but worse than the 4 Gy fixed interval method on day 5 and day 7. Performance comparison was shown from Table 6.

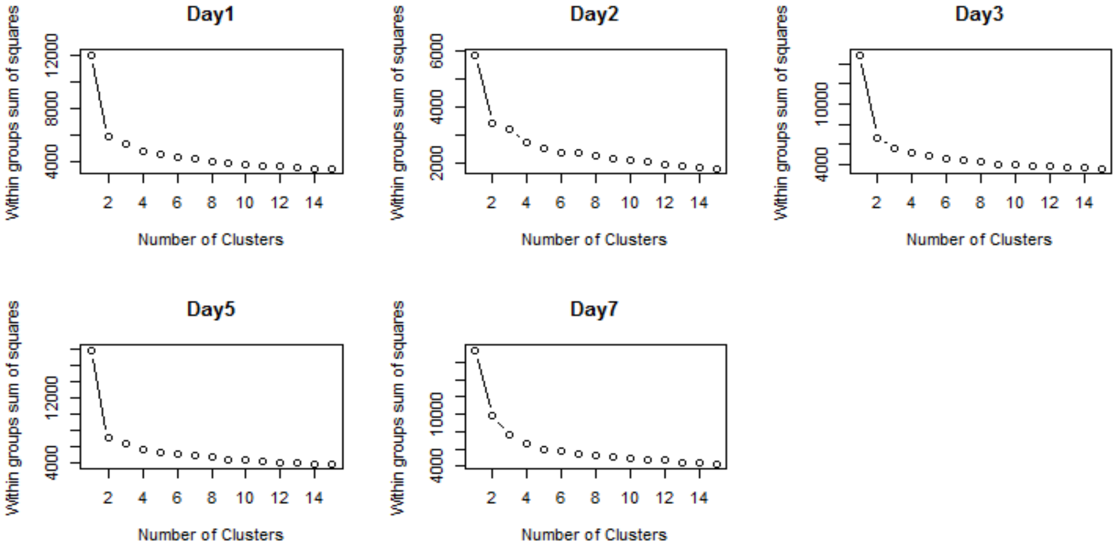

Figure 7 Within-group SSE with the Change of k. The changes of SSE with k were very similar from day 1 to day 7. With more clusters, sum of squared errors (SSE) always decreases. The best number of clusters can be found at the elbow of the curves.

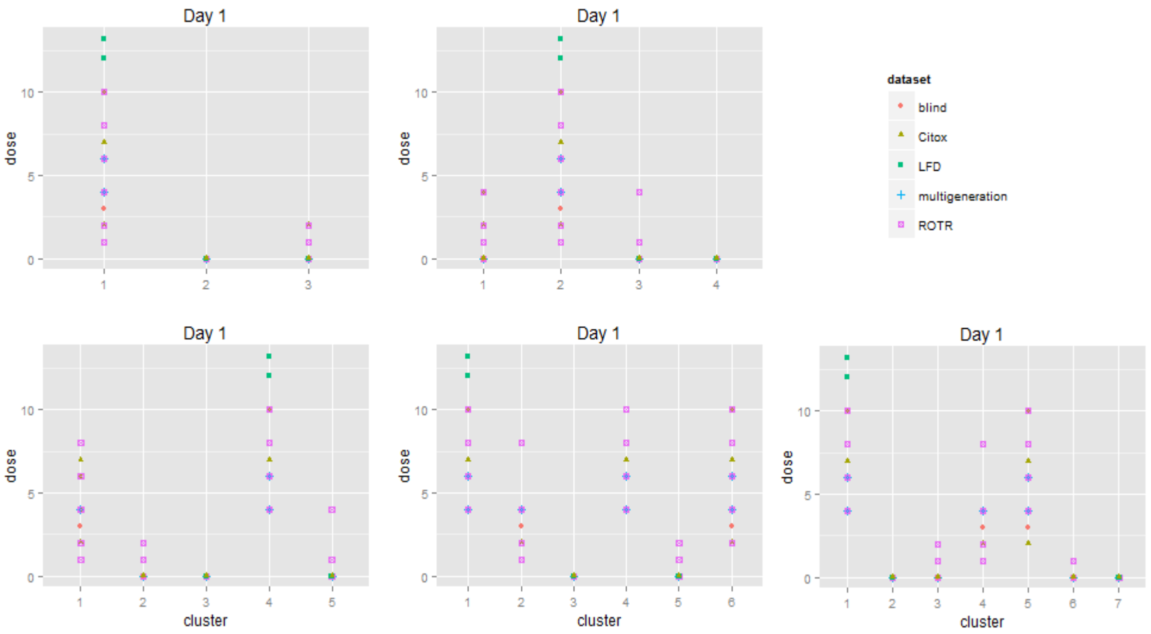

Figure 8 Clustered Samples on Day 1. We investigated the number of clusters (k) from 3 to 7. Color indicates data sources. Data from different sources were not separated well. Y-axis represents actual dose. Samples with close exposed doses were clustered.

Table 5 Model Comparison of Nested Modeling Methods (Day 1)

Model	<1 Gy %								
	0	1	2	4	6	7	8	10	avg
Naïve model	100	60	75	60	64	92	30	35	66
Nested model (3 Gy interval)	100	80	91	67	60	92	60	45	74
Nested model (4 Gy interval)	100	80	84	65	60	92	55	35	72
Nested model (5 clusters)	100	100	88	53	64	83	45	30	69
Nested model (6 clusters)	100	70	81	45	63	83	65	20	66

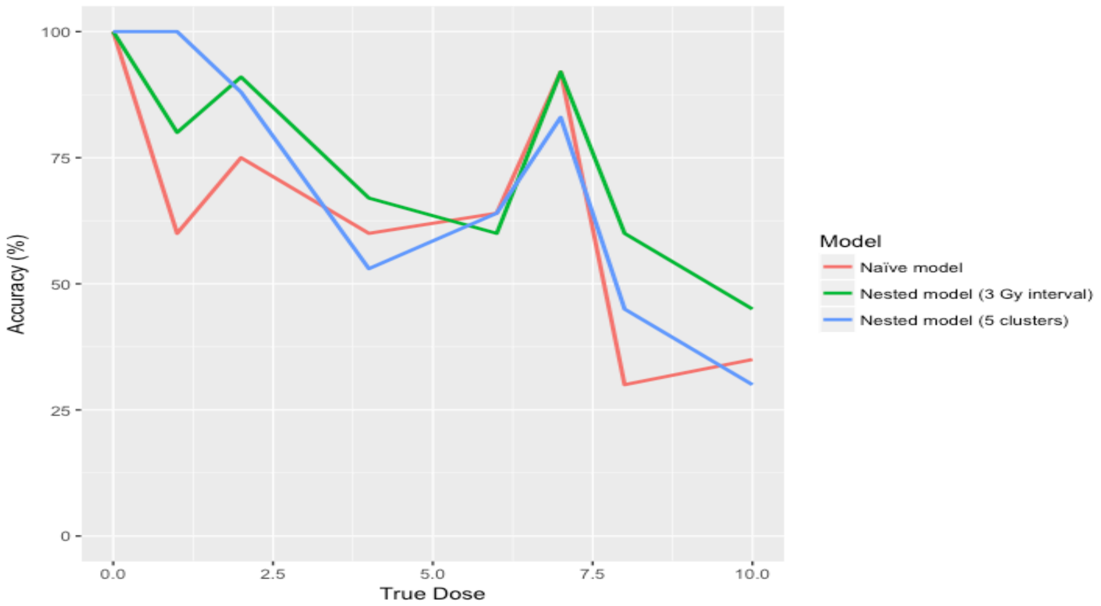

Figure 9 Nested Models Comparison on Day 1. Nested model using fixed interval in general improved the prediction accuracy within 1 Gy. Nested model using clustering didn’t improve the performance significantly.

Table 6 Model Comparison of Nested Modeling Methods (Day 3 - 7)

Methods	Day	<1 Gy (%)								
		0	1	2	4	6	7	8	10	avg
Naïve model	3	98	100	91	78	72	83	70	33	78
Nested model (3 Gy)		98	100	91	82	74	83	85	57	82
Nested model (4 Gy)		98	100	88	88	74	67	60	57	80
Naïve model	5	98	90	81	62	64	25	65	20	66
Nested model (3 Gy)		98	100	84	66	68	56	65	25	71
Nested model (4 Gy)		98	100	81	70	70	63	70	20	72
Naïve model	7	96	90	69	78	75	67	65	50	75
Nested model (3 Gy)		96	90	88	73	75	67	60	59	76
Nested model (4 Gy)		100	90	88	76	75	67	65	59	78

Table 7 Model Performance: MSE and Prediction Accuracy within 1 Gy (%)

Day\Dose	MSE	0	1	2	4	6	7	8	>10	Avg.
1	1.38	100	80	94	65	58	92	60	33	72
2	0.65	100	-	91	70	82	83	-	67	82
3	0.70	98	100	94	82	74	75	80	52	82
5	1.40	98	100	90	64	68	50	70	19	71
7	0.85	100	90	88	78	74	67	65	64	79
Avg.	1.06	99	93	91	72	69	73	69	44	76

Table 8 Model Performance: Mean Dose Estimate (Gy)

Day\Dose	0	1	2	4	6	7	8	>10
1	0.1	1.5	2.5	4.6	6.5	7.2	6.9	8.2
2	0.1	-	2.4	3.9	6.4	7.4	-	9.0
3	0.1	0.9	2.3	4.4	6.4	7.3	7.6	8.4
5	0.1	1.1	2.2	4.9	6.0	6.4	7.4	8.1
7	0.1	1.3	2.7	4.6	6.0	6.2	6.8	8.4
Avg.	0.1	1.2	2.4	4.6	6.2	6.9	7.2	8.4
SD	0.2	0.8	0.8	1.0	1.1	1.0	1.3	1.2

The model performance of the nested model was shown in Table 7. The final predictions were obtained under leave-one-out cross-validation (LOOCV). By building day-specific models and nested modeling, the prediction accuracy was improved from 66% to 76%. The largest improvement was observed on samples that received radiation at the dose range of 2-4 Gy and more than 7 Gy. As shown in the table, the biodosimetry can successfully classify non-affected population and irradiated population with an accuracy of 98%. Prediction accuracy is the highest on day 2 and day 3. As seen in Table 8 and Figure 10, dose estimates are close to the true dose delivered but prediction variability is quite large on day 5 and day 7. The dose estimates for samples irradiated at a higher dose are still likely to be underestimated.

Table 9 Model Performance: Gene Panel of the Biodosimetry

Gene	Day 1	Day 2	Day 3	Day 5	Day 7
CXXC5	X	X	X		
TEX10	X	X	X		
CD97	X	X	X		
MYC	X	X	X	X	
ACAA1	X	X	X	X	
SPECC1		X	X	X	X
PNOC		X	X	X	X
ALAS2			X	X	X
ALOX5	X	X	X	X	
CAMK4	X	X	X	X	
CDKN1A	X	X	X	X	X
COCH	X	X	X	X	
HBA2			X	X	X
HCK		X	X	X	
MOB3B			X	X	X
PDE4B					X
PPP6R3				X	X

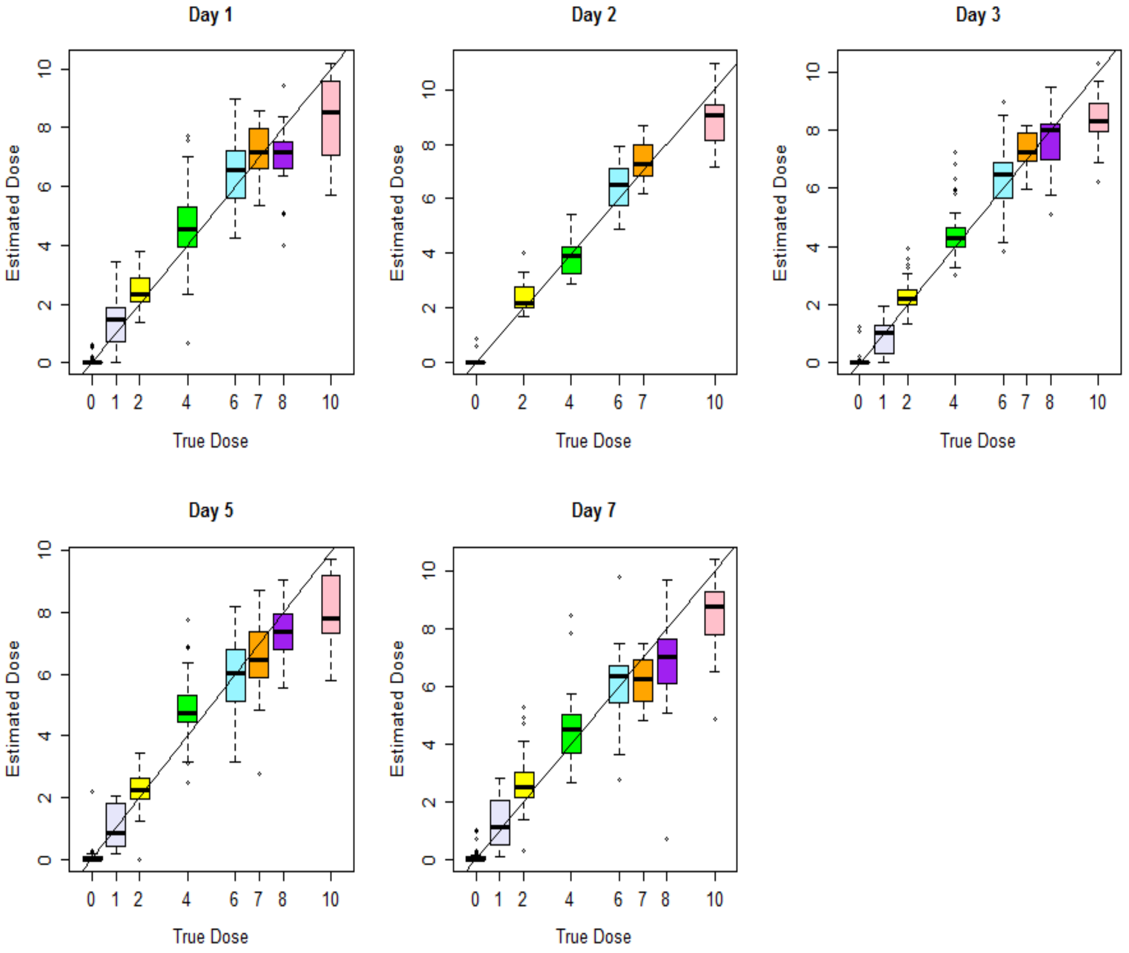

Figure 10 Model Performance: Dose Estimates under LOOCV. Estimated dose is plotted against true dose delivered. Error bars show the actual predictions for all samples. Dose levels are indicated by different colors.

2.4 Discussion and conclusion

This study validates the use of high-throughput assay on PB samples for biodosimetry applications in radiation mass causalities. The biodosimetry is designed to be used in a general scenario, regardless of the processing lab or irradiator. However, in a possible occurrence of a large radiological or nuclear event, the medical system is easily overwhelmed, so as the laboratories where the samples are processed. The use of multiple laboratories could provide assurance of the generalizability and capability of an adequate biodosimetry[38]. To meet this requirement, we modeled a full set of heterogeneous samples, collected from different labs and irradiated with different irradiators. This posed a challenge of heterogeneity for modeling.

The predictive value of the biodosimetry is critical when it is used to aid in medical triage and to manage the care of those with radiation injuries. In this applied project, we applied RF regression to predict radiation dose from gene expression. To deal with the heterogeneity of the dataset, we built five day-specific models and applied a nested modeling technique to reduce variability in each submodel. Compared to a grand all-day combined RF model, the prediction accuracy within 1 Gy was improved from 67% to 76%.

The grand all-day combined RF model is not satisfactory. One reason may be the heterogeneity in the all-day combined data. Different sets of gene were dose-responsive on different days. It was hard to capture the heterogeneity in one RF model. Such complexities influenced the model performance. Another reason may be that RF cannot handle well the interaction effects between day and gene. Because most of the genes had interaction effects with day, day was expected to play an important role in the predictive model. However, in the feature selection process, day was not as predictive as individual gene in the model if we combined all observations in one dataset. One possible reason may be that the day effect was mostly an interaction effect. The interaction of day with each gene was predictive of the radiation dose. But the marginal effect of day was not significant. Because decision tree is a greedy algorithm that only considers the information gain at the current single split, it may not consider the day factor as important as other genes. Because almost all gene effects depended on day, the day effect was beyond all gene effects. We would not want to treat the day factor the same as the other genes. Therefore, day would be better at the top level of the decision tree rather than at some random decision nodes. The prediction of the day-specific models had smaller MSE and higher prediction accuracy than the grand all-day model. The largest improvement is seen at the dose range from 1 Gy to 6 Gy, which is the most important dose range concerning clinical implications.

Nested models further improved accuracy by ensembling submodels and predicting in a more refined prediction space. We observed that the prediction accuracies for higher dose samples were often much worse than for lower dose samples. Using day-specific models, the prediction accuracy within 1 Gy were below 70% for samples irradiated by more than 7 Gy, and was only 38% for samples irradiated by more than 10 Gy. The worse accuracy for higher dose was also observed in other biodosimetry. Paul and Amundson pointed out, as doses rise above 5-8 Gy range, any biodosimetric method based on lymphocytes, including the gold-standard cytogenetic methods, will be of decreasing utility[39].

The difference in accuracy across delivered dose may be caused by the non-linearity of dose response. Paul and Amundson found similar pattern of dose-responsive behavior: the majority of the genes in their profiles were responding only at one dose, or showing increased expression at one dose and decreased expression at another[39]. It motivated us to separate samples into smaller groups with smaller dose ranges. So we can focus on improving prediction at specific dose range. By nested modeling, the accuracy of samples irradiated by more than 7 Gy was improved to above 70%, and accuracy of samples irradiated by more than 10 Gy was improved to 44%.

We also tested clustering algorithm to generate subgroups for nested modeling. Clustering methods generally didn't work well for this dataset. One reason may be because that the model used to determine clusters was not satisfactory. In clustering, although dose information was unknown, the clusters seemed to be separated by the radiation dose, rather than data sources as we expected (Figure 8). The variability caused by different sources was not removed. Because samples were not separated well in a meaningful way and clusters were subject to change from run to run, using cluster label as a response variable may only capture some noise and part of the dose information. Secondly, some clusters, especially clusters with lower dose samples, included too few response values, which was difficult for regression prediction. Thirdly, variability of responses in higher-dose cluster was still large. Predictions of those submodels could be even worse with fewer samples in each subgroup.

The accuracy improvement by nested models may be caused by feature selection. By dividing samples into smaller groups with close dose estimate, we were able to select the most relevant genes for the specific dose range. In a literature review of about 300 publications on protein biodosimetry of human exposure to ionizing radiation, different panels of gene were found to produce a unique response pattern depending on the dose and time after exposure. A panel of biomarkers, each with different dose and time optima, is highly recommended to improve individual radiation biodosimetry[40]. While considering constraints and domain knowledge, we downsized the gene panel to 16 genes, which is significantly smaller than existing methods. For example, twenty-five features were selected only to classify the unexposed and exposed samples[41]. To distinguish at a broader range of radiation dose, a 74-gene signature were used to classify between 4 doses and controls[39].

Other than the improved prediction accuracy, our biodosimetry has some general advantages, compared to existing methods. Firstly, most of biodosimetries are often used to distinguish between a few dose levels or predict in a small dose range. For example, a 3-nearest neighbor classifier built from patient data can predict samples as exposed to 0, 1.25 or 3.75 Gy with 94% accuracy[41]. The Dicentric Chromosome Assay (DCA) is considered the gold standard in biological dosimetry. It only has a dose range of approximately 0.5-5 Gy[42,43]. Beyond that, the relation between dose and dicentrics breaks down because higher radiation dose would reduce cell proliferation. As damaged cells cycle slower than undamaged cells, fewer cells can actually reach metaphase[44]. In contrast, our algorithm can predict in a continuous scale of radiation exposure from 0 Gy to 10 Gy. Secondly, we allow for a rapid, single-time measurement. Because DCA is time consuming as all scoring must be performed manually, it is difficult to implement in a mass casualty event[45]. Our biodosimetry take day a predictor and provide a dose estimate for patients from day 1 to day 5 after radiation exposure.

One big limitation of the algorithm is the complexity, especially from the nested models. The computational cost of modeling was very high. By building day-specific models, we increased the number of models to five. In nested models, we built five or nine submodels according to day. Feature selection was done for each initial model as well as all submodels. The computational cost for testing is affordable though. We only need to run one day-specific model to choose submodels and then use up to three submodels to predict dose for a new sample. The computation time of testing is more of a concern. To achieve a fast and accurate biodosimetry, this complex algorithm is a compromise.

Another concern is the chance of overfitting. Although evaluated by cross-validation, this algorithm may be suspicious of overfitting because of the complexity. To avoid overfitting, we did feature selection for each submodel. The number of genes used in each submodel was quite small so each submodel should be robust to overfitting. Moreover, the naïve model was only used to determine applicable submodels. The initial dose estimate indicated a wide range of possible submodels that may apply to the new samples. The actual dose estimates from the naïve model were not used for the final dose prediction.

In future work, we can generalize the nested modeling technique to deal with heterogeneous data in general. In nested models, we firstly build a model to create subgroups, in which the samples are more homogenous. Then we target on each homogenous dataset and build a submodel for it. By using nested modeling, we aim for creating better predictions from submodels and then improve the performance of the overall model. To generalize this methodology, we can test the technique on various datasets, varying by their heterogeneity and/or complexity and find out when this method would be the most applicable. We can also try different strategies to create homogenous subgroups and determine the number of subgroups. In this application, subgroups were created by dose estimate within either 3 Gy or 4 Gy intervals. Unsupervised clustering did not work well in this applied project because no genes are good predictors of clusters. In situation where we can successfully cluster subjects into meaningful groups, an unsupervised algorithm may be preferred. Finally, we need to find the best way to aggregate predictions from submodels. For simplicity, we took an average of predictions from submodels that were applicable. In future, we can also try weighted average by the distance to or probability of each submodel.

However, the use of nested modeling techniques should be extremely careful. Overfitting is always a concern for complex algorithms. When sample size is limited and feature set is large, we should reduce the selected features in the predictive algorithm as much as possible. In this applied project, we selected features for each submodel using a data-driven selection algorithm VSURF, while considering prior knowledge and constraints. We tried adding, removing or replacing genes manually to incorporate domain knowledge and constraints. However, it requires a lot of repeat work and we may omit useful features during the manual selection process. In biomarker discovery studies, incorporate domain knowledge into an automated feature selection algorithm is a common challenge and we address it in the next chapter.

CHAPTER 3

A NOVEL METHOD OF BIOMARKER DISCOVERY WITH GUIDED REGULARIZED RANDOM FOREST

3.1 Introduction

Feature selection is usually needed along with predictive modeling. In general, feature selection can help to provide more insights into the underlying relationships or processes by focusing on a smaller number of features; generate more reliable predictions by excluding noises; and provide faster and more efficient models for future studies and testing[46]. In most biomarker discovery applications, we typically assume that all features are equally important before the selection procedures. In reality, we usually have some prior knowledge or constraints for some features. It has been shown that use of prior knowledge induces a large gain in stability with improved classification performance[47]. However, there is a lack of systemic ways for feature selection with constraints in biomedical studies. In this chapter, we developed a novel feature selection method while considering some domain knowledge or constraints.

Traditionally, there are three general approaches for feature selection: filters, wrappers and embedded methods[48,49]. Filter type methods are usually based on the intrinsic characteristics, which determine the relevance to the target. Simple methods based on statistical test (t-test, F-test) have been shown to be effective. One common practice of these methods is to simply select the top-ranked features according to statistics, e.g., p-value or VI score. These methods can be implemented easily and efficiently. But they would also result in high redundancy in the feature sets and the cutoff is quite subjective. Another criticism is that they ignore the interaction between variables since most proposed techniques are univariate.

In wrapper methods, feature selection is "wrapped" around a learning method. They utilize the learning machine of interest as a black box to select subsets of variables[49]. The usefulness of a feature can be directly determined by the performance of the learning method. By using the learning machine as a black box, wrappers are simple but require either "brute force" search or more efficient greedy search strategies. Common practices include forward selection, backward selection or stepwise selection. VSURF and varSelRF are examples of wrapper methods using RF as the learning method. While they keep all the advantages of RF, ensembles of RF demand a great amount of computation.

"Embedded" methods implement the same idea as wrapper but proceed more efficiently. They usually define a loss and directly optimize a two-part objective function with a goodness-of-fit term and a penalty for a large number of variables. Examples include L1-regularized regression via Lasso[50] and the use of weights for each feature in linear classifier, such as SVMs[51]. These weights are used to reflect the relevance of each variable in a multivariate way. Variables with very small weights are removed from the feature set.

3.1.1 Guided Regularized Random Forest

In the framework of RF, the regularized random forest (RRF) is an example of "embedded" method. It uses RF in an embedded way to select features at each node. RRF was initially proposed to reduce redundancy for feature selection by building only one ensemble, instead of multiple ensembles[16]. The ordinary RF models have a built-in mechanism to perform feature selection at each splitting node[52]. Variable with the highest information gain would be selected at the splitting node. However, with a small number of instances and a large number of features, many features can share the same information gain at a node. Therefore, RF is likely to select a feature that is not strongly relevant. RRF applies the tree regularization to RF and can select a compact subset of features. RRF is built in a similar way as RF. The main difference is that the information gain in RRF is regularized by a penalty coefficient λ_i. When λ_i is determined from variable importance score, the algorithm is referred to as guided RRF (GRRF).

$$Gain_R(X_i, v) = \begin{cases} \lambda_i \, Gain(X_i, v) & i \notin F \\ Gain(X_i, v) & i \in F \end{cases} \#(3.1)$$

F starts from an empty set and then accumulate each selected feature used for splitting. If feature i is not selected in previous nodes, a regularization term λ_i is used to penalize feature i for splitting node v. Therefore, RRF can penalize variables based on redundancy. Penalty coefficient λ is constrained between 0 and 1. A smaller λ leads to a larger penalty. The feature with the highest penalized information gain $\mathrm{Gain_R}(\mathrm{X_i}, \mathrm{v})$ would be added to the feature set F.

GRRF guilds the penalty by variable importance of each feature. Variables whose VI score is higher tend to be penalized less. When implementing the GRRF, the authors assigned a penalty coefficient λ_i as a function of VI. VI is normalized to meet the constraint of penalty coefficient.

$$VI_i' = \frac{VI_i}{max_{j=1}^{p} VI_i} \#(3.2)$$

$$\lambda_i = (1 - \gamma) + \gamma VI_i' \#(3.3)$$

In equation (3.3), regularized coefficient $\gamma \in [0,1]$ controls the degree of regularization. A higher γ indicates a more important role that VI plays in determining the penalty. Deng and Runger found the size of feature set is decreasing with the increase of γ but is less sensitive to γ when γ is greater than 0.5. Error rate is robust to the change of γ[16].

3.1.2 Limitation of GRRF

RRF suggested an idea of penalizing features in the process of node splitting and GRRF further suggested weighting the penalty by the importance of variables. Inspired by these ideas, we incorporate domain knowledge as a weight instead of VI for feature selection using RF by applying the framework of GRRF. However, there are some limitations that need to be addressed on.

1. The variable set selected by GRRF has a large variation. Because the penalty is depending on whether or not the feature is selected in previous nodes, the order of selection brings extra randomness.
2. Unlike wrapper methods, GRRF doesn't select features while considering or optimizing the model performance. Among the 10 datasets they investigated, the prediction accuracy using features selected by GRRF is no better than an ordinary RF using all features.
3. Because there is only one ensemble built, the feature set and model performance entirely depend on the penalty coefficient. However, there is a lack of guidance on how to set the parameter and the performance is generally insensitive to the tuned parameter.

4. GRRF uses VI to guide the regularization. VI is based on RF itself and is not stable from run to run. It also points out that the permutation VI overestimates the VI of highly correlated variable[53].

3.1.3 New algorithm: Know-GRRF

GRRF, like all other data-driven feature selection methods, in not intended to incorporate domain knowledge. We modified GRRF to take domain knowledge into account for feature selection, thus we refer it to Know-GRRF. We firstly redefined the penalty coefficient λ, as a function of domain knowledge indicated in the regularization term. The coefficient defined in GRRF can regulate the size of feature sets but is not improving the model performance. We proposed a different way to determine λ.

Secondly, we implemented GRRF for regression. Similar to penalizing Gini information gain in classification, we penalize the decrease of MSE for regression problem. GRRF for regression was implemented in the *RRF* R package (V1.7) available at CRAN, the official R package archive. To reduce randomness, we set mtry to the number of all features for GRRF regression.

We developed a more generalizable method for feature selection using Know-GRRF. The biggest challenge here is to more efficiently optimize the regularization term to achieve the best model performance. We developed an “embedded” method to search the best regularization parameter while optimizing model performance. For example, we can minimize Akaike's information criterion (AIC) of the training model. Similar to L1 and L2 regularization, AIC is a two-part loss function of goodness-of-fit measure and number of features.

$$AIC = 2k - 2\ln\left(\hat{L}\right) \#(3.4)$$

$$\ln\left(\hat{L}\right) = \begin{cases} \sum_{1}^{n} y_i \ln \hat{p}_i + (1 - y_i)\ln(1 - \hat{p}_i) & (classification) \\ -\frac{n}{2} ln \frac{\sum_1^n \sum (y_i - \hat{y}_i)^2}{n} & (regression) \end{cases} \#(3.5)$$

n is the sample size and k is the number of selected features. In classification problems, $\hat{p}_i$ is the probability of being predicted as y_i. If $\hat{p}_i = 0$, it was replaced by $\hat{p}_i = \frac{1}{2n}$, or if $\hat{p}_i = 1$, it was replaced by $\hat{p}_i = 1 - \frac{1}{2n}$.

The proposed algorithm for Know-GRRF is summarized as follows:

1. Select a statistic (VI, correlation, q-value, etc) or define domain knowledge as numerical scores to reflect the relative importance of all features. Normalize the scores by dividing the maximum value of the score.

$$score_i' = \frac{score_i}{max_{j=1}^{p} score_i} \#(3.6)$$

2. Build a GRRF model. Set mtry to the number of all features and penalty coefficient as an exponential function of the normalized score.

$$\lambda_i = \ score_i'^{\delta} \#(3.7)$$

3. Build a RF model using the feature set returned by GRRF in Step (2) and compute model performance, e.g., MSE or AIC using OOB predictions.

4. Repeat Step (2) and (3) n times. Return the mean of model performance over n runs.

5. Optimize δ in equation (3.7) to minimum the mean of MSE or AIC in Step (4).

6. Set δ to the optimized value in Step (5). Build GRRF models m times and return m sets of selected features. Run a stability test by adding features sequentially according to the selection frequency and select the set of features that lead to the best model performance, i.e, smallest AIC.

3.2 Method

To test Know-GRRF, we applied it on the real-world data sets from the radiation biodosimetry research project. We incorporated cross-species correlation as domain knowledge for Know-GRRF, and compared it to some existing methods. We also test the generalizability of Know-GRRF on simulated datasets. Because domain knowledge is not available for simulation, we use intrinsic data characteristics to select features using Know-GRRF.

3.2.1 Radiation Datasets

a. Twenty-five human subjects who went through radiation as part of cancer therapy were included in this study. They received accumulate radiation dose of 3.6 Gy, 7.2 Gy and 10.8 Gy on day 1, 2, 3 consecutively. Thirty-five genes, including two reference genes, were profiled using qPCR. Genes with missing value in more than half of the observations were removed from further analysis. The final set of candidate genes for feature selection was 28 (N=115).

b. Part of NHP data on day 1 was used to train the model (Citox and ROTR datasets, N=190). The same set of 28 genes was candidate for the predictive model.

c. NHPs who received single dose of radiation at dose level of 4 Gy, 7 Gy and 10 Gy on day 1 were used to map human genes to NHP genes. (N=96).

d. Cross-species gene correlations of the 28 genes were obtained from 10 human subjects and 10 NHP subjects who received radiation at fractional dose in another study[36]. Radiation was given to human and NHP in the same way at 0, 3.6, 7.2, or 10.8 Gy. Cubic regression lines were fitted for each gene and Pearson's R was calculated across fitted values of human and NHP for at 0, 3.6, 7.2, 10.8, and 13.2 Gy.

3.2.2 Human-to-NHP Conversion Models

Because human gene and NHP gene are at different ranges, tests on human samples suggested that human gene expression values may need to be adjusted prior to application of the NHP model. A "multi-gene" approach utilizing all gene values for cross-species conversion were used before applying NHP biodosimetry models[36]. For simplicity, we developed some "single-gene" models using univariate simple linear regression models to map human gene to NHP gene one by one. NHPs who received single dose at dose level of 0 Gy, 4 Gy, 7 Gy and 10 Gy were used to train the conversion model. Mean ΔCt of a given human gene at four dose levels (0, 3.6, 7.2, 10.8) were predictors and mean ΔCt of the corresponding NHP gene at four dose levels (0, 4, 7, 10) were responses. One simple linear regression model was built for each gene. Before applying NHP model on human data, we convert human data by 28 regression models.

3.2.3 Feature Selection Methods

We used Know-GRRF to select genes for NHP model, while optimizing the performance on human data, by incorporating domain knowledge of cross-species correlation. To apply Know-GRRF, we firstly normalized cross-species correlation by equation (3.8) and penalty coefficient is modified as equation (3.9). We investigated the performance of Know-GRRF with a parameter setting: $\delta \in \{0.1, 0.5, 1, 2, 3, \dots, 18, 19\}$. At each of the setting of δ, we run Know-GRRF 10 times and retrieved 10 corresponding sets of selected features. With each set of features, we built a RF model using NHP data then applied it to predict human-converted data. To reduce the variability of feature sets, we run a stability test over the 10 sets of features. Features that were consistently selected by Know-GRRF were retained in the final feature sets. We found the best set of features at $\delta = 12$ and stability is greater than 90% (features were selected in 10 out of 10 runs). We examined the pattern of penalty coefficient with the change of δ.

$$correlation_i' = \frac{correlation_i}{max_{j=1}^{p} correlation_i} \#(3.8)$$

$$\lambda_i = correlation_i'^{\delta} \#(3.9)$$

For comparison, we also applied VSURF and GRRF on the same set of data. VSURF were used under default setting and we used the “feature set for prediction” to build NHP model. We also tested two versions of GRRF, using VI and cross-species correlation respectively. The penalty coefficient was determined in the same way of the original GRRF. To incorporate domain knowledge, we replaced VI with cross-species correlation and investigated the performance of GRRF at a parameter settings:

$\gamma \in \{0, 0.05, 0.1, \ldots, 0.95, 1\}$. Similar to Know-GRRF, we run GRRF 10 times at each of the setting of γ. Using the resulted 10 sets of features, we built 10 NHP models individually and tested on human data. We also examined the pattern of penalty coefficient with the change of γ.

3.2.4 Performance Metrics

We built NHP models with selected features by each method and calculated MSE of NHP data and human data. NHP MSE was calculated from the OOB predictions and Human MSE was from independent test. We investigated the number of features, NHP MSE and human MSE under each setting of GRRF and Know-GRRF. To measure the variability, we run both methods 10 times at each parameter setting to calculate standard deviation. The parameters (γ or δ) were set at the best performance and we compared the performance to VSURF. The final performance for each method was calculated from 10 runs of RF models using one set of features. NHP MSE was calculated by LOOCV and human MSE were calculated as independent testing.

3.2.5 Simulation

To test the generalizability of Know-GRRF, we compared Know-GRRF to several other methods on simulated datasets for both classification and regression by the following procedures.

One hundred i.i.d random variables (X_1, X_2, … X_{100}) were generated under standard normal distribution (N=200). Half of the observations were used as training set for feature selection, and returned features were applied to the other half of the dataset for modeling. With each set of features selected by different methods, we built RF models under default setting. To account for the model variability, ten RF models were built using each set of selected features and standard deviation was assessed. We compared prediction error rates or MSE averaged from 10 runs.

Table 10 True Relationship in Simulated Scenarios

Scenario	Relationships
Linear	$\mathrm{Y} = 0.3X_1 + 0.5\mathrm{X}_2 + 0.7\mathrm{X}_3 + 0.9X_4 + 1.1\mathrm{X}_5 + 1.3\mathrm{X}_6 + 1.5\mathrm{X}_7 + 1.7\mathrm{X}_8 + 1.9\mathrm{X}_9 + 2.1\mathrm{X}_{10}$
Higher-order	$Y = 0.9X_4 + 1.1\mathrm{X}_5 + 1.3\mathrm{X}_6 + 1.7\mathrm{X}_8 + 1.9\mathrm{X}_9 + 2.1\mathrm{X}_{10} + 1.7X_{11}^2$
Interaction	$Y = 0.1 + 0.9X_4 + 1.1\mathrm{X}_5 + 1.3\mathrm{X}_6 + 1.7\mathrm{X}_8 + 1.9\mathrm{X}_9 + 2.1\mathrm{X}_{10} + 1.7X_{11}X_{12}$

Six scenarios for classification or regression were simulated varying in complexity, including linear relationship, higher-order relationship and interaction. The true relationship was shown in Table 10. Response variable (Y) was dichotomized to binary for two-class classification problem.

Because there is no domain knowledge in simulated data, we used intrinsic data characteristics for regularization in Know-GRRF. We tested the use of q-value in Know-GRRF for two-class classification. Two samples t-test is done to get the p-value and q-value is the adjusted p-value using Benjamini & Hochberg (BH) method[54]. The regularization term δ in Know-GRRF was found by optimizing the averaged AIC (n=10 in Step 4) of reduced models. Results were compared to varSelRF under default and original GRRF using normalized VI at the setting of $\gamma = 0.5$ for the best performance.

$$\lambda_i = {q_i'}^{\delta} \#(3.10)$$

$$q_i' = \frac{1 - q_i}{max_{j=1}^{p}(1 - q_i)} \#(3.11)$$

For regression, we tested the use of VI in Know-GRRF. Same normalized VI scores were used for both GRRF and Know-GRRF. The regularization term δ in Know-GRRF was found by optimizing the averaged AIC (n=10 in Step 4) of reduced models. Results were compared to VSURF under default and original GRRF at the setting of γ = 0.9 for the best performance.

$$\lambda_i = {VI'_i}^{\delta} \#(3.12)$$

$$VI'_i = \frac{VI_i}{max_{j=1}^{p} VI_i} \#(3.13)$$

3.3 Result

3.3.1 Radiation Data

Table 11 Sample Size of Human Subjects and NHP Subjects

Dose (Gy)	0	1	2	3.6/4	6	7.2/7	8	10.8/10	Total
Human	42*	-	-	24	-	25	-	24	115
NHP for conversion	32	-	-	32	-	12	-	20	96
NHP for modeling	32	10	32	32	32	12	20	20	190

*Pre-irradiated samples were included as subjects receiving 0 Gy.

Table 12 Human Demographic Characteristics

Subgroup	Female	Male	White	Hispanic	Age 40+	Age<40	Total
Patient	16	9	13	3	15	10	25
Sample	72	43	59	13	70	45	115

Table 11 is the sample size of NHP subjects and human subjects. NHP subjects used for conversion were those irradiated at the closest dose level to human subjects. The complete set of NHP data on day 1 was used for modeling. Demographic characteristics of 25 patients were shown in Table 12. Accumulate radiation was given on three days.

Samples were obtained from patients before the radiation therapy and 24 hours after each irradiation. The total number of samples for testing is 115.

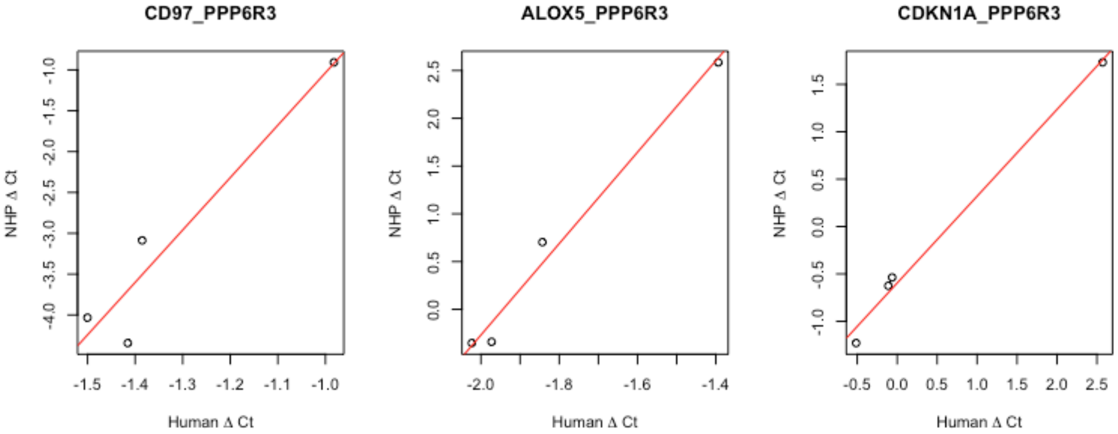

Figure 11 Regression Model for Human-NHP Conversion. Linear regression models were built to map human genes to NHP genes one by one. Plots of three representative genes are shown. The red lines represent the linear models we used for conversion. Cross-species correlation is low for gene CD97, moderate for gene ALOX5 and high for CDKN1A.

Dose-responsive correlation for human and NHP data are shown in Figure 12. There are some differences in dose-responsive pattern between NHP and human data. For example, the most dose-responsive genes in NHP data, CAMK4 and CD97, only show moderate correlations with dose in human data. On the other hand, MYC, which has the highest correlation with dose for human, is not among the most dose-responsive genes for NHP. To improve the predictability of NHP model on human data, we used the cross-species correlations as the domain knowledge for feature selection.

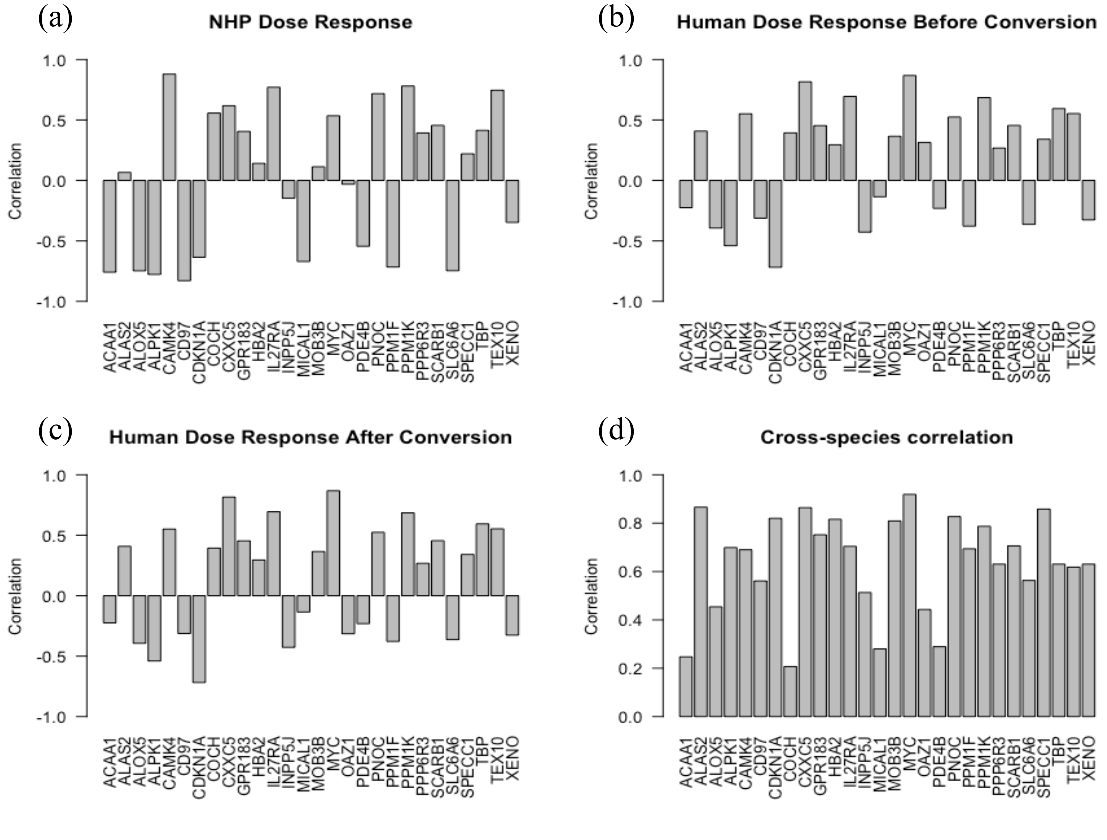

Figure 12 Dose-responsive Correlation and Cross-species Correlation. Dose-responsive genes differ between human genes and NHP genes. Figure (b) and (c) are human gene-dose correlation. Dose-responsive genes remain the same after linear conversion. Figure (d) displays cross-species correlation.

As shown in Figure 13, the number of features selected decreased with the increase of regularized coefficient γ in GRRF or δ in Know-GRRF. The number of feature set was less sensitive to γ and was still relatively large (~24) even when γ reached the maximum. On the other hand, the number of features decreased fast with δ and became really small (~3). Figure 14 showed the human independent test performance with the change of γ and δ. Human test performance was slightly better with larger γ but didn't change much at different setting of GRRF. Because the number of genes selected was fewer when γ is large, we set γ to 0.9 for GRRF. Human test performance was the best using Know-GRRF at the setting of $\delta = 12$. There was no variability in the selected feature sets among 10 runs. Thus, we set δ to 12 and stability is 100% for Know-GRRF. As suggested by Figure 15, domain knowledge played a more important role and coefficients were more differentiable when γ or δ was larger. However, γ is constrained from 0 to 1 and could reach a maximum. Thus, δ in Know-GRRF is a better parameter to control the degree of regularization than γ in GRRF.

Table 13 compares NHP sample MSE and Human sample MSE using different sets of features that were chosen by each of the method. Each method selected one set of features under a specific setting. With each of the selected feature set, 10 RF models were built and SD reflects the variability of RF models. Using all 28 genes in RF result in low MSE for NHP but high MSE for human data. VSURF and GRRF using VI score reduced the feature set but were not able to improve model performance. Incorporating cross-species correlation, GRRF reduced human MSE from 9.05 to 7.90 using a set of 13 genes. The method we proposed, Know-GRRF, reduced human MSE to 4.01 using a set of only 3 genes. NHP performance under LOOCV decreased with the use of cross-species correlation.

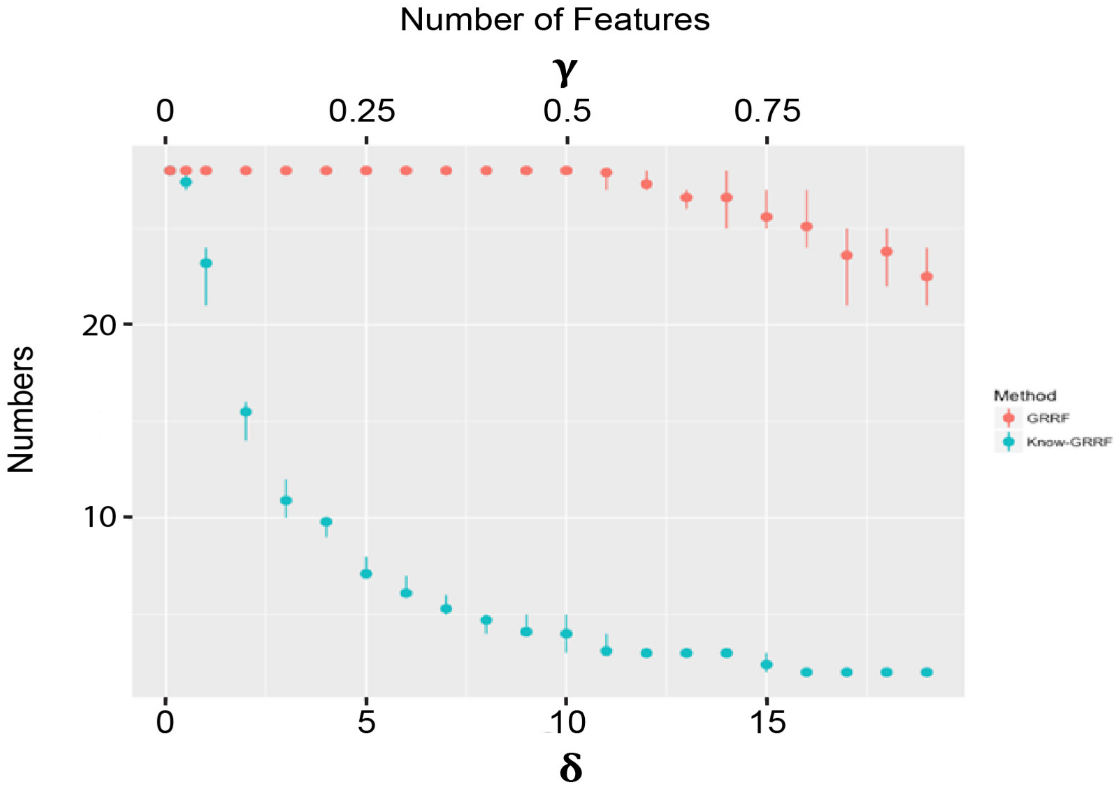

Figure 13 Comparison of Number of Selected Features using GRRF and Know-GRRF without Stability Test. X-axes are tuned parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\delta}$. Y-axis is the number of features selected. GRRF was tested with a parameter settings: $\boldsymbol{\gamma} \in \{\mathbf{0}, \mathbf{0.05}, \mathbf{0.1}, \dots, \mathbf{0.95}, \mathbf{1}\}$, shown in red dot. Know-GRRF was tested with a parameter setting: $\boldsymbol{\delta} \in \{\mathbf{0.1}, \mathbf{0.5}, \mathbf{1}, \mathbf{2}, \mathbf{3}, \dots, \mathbf{18}, \mathbf{19}\}$, shown in blue dot. Error bar shows the variability of feature numbers among 10 runs of both methods. The line ranges from the minimum to the maximum number of features for each setting. Variability reflects the selection variability.

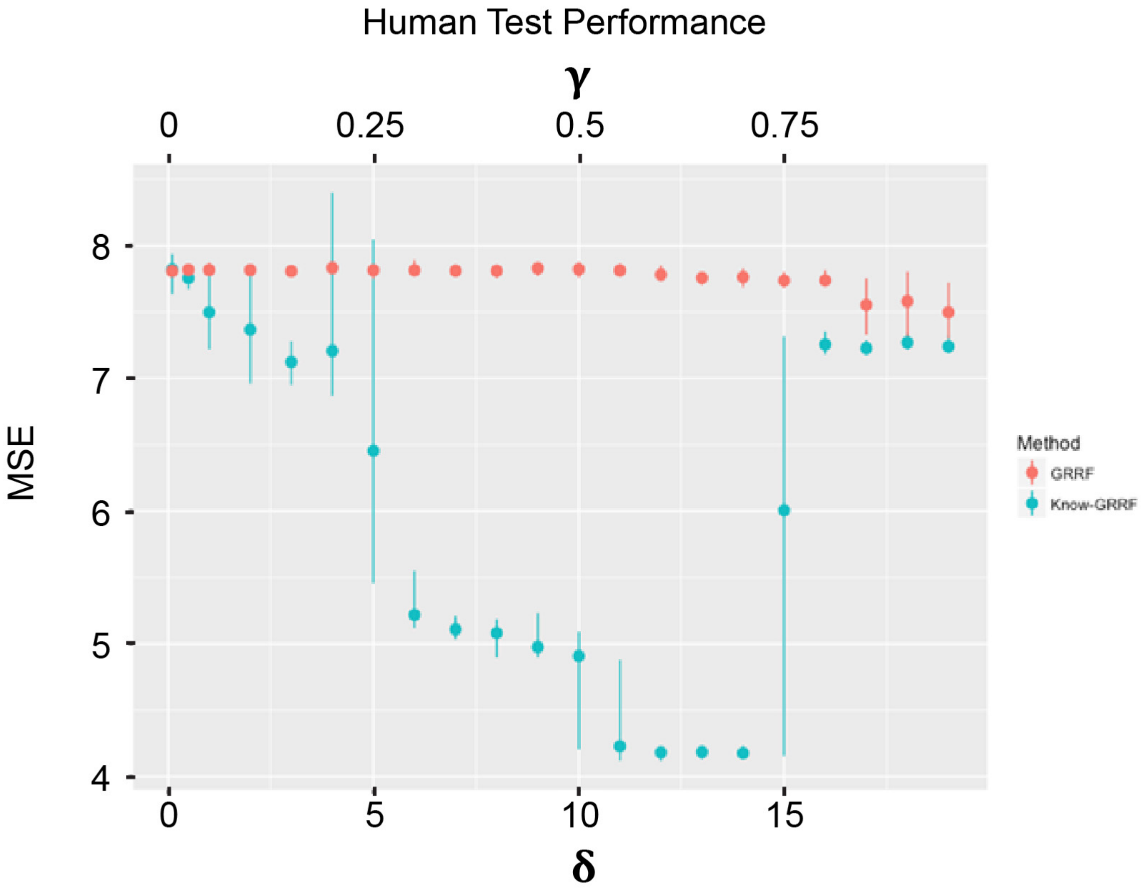

Figure 14 Comparison of MSE in Human Sample Predictions using GRRF and Know-GRRF without Stability Test. X-axes are tuned parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\delta}$. Y-axis is the MSE of human sample predictions. GRRF was tested with a parameter settings: $\boldsymbol{\gamma} \in \{\mathbf{0}, \mathbf{0.05}, \mathbf{0.1}, \dots, \mathbf{0.95}, \mathbf{1}\}$, shown in red dot. Know-GRRF was tested with a parameter setting: $\boldsymbol{\delta} \in \{\mathbf{0.1}, \mathbf{0.5}, \mathbf{1}, \mathbf{2}, \mathbf{3}, \dots, \mathbf{18}, \mathbf{19}\}$, shown in blue dot. Error bar shows the variability of feature numbers among 10 runs of both methods. The line ranges from the minimum to the maximum number of features for each setting. Variability reflects both the feature selection variability and modeling variability.

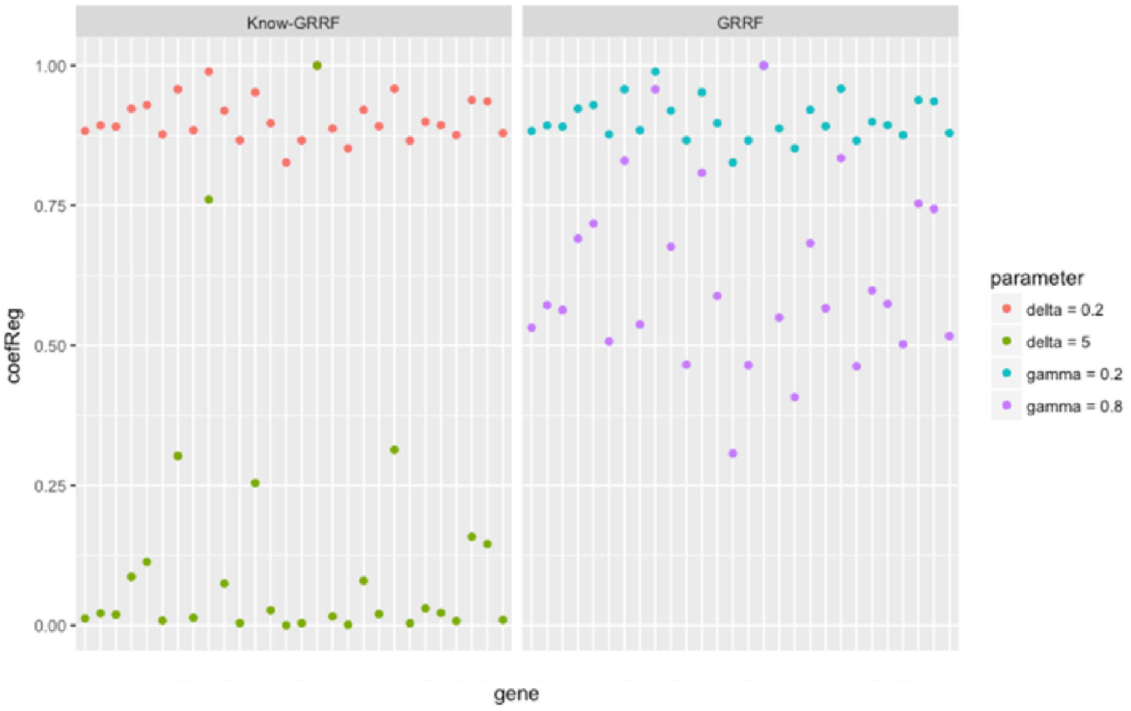

Figure 15 Penalty Coefficient for Each Gene. The left panel is coefficients calculated from equation (3.12) in Know-GRRF, at the setting of $\boldsymbol{\delta = 0.2}$ and $\boldsymbol{\delta = 0.5}$. The right panel is coefficients calculated from equation (3.3) in original GRRF, at the setting of $\boldsymbol{\gamma = 0.2}$ and $\boldsymbol{\gamma = 0.8}$. Coefficients are more differentiable when $\boldsymbol{\delta}$ or $\boldsymbol{\gamma}$ is larger. Know-GRRF can better differentiate variables and not constrained by a limit.

Table 13 Model Comparison using Different Sets of Features

Methods	Genes	MSE (SD)	
		NHP	Human
No	All 28 genes	1.52 (0.025)	7.82 (0.058)
VSURF	CAMK4, CD97, ALPK1, ALOX5, CXXC5	1.62 (0.017)	9.51 (0.160)
GRRF (VI, γ=0.9)	ACAA1, ALOX5, ALPK1, CAMK4, CD97, CDKN1A, IL27RA, PPM1K, SLC6A6, TBP, TEX10, XENO	1.59 (0.019)	8.30 (0.074)
GRRF (correlation) γ=0.9)	ACAA1, ALAS2, ALOX5, ALPK1, CAMK4, CD97, CDKN1A, COCH, CXXC5, GPR183, IL27RA, INPP5J, MOB3B, MYC, OAZ1, PNOC, PPM1F, PPM1K, PPP6R3, SCARB1, SLC6A6, SPECC1, TBP, TEX10, XENO	1.52 (0.026)	7.31 (0.064)
Know-GRRF (δ=12, stability=100%)	CDKN1A, CXXC5, MYC	3.21 (0.075)	4.01 (0.050)

Table 14 and Figure 16 show the dose estimates for NHP under LOOCV and human samples predictions using features selected by VSURF and Know-GRRF. Using cross-species correlation in Know-GRRF improved the dose estimate of human samples, especially subjects who received a high dose of radiation.

Table 14 Dose Estimate for NHP under LOOCV and Human Data (Gy) and (SD)

	Dose	0	1	2	4/3.6	6	7/7.2	8	10/10.8
NHP	VSURF	0.32 (0.40)	1.58 (1.01)	2.37 (0.77)	4.42 (1.22)	6.35 (1.00)	6.88 (0.85)	6.91 (1.19)	8.40 (0.99)
	Know-GRRF	0.33 (0.39)	2.18 (1.12)	3.54 (1.17)	4.89 (1.46)	6.01 (1.38)	5.83 (0.99)	5.96 (1.47)	7.17 (0.98)
Human	VSURF	1.90 (1.48)	-	-	4.53 (1.85)	-	5.42 (1.71)	-	6.36 (1.99)
	Know-GRRF	0.40 (0.62)	-	-	3.78 (1.11)	-	5.72 (1.49)	-	7.49 (1.48)

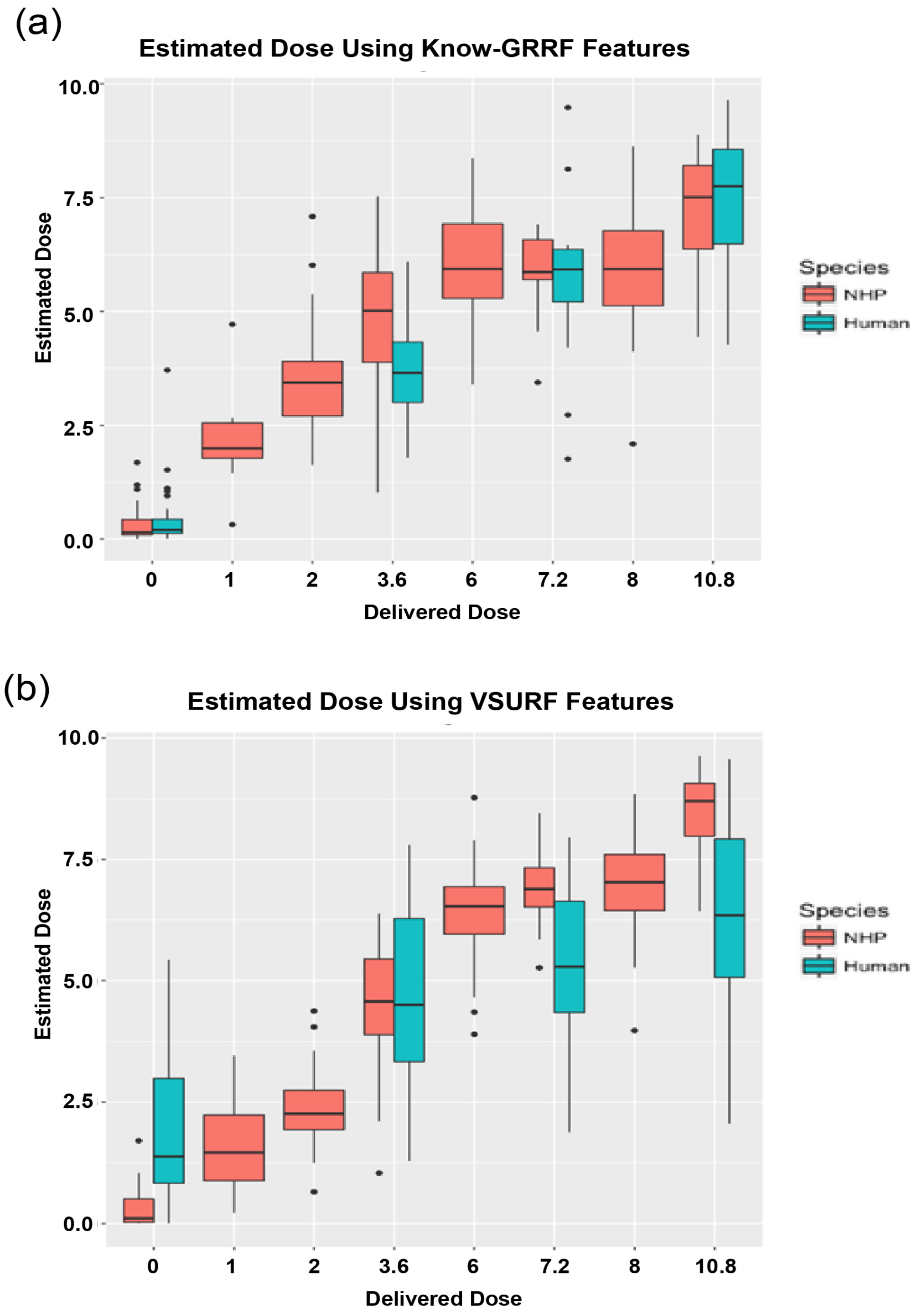

Figure 16 Dose Estimates for NHP and Human Samples. Figure (a) shows results using features selected by Know-GRRF, (b) shows results using features selected by VSURF. NHP were irradiated at dose level of 0, 1, 2, 4, 6, 7 and 10 Gy and human subjects were irradiated at dose of 0, 3.6, 7.2, 10.8 Gy.

3.3.2 Simulation Results

Table 15 Methods Comparison in Linear Classification (Scenario 1)

Method	Features	Error /SD (%)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	0.30 /0.036	-	-
varSelRF	5 9 28 36 42	0.28 /0.011	20%	3%
GRRF ($\gamma = 0.5$)	5 6 7 9 12 28 42 61	0.24 /0.020	40%	4%
Know-GRRF ($\delta = 2.30$, Stability=100%)	5 6 7 9 10 42	0.25 /0.012	50%	1%
Known	1 to 10	0.19 /0.016	-	-

Table 16 Methods Comparison in Classification with Higher-order Term (Scenario 2)

Method	Features	Error /SD (%)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	0.30 /0.012	-	-
varSelRF	4 5 9 10 30 61	0.26 /0.014	57%	2%
GRRF ($\gamma = 0.5$)	4 5 9 10 61 85	0.27 /0.015	57%	2%
Know-GRRF ($\delta = 1.97$, Stability=60%)	4 5 6 9 10 61	0.23 /0.020	71%	1%
Known	4 5 6 8 9 10 11	0.22 /0.019	-	-

Table 17 Methods Comparison in Classification with Interaction (Scenario 3)

Method	Features	Error /SD (%)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	0.31 /0.025	-	-
varSelRF	5 8 9 10 29	0.24 /0.012	50%	1%
GRRF ($\gamma = 0.5$)	5 8 10	0.27 /0.014	38%	0%
Know-GRRF (δ=0.81, Stability=100%)	5 8 9 10 52	0.25 (0.020)	50%	1%
Known	4 5 6 8 9 10 11 12	0.24 /0.018	-	-

In scenario 1, Know-GRRF returned the same set of genes from all 10 runs, so no stability test is needed. In scenario 2, we did a stability test by sequentially adding features according to their selection probability. When stability is 60%, averaged AIC was the smallest. Thus, we selected features that were outputted 6 or more times from 10 runs. Similarly, in scenario 3, when stability criterion was set to 100%, averaged AIC was the smallest. The final feature set of features that are selected consistently in all 10 runs. Figure 18 (a) shows the result of stability test in scenario 2 and 3.

In classification problem, Know-GRRF is able to select more relevant features and fewer irrelevant features than varSelRF and GRRF. True positive rate (TPR) is higher and false positive rate (FPR) is lower. OOB error from RF using feature selected by Know-GRRF is comparable or better than other methods. The optimization function can automatically find δ that leads to the least AIC. Setting stability criteria can result in less variability in the returned feature set. Variability of feature selection was not assessed here. SD is the variability of RF models using same set of features. In simulated data with higher-order relationship, no method was able to select X_{11}. The reason may be because that dichotomizing the response variable alleviates the relationship, q-value and VI are small for X_{11}. The features (X_{11} and X_{12}) in interaction were not selected by any of the algorithms either. It is still a difficult scenario for feature selection.

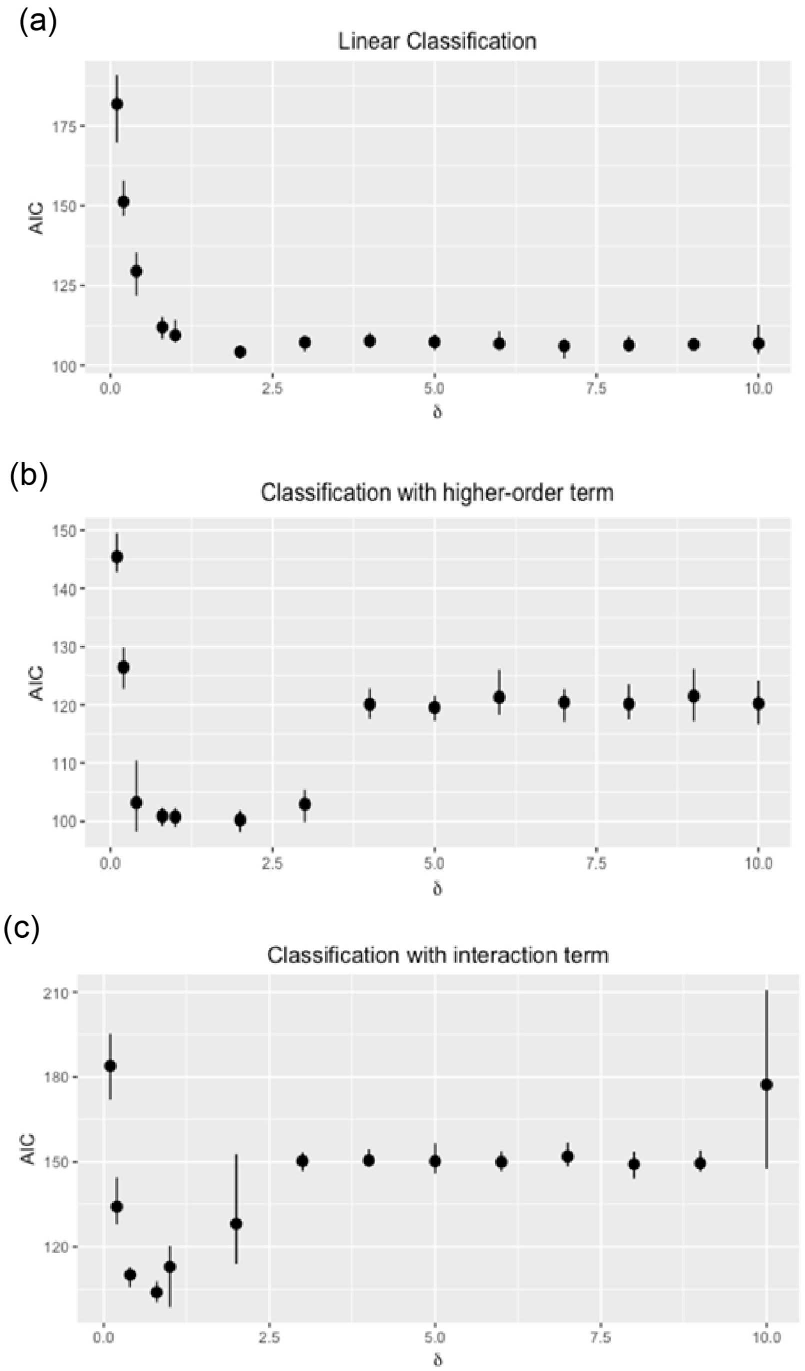

Figure 17 Classification Performance of Know-GRRF without Stability Test. To confirm that the optimization function can find the best parameter, we investigated Know-GRRF at the setting of $\boldsymbol{\delta} \in \{\mathbf{0.1}, \mathbf{0.2}, \mathbf{0.4}, \mathbf{0.8}, \mathbf{1}, \mathbf{2}, \dots, \mathbf{10}\}$ and examined how averaged AIC was changed with $\boldsymbol{\delta}$**.** The best value of $\boldsymbol{\delta}$ can be found by *optimization* function in R. Variability reflects both the selection variability and modeling variability.

Table 18 Methods Comparison in Linear Regression (Scenario 4)

Method	Features	MSE (SD)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	13.66 (0.241)	-	-
VSURF	6 7 8 9 10	8.02 (0.159)	50%	0%
GRRF ($\gamma = 0.9$)	3 4 6 7 8 10	10.34 (0.156)	60%	0%
Know-GRRF ($\delta = 0.69$, Stability=100%)	3 4 6 7 8 9 10	7.79 (0.154)	70%	0%
Known	1 to 10	8.08 (0.217)	-	-

Table 19 Methods Comparison in Regression with Higher-order Term (Scenario 5)

Method	Features	MSE (SD)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	33.30 (0.478)	-	-
VSURF	8 10 11	33.23 (0.429)	43%	0%
GRRF ($\gamma = 0.9$)	6 8 9 10 11 41 51	36.25 (0.328)	71%	2%
Know-GRRF ($\delta = 2.75$, Stability=100%)	8 10 11	33.16 (0.325)	43%	0%
Known	4 5 6 8 9 10 11	36.43 (0.472)	-	-

Table 20 Methods Comparison in Regression with Interaction (Scenario 6)

Method	Features	MSE (SD)	TPR (%)	FPR (%)
No feature selection	All 100 Xs	17.61 (0.208)	-	-
VSURF	6 8 9 10	18.93 (0.259)	50%	0%
GRRF ($\gamma = 0.9$)	6 8 9 10 51 93	19.38 (0.198)	50%	2%
Know-GRRF ($\delta = 0.98$, Stability=100%)	6 8 9 10	18.99 (0.169)	50%	0%
Known	4 5 6 8 9 10 11 12	18.58 (0.160)	-	-

Know-GRRF returned the same set of features from 10 runs for scenario 5. No stability test is needed. We did a stability test for scenario 4 and 6. We added features sequentially according to their selection probability and calculated averaged AIC for each set of features. The stability criteria were chosen at the smallest AIC. The final feature set was the set that lead to the best model performance. Figure 18 (b) shows the result of stability test in scenario 4 and 6.

For regression problems, performance of Know-GRRF and VSURF are quite comparable, but better than GRRF consistently. They selected fewer features and resulted in smaller MSE than GRRF. All three methods successfully identified the higher-order term (X_{11}) in Scenario 5. It confirms that the failure in classification problem may be caused by dichotomization of the response variable. However, features in interaction (X_{11}, X_{12}) were not selected by any of the algorithms. The reason may be that feature are selected or not based on the model performance, whereas the RF model itself may not be able to handle this interaction perfectly. Note even if we added X_{11} and X_{12} in the feature set to build a RF model, the MSE was not reduced.

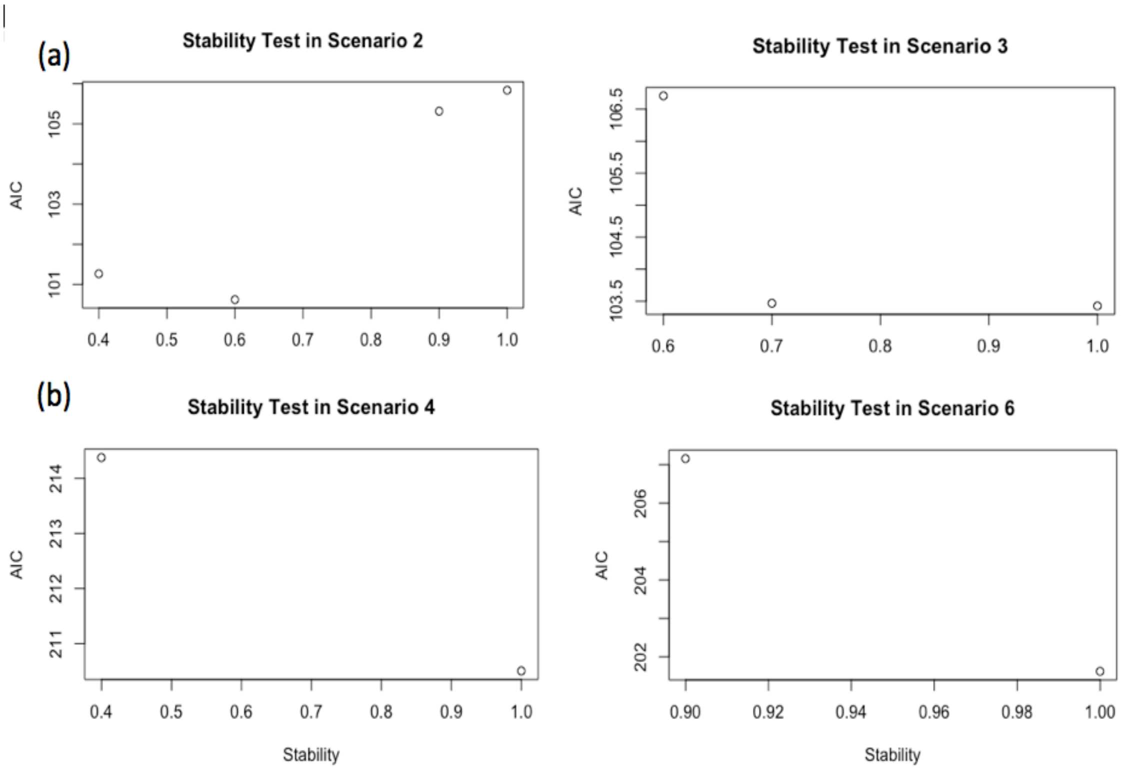

Figure 18 Stability Test on Some Scenarios. Stability criteria were set at the smallest AIC. Except scenario 2, AIC was the smallest when stability is 100%. Features that were consistently returned in all 10 runs were in the final feature set. In scenario 2, features that were returned 6 or more times were in the final feature set.

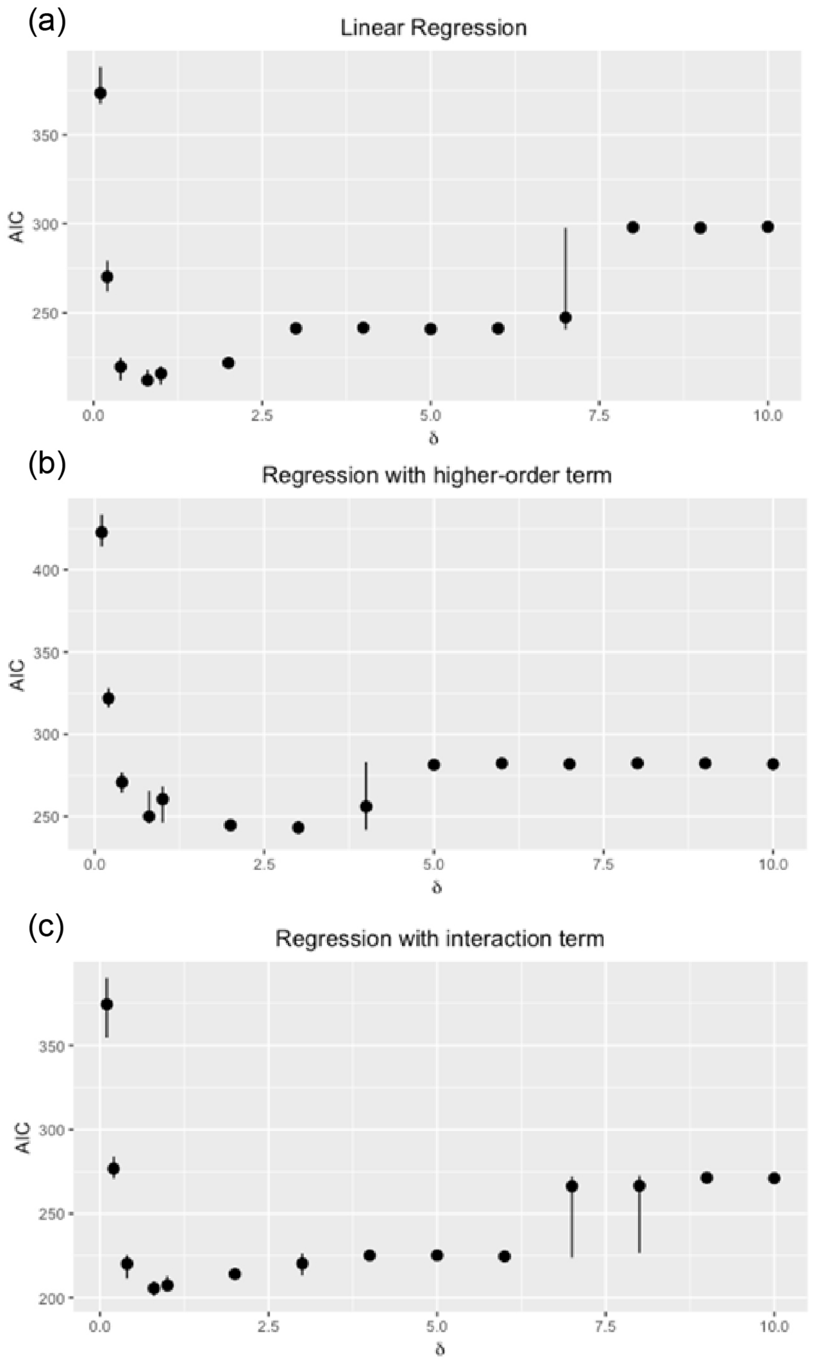

Figure 19 Regression Performance of Know-GRRF without Stability Test. To confirm that the optimization function can find the best parameter, we investigated Know-GRRF at the setting of $\boldsymbol{\delta} \in \{\mathbf{0.1, 0.2, 0.4, 0.8, 1, 2, \ldots, 10}\}$ and examined how averaged AIC was changed with $\boldsymbol{\delta}$. The best value of $\boldsymbol{\delta}$ can be found by *optimization* function in R. Variability reflects both the selection variability and modeling variability.

3.4 Discussion and Conclusion

The methods comparison on radiation dataset showed the importance of domain knowledge in feature selection. In general, a knowledge-based feature selection method has some significant advantages. Firstly, for high-dimensional data, e.g., microarray data, a large number of variables are simply noise and may cause overfitting. Incorporating domain knowledge can help to find out truly impactful variables and increase the model accuracy. Secondly, the domain knowledge can be used to make sense of the model and understand the mechanism or biological relationships. Thirdly, data-driven feature selection is subject to change with different samples or different data. Domain knowledge is consistent and would increase the reproducibility of feature sets and thus increase the generalizability of a predictive model.

The idea of using domain knowledge for feature selection has been studied widely. Helleputte and Dupont suggested transfer learning, which focus on extract knowledge from existing source and apply it to a different but related task[47]. Most of the knowledge are used for "pre-processing" as filters or "ad-hoc" explanation after modeling. For example, they typically rank genes according to their differential expression among phenotypes and pick the top-ranked genes[55]. Zhou developed gene-gene mutual information to prioritize candidate genes associated with a given disease[15]. Barzilay studied SVM and proposed to construct a kernel function according to some domain knowledge in texture recognition[56]. Ding utilized mutual information as a measure of relevance of genes and proposed a minimum redundancy, maximum relevance (MRMR) feature selection framework[57]. Genetic algorithms are widely applied to feature selection problems using evolutionary computation as domain knowledge[58]. These methods mostly incorporate domain knowledge as a filter. Here, we proposed an embedded method to wrap domain knowledge using RF.

We modified the framework of GRRF to incorporate cross-species correlation as the domain knowledge. We downsized the feature set to three genes and reduced human sample prediction error (MSE) from 9 to 4. Three genes selected by Know-GRRF include CDKN1A, CXXC5 and MYC. CDKN1A is a well-known gene to be transcriptionally regulated by p53 in response to stresses such as ionizing radiation[36] and reported by multiple groups as the top candidate gene for use in biodosimetry[40,41,59,60]. CXXC5 is related to DNA damage and was reported as a potential therapeutic target in primary acute myeloid leukemia cells in response to radiation therapy[61]. MYC is found as a novel biomarker for radiation. It is an important gene that regulates a range of cellular processes including biogenesis and protein synthesis[62]. The role of CXXC5 and MYC in response to radiation is worth studying in the future.

One concern of using cross-species correlation to help with feature selection is that the NHP predictive model performs worse than using data-driven algorithms only. A compromise is made here to gain better predictions for human samples. This is a common dilemma in cross-species transfer of learning. Park suggested that biodosimetry models based on interspecies-correlated genes had comparable dose-prediction accuracies on both species compared to using the full set of dose-responsive genes[36]. In their study, the better performance on NHP may because they found 52 genes that are highly correlated in expression pattern between species. Among a larger set of genes, it is more likely to include dose-responsive genes for NHP.

Our new method Know-GRRF improved the original GRRF in some aspects. Firstly, Know-GRRF modified the penalty coefficient to control the degree of regularization using an exponential function. In this way, the number of feature selected and the resulting model performance are more sensitive to the tuned parameter. The larger the regularization parameter is, the higher influence we put on the domain knowledge. Secondly, we proposed a more systematic way to tune the parameter by optimizing an objective function (AIC) of the resulting models. According to the figures showing the relationship between AIC and δ, a global minimum can often be found by convex optimization. Thirdly, we added a stability test to find consistently selected features. All RF and RF-based methods have large variation in both feature selection and predictive modeling. By running a stability test, we can remove some random, irrelevant features selected by GRRF and improve the reproducibility from run to run.

Furthermore, we generalize the use of Know-GRRF for feature selection in more general cases, even when no domain knowledge is available. We compared it to current widely used methods VSURF and varSelRF in six simulated scenarios. Performance of Know-GRRF is comparable to VSURF or varSelRF, if not better. But the computational time is much less, especially compared to VSURF, which adopted a stepwise selection procedure. We also tested using q-value instead of VI as the weight for each feature in classification. The results suggest the use of other statistics can be successfully applied to the Know-GRRF framework. Similarly, we may use other domain knowledge, such as mutual information and evolutionary weight, to regulate the penalty coefficient.

There are some limitations of Know-GRRF. In our simulation of scenarios with interaction terms, none of those tested methods were able to identify the features with interaction. Because Know-GRRF is not intended to solve XOR problems, we may need some other methods to tackle this challenge. If we have domain knowledge of interaction, for example, gene-gene interaction, we may be able to incorporate it to find interacting features. But this is beyond of the scope of this dissertation.

Secondly, it requires some computation for optimization and build more ensembles than GRRF. To search for the best value of the regularization term, we need to solve a convex optimization problem. We may set a larger tolerance to achieve faster computation, because there is a wide range of minima as solutions. Stability test requires more ensembles but can reduce variability from run to run. And meanwhile, we can assess the variation in the process of stability test. Know-GRRF is still based on RF, so variation is hard to be eliminated. However, we are able to remove random selected features by using a stability test.

Compared to VSURF and varSelRF, the use of Know-GRRF requires more steps than one encapsulated function. To obtain weights, we need extra analysis to get any intrinsic data characteristics such as q-value or VI. We also need optimization to determine the right scale of penalty. While allowing for more flexibility, the implementation of Know-GRRF requires more effort.

Moreover, though the prior knowledge can be helpful in improving the stability and model performance, using such information to guide the feature selection may meet certain limitations since biomarker discovery aims at finding new features rather than known ones[63]. We need to control the degree of regularization in Know-GRRF to emphasize or reduce the use of knowledge in feature selection.

For future study, we can simplify the implementation of Know-GRRF with built-in functions and options, so users can use this method easier. It could also be interesting to apply Know-GRRF to larger bioinformatics data. Relevant domain knowledge is crucial to improve the performance of Know-GRRF compared to other methods. For example, we incorporated evolutionary weight as the penalty term to select genes that can predict 5-year survival for breast cancer patients (GEO data in NCBI). The performance was not improved from using other methods, including varSelRF and univariate t-test controlling false discovery rate. To provide more useful insights for feature selection, we may need more specific domain knowledge regarding breast cancer survival or mutual information of gene and breast cancer. We can also combine the use of domain knowledge and intrinsic data characteristics. Because Know-GRRF heavily rely on the regularization coefficient, deriving the coefficients from several sources may improve the overall performance.

The stability of feature selection is of interest for future work. Because of the randomness of RF, all methods based on RF have a large variability. It is very likely that those methods, including VSURF, GRRF or varSelRF would return different sets of features from multiple runs. We used a stability criterion in Know-GRRF to reduce this variability. Variables that were selected by chance can be removed from the feature set. How to set this criterion is quite subjective now and may be worth studying in the future.

In a short conclusion, we proposed a framework of Know-GRRF to incorporate domain knowledge for feature selection. Instead of manually testing different combinations of genes, we can automatically select a set of genes that are mostly relevant from both data perspective and knowledge perspective. The method improved the GRRF in the way of constructing penalty coefficients and optimized the resulting model performance. Moreover, Know-GRRF can be generalized to do feature selection in general cases using intrinsic data characteristics for regularization. The performance is comparable to existing “wrapper” method VSURF and varSelRF, and better than “embedded” method GRRF.

CHAPTER 4

A NOVEL METHOD TO GENERATE PREDICTION INTERVAL USING RANDOM FOREST REGRESSION

4.1 Introduction

In solving the problems of model-based prediction, decision makers often require not only an accurate point prediction of certain variables but also the uncertainty associated with the prediction. Uncertainty of the model output can be estimated using prediction interval (PI). For the biodosimetor, it is desirable not only to provide a point estimate of radiation dose, but also to provide an interval for future predictions. Such interval indicates the dispersion of observations around the predicted value. It reflects the uncertainty of model output and can be used as an indicator of prediction precision.

Prediction interval usually consists of an upper and a lower limit between which the future value is expected to lie with a prescribed probability. The concept should be distinguished from another commonly used statistical term *confidence interval* (CI). CI applies to interval estimates for fixed but unknown parameters, while PI is an interval estimate for an unknown future value. PI deals with the accuracy of an estimate with regard to the actual observed value, rather than an estimate of population mean, thus it is more practical in real-world application[64]. Because PI account for not only the uncertainty in predicting the population mean, but also variability in data, PI is usually wider than CI.

Parametric methods for prediction interval involve an estimation of residual variance. In traditional linear regression model, PIs are given as a linear combination of prediction and standard deviation of residuals. In more complex models, such as the random forest, strong assumptions about errors are hard to be met and conditional distribution of prediction is hard to be determined. Researches on the error variance estimation, bootstrap techniques and other non-parametric methods demonstrate potential to construct PI for random forest. In this chapter, we first introduced some traditional methods to create PI and their limitations on this certain project. To deal with some specific problems, we then introduced a novel method we developed for constructing PI using random forest, called RFerr. We applied this new method to generate PI for the biodosimetor. In addition, we also tested the methodology in simulation study and compared it to existing methods on benchmark datasets.

4.1.1 Prediction Interval in Linear Regression

In classical linear regression problems, the task is to estimate a function $f(\mathbf{x};\theta)$ given data points D={$\mathbf{x}$, $\mathbf{y}$}, where θ is the true values of the parameters of the regression model. The least square estimate of θ is $\hat{\theta}$ and $\hat{y}_i = f\left(x_i;\hat{\theta}\right)$. The true value $y_i = \hat{y}_i + e_i, i = 1,2, \dots n$, where n is number of data points and e_i is the model error, which is assumed to be independently and identically distributed (iid) with the distribution N(0, σ^2). Assuming prediction is unbiased and e_i ~ N(0, σ^2) , most of the methods construct 100(1-α)% prediction interval as $(\hat{y} - Z_{\alpha/2}\sigma, \hat{y} + Z_{\alpha/2}\sigma)$. An unbiased estimate of σ^2 with n-p degree of freedom is given by the equation (4.1).

$$s^2 = \frac{\sum(y_i - \hat{y}_i)^2}{n-p} \#(4.1)$$

Noted that CI should be distinguished and it depends on the standard error of estimation and can be calculated as equation (4.2)

$$\hat{y} \pm Z_{\alpha/2}\frac{\sigma}{\sqrt{n}} \#(4.2)$$

Both PI and CI are symmetric about an unbiased estimator $\hat{y}$, and $Z_{\alpha/2}$ is the cumulative probability of standard normal distribution at the level of $\alpha/2$. This method is introduced in most of textbooks on regression[65].

This estimation above assumes that error has Gaussian distribution with mean zero and a constant variance in the output space. When the assumption is not true, estimation of σ^2 was modified to be

$$s_{y_i}^2 = s^2\left(1 + \frac{1}{n} + \frac{(x_i - \bar{x})^2}{(n-1)s_x^2}\right), \#(4.3)$$

where s^2 is the error variance, s_x^2 is the sample variance and $\bar{x}$ is the sample mean[66]. It can be seen from this equation that the error variance at y_i is always larger than s^2 and it depends on the distance between x_i and $\bar{x}$. The further the observation x_i is from the center of input space, the larger the error variance is. This approach can be generalized to multivariate linear regression. However, since the linear regression variance estimator approach requires many assumptions to use, which is not practical in most of the times, it is not ideal for non-linear or more complex models containing many equations.

4.1.2 Bootstrap Techniques

The traditional method needs to estimate error variance to make inference about prediction interval. It is considered as parametric approach. A non-parametric approach is more preferred when no assumption can be made about the sampling distribution. In this case, simulation and resampling based techniques, such as bootstrapping, or Monte Carlo techniques are used to construct PI.

The idea of bootstrap is to take samples from population and use the sample characteristics to infer population[67]. Nonparametric bootstrapping refers to resampling from an empirical distribution of the real data, while parametric bootstrapping refers to resampling from a known theoretical underlying distribution.

Nonparametric bootstrapping was applied to construct PI for standard linear regression model[68] and autoregressive model[69]. For linear regression, they generated prediction errors using bootstrap algorithm and obtained residuals by sampling with replacement from the empirical distribution of the residuals. PI limits were calculated as the combination of predicted value and upper or lower quantiles of the residuals. In a sense, the bootstrap "simulates" the distribution of prediction error by resampling[68]. The intervals were contrasted to other nonparametric procedures in some Monte Carlo experiments and they were found to be consistently liberal, especially for small sample size.

These results for regression had a similar application for time series in autoregressive model[69]. Standard forecast techniques usually assume that the error sequence of the series up to time t, $\{y_1, y_2 \ldots y_t\}$ is Gaussian. The conditional distribution of Y_{t+k} is Gaussian as well. Bootstrap prediction interval provides a nonparametric conditional distribution of Y_{t+k} and demonstrates its potential in a simulation study. Moreover, their preliminary results also suggested the use of bootstrap bias correction can improve coverage without increasing interval length. In addition, smoothing the empirical distribution of the residuals before resampling can potentially improve the interval estimate.

In a short summary, bootstrap prediction interval algorithm is a useful addition to the traditional measures of prediction uncertainty. These methods created prediction intervals using the measure of empirical measures of prediction errors calculated from resampling, so the property of intervals doesn't depend on the sampling distribution. However, such techniques have some limitation. Firstly, it is generally computational time consuming for resampling and modeling, especially for complex models. Secondly, nonparametric bootstrapping has its limitation with small sample size. More importantly, because the intervals were based on residuals, such techniques require strong assumption that the model is correctly specified otherwise the contribution of model bias would be neglected[68].

4.1.3 Error Variance of Random Forest

Random forest is a robust model for improving predictive accuracy and it is simple to understand and implement. However, it lacks the theoretical framework in which distributional statistics can be easily determined. In addition, the prediction from random forest model is not unbiased so that the prediction intervals are not symmetric about the prediction estimation. Therefore, methods are needed to make statistical inference for random forest model.

Inference about prediction usually requires an estimator of variability, such as error variance, or a sampling distribution of a pivotal quantity. Residuals in random forest regression are composed of bias and variance. Mendez and Lohr suggested estimating error variance based on mean squared error and bias correction using bootstrapping[70]. The estimator was proposed by subtracting an estimator of average bias from mean squared error. Averaged bias can be calculated from parametric bootstrapping or non-parametric bootstrapping. However, this method assumed error variance was equal. Coulston proposed a semiparametric way using Monte Carlo approach to approximate prediction uncertainty for random forest regression[71]. They use bootstrap resampling to parameterize a large number of RF models and assess prediction errors. Then they quantify prediction error based on error distribution and calculate interval as a linear combination of point estimate and its standard deviation. By using this method, they generate conservation (wider than necessary) prediction intervals.

4.1.4 Quantile Regression Forest

A nonparametric method often refers to a method that does not rely on assumptions about the probability distribution. Quantile regression is such a method, which takes the empirical distribution from training data and computes quantiles from the empirical distribution.

Quantile regression forest (QRF) was introduced by Meinshausen to create prediction interval for random forest output[72]. Similar to random forest, QRFs are ensembles of regression trees. QRF uses a non-parametric approach to construct conditional quantiles from the empirical distribution of response variable provided by RF model. The key difference of QRF and RF is that, for each terminal node, random forest keeps only the mean of samples that fall into the node and neglects all other information. In contrast, QRF keeps not only the mean, but also the whole empirical conditional distribution of each terminal node. Thus for a new sample, the point prediction is the same as prediction from RF, while the prediction interval can be computed from the empirical distribution provided by QRF.

There are several advantages to using QRF. Meinshausen proved the consistency of its estimates and showed in numerical examples that the algorithm is competitive in terms of predictive power. In addition, QRF shares many of the strengths of RF, including its flexible modeling, performance in high-dimensional data and its robustness to noise variables. Last but not the least, since QRF uses an empirical distribution of the response variable to calculate quantiles, it does not rely on any distributional assumptions about the predicted values or residuals. It has been proved to perform well in terms of prediction coverage, especially in situations where the conditional distributions are not Gaussian.

QRF has limited applicability in some cases. Firstly, because QRF uses the empirical distribution of the response variable from training data, the distribution may be restricted to the limited number of unique values. Especially when the response variable has few unique values in the training data, the empirical distribution may not be well estimated. In such cases, we have found the coverage probability of 95% PI generated by QRF were often close to 100% and the intervals were usually wide. Secondly, because QRF is based on RF, which constructs trees to accurately estimate the mean but not the complete conditional distribution, QRF may neglect some variables that are associated with the variability but not the mean. In the cases where there are variables that are strongly associated with the variability in the prediction but are not associated with the response itself, the empirical distribution of residuals might be useful for quantifying the uncertainty in the response and generating prediction intervals.

To deal with the discreteness of response variable in the radiation dataset, we need a more continuous variable for QRF model. Shrestha and Solomatine proposed a method for estimating prediction uncertainty using machine learning techniques[64]. The method is based on the idea that the residuals from model output are the best available quantitative indicator of the discrepancy between the model and the real world. These residuals can be modeled using the model inputs by mapping functions error=f(x; θ). The input space is partitioned into different clusters having similar residuals or residuals with similar distribution. The prediction interval $(\mathrm{PI}_e^L, \mathrm{PI}_e^U)$ for each cluster can be computed from empirical distribution of the corresponding residuals. Once the quantiles of residuals are obtained, the final prediction interval for jth observation can be computed by simply adding the model output as $(y_i - \mathrm{PI}_e^L, y_i + \mathrm{PI}_e^U)$. This method is referred as the local uncertainty estimation model (LUEM).

Inspired by the idea of LUEM, we built a model of prediction residuals to reflect the uncertainty of predictions. We propose a novel approach, RFerr, which combines the notion of the residual model and the use of empirical distribution of historical data to compute prediction intervals. RFerr integrates the QRF algorithm with a model of error to generate PI. Instead of utilizing the empirical conditional distribution of a response variable, RFerr constructs PI for non-parametric models by analyzing the empirical conditional distribution of prediction errors. In this approach we model the prediction residuals using RF to generate the distribution of predictions and calculate PIs for new samples. RFerr is intended to generate PI, particularly in settings where QRF does not perform well.

4.1.5 Description of the RFerr Algorithm

This proposed method can be applied to generate prediction interval for the random forest model we built for the biodosimetor. It is actually applicable to any kind of predictive model. Once the predictive model is built, the prediction residuals are calculated as the difference of predicted values and observed values. These residuals are then used to train a second-level error model by QRF. The final prediction interval of response can be obtained by shifting the conditional distribution of residuals by the original predicted value. The proposed algorithm to generate prediction interval is summarized as follows:

1. Build a predictive model M1 from the training data $\mathrm{L} = (\mathrm{X}, \mathrm{Y})$, with Y as a continuous response variable. $\mathrm{Y} = \mathrm{f}(\mathrm{X})$.

2. Obtain the prediction error defined as $\mathrm{E} = \widehat{\mathrm{Y}} - \mathrm{Y}$. Predicted $\widehat{\mathrm{Y}}$ can be cross-validation prediction or out-of-bag (OOB) prediction if available.

3. Given a new sample $\mathrm{sample_i} = \mathrm{x_{new}}$, use the predictive model M1 built in Step 1 to get the predicted value $\hat{\mathrm{y}}_\mathrm{i}$ for the test sample. $\hat{\mathrm{y}}_\mathrm{i} = \mathrm{f}(\mathrm{x_{new}})$.

4. Grow regression forest model M2 from training data, using prediction residuals E as response and X as predictors. $\mathrm{E} = \mathrm{g}(\mathrm{X})$. M2 has all properties of a regression forest.

5. Drop OOB samples $\mathrm{x}_{(\mathrm{oob},\theta)}$ down each regression tree $\mathrm{T}(\theta)$ in M2, as well as the new sample $\mathrm{x_{new}}$. Extract all observed values of $\mathrm{x}_{(\mathrm{oob},\theta)}$ in leaf nodes $\mathrm{l}(\theta, \mathrm{x_{new}})$ that $\mathrm{x_{new}}$ falls in. Similar to the weights defined in random forest, here we let the weight

vector $\mathrm{w_i(x, \theta)}$ be given by a positive constant if $\mathrm{x_i}$ is part of leaf $\mathrm{l(\theta, x_{new})}$ and is an OOB sample. Otherwise $\mathrm{w_i(x, \theta)} = 0$. The weights sum to one, and thus

$$\mathrm{w_i(x, \theta)} = \frac{1_{\{\mathrm{x_i} \in \mathrm{R}_{\mathrm{l(\theta, x_{new})}} \& \mathrm{x_i} \in \mathrm{x_{(oob,\theta)}}\}}}{\#\{\mathrm{j: x_j} \in \mathrm{R}_{\mathrm{l(\theta, x_{new})}} \& \mathrm{x_j} \in \mathrm{x_{(oob,\theta)}}\}} \#(4.4)$$

Let $\mathrm{w_i(x)}$ be the averaged weights over the collections of regression trees.

$$\mathrm{w_i(x)} = \mathrm{k}^{-1} \sum_{\mathrm{t}=1}^{\mathrm{k}} \mathrm{w_i(x, \theta_t)} \; \#(4.5)$$

6. As analogies of QRF, we approximate the conditional distribution function of error E given X = x using the weights defined in Step (5).

$$\hat{\mathrm{F}}(\mathrm{e} | \mathrm{X} = \mathrm{x}) = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{w_i(x)} 1_{\{\mathrm{e_i} \leq \mathrm{e}\}} \; \#(4.6)$$

Lastly, compute the conditional quantile of error from the distribution function given x_{new}, denoted as $Q_{el}(X = x_{new})$ and $Q_{eu}(X = x_{new})$. Prediction interval of the test sample $X_i = x_{new}$ is given as $[\hat{y}_i - Q_{eu}(X = x_{new}), \hat{y}_i - Q_{el}(X = x_{new})]$.

Random forest has been shown consistent for a simple model[73] and is established as universal consistent averaging rules with a number of theorems[74]. Furthermore, Meinshausen proved the consistency of QRF under less stringent assumptions[72]. The major difference of RFerr and QRF is the use of OOB samples instead of all samples in regression forest, which modified the weights but would not affect the proof of consistency. As we know, prediction error is the difference of the true response and the predicted response, so the distribution function of error follows the difference of the true response and the distribution of predicted response. Since both the point estimate and the quantile estimate of error are consistent estimates, the difference of the quantities are consistent estimates of response.

4.2 Methods

4.2.1 Datasets

To test RFerr, we used three types of data sets: (1) the real-world data sets from radiation biodosimetry research project, (2) existing benchmark data sets previously used for PI estimation by QRF and (3) simulated data.

The real-world data were NHP data in Citox lab obtained on day 2 after irradiation (Day 2, N=68). Because the predictive radiation dose was desired to be at a continuous scale, but radiation was delivered at certain discrete levels, the dataset is ideal to test RFerr algorithm.

Table 21 Characteristics of Datasets

Dataset	Source	# of variables	Sample size	Response variable	# of unique responses
Radiation	Real-world	19	68	Dose of radiation exposed	6
BostonHousing	mlbench	13	506	Median value of owner-occupied homes	229
Ozone (complete cases)	mlbench	12	203	Daily maximum one-hour-average ozone reading	35
BigMac2003	alr3	9	69	Minutes of labor to purchase a Big Mac	40
Fuel2001	alr3	5	51	Ratio of total gallons of gas sold and the approximate number of miles driven	51

To test the generalizability of this method, we compared it with QRF on some well-known benchmark datasets used in the paper[72]. The datasets used in the paper of

QRF were from R package *mlbench* and *alr3*. Dataset information is shown in Table 21.

Furthermore, we simulated a scenario where there exist some variables, which are predictive of the response variability instead of the mean. The hypothesis is that those variables are not predictive about the response mean hence they are not likely to be used in QRF model. In contrast, variables that are related to the variability are predictive about the prediction errors/dispersion, so they will be useful for RFerr.

Specifically, we simulated 100 datasets, each with 500 observations. To generate each observation, we simulated six independent standard normal random variables, five of which ($X_1 – X_5$) are predictive of response while the other one (X_6) is predictive of response variability.

$$\mathrm{Y} = \mathrm{b}_0 + \mathrm{b}_1\mathrm{x}_1 + \mathrm{b}_2\mathrm{x}_2 + \mathrm{b}_3\mathrm{x}_3 + \mathrm{b}_4\mathrm{x}_4 + \mathrm{b}_5\mathrm{x}_5 + \mathrm{ex}_6\#(4.7)$$

In equation (4.7), e is a standard normal variable. Coefficients b_0, …, b_5 were set to 0.2, 0.5, 0.6, 0.7, 0.8 and 0.9, respectively. We applied both RFerr and QRF on each simulated dataset.

4.2.2 Modeling

For the radiation dataset, regression, QRF and RFerr were used to create prediction interval of exposed dose. Simple linear regression was applied to predict dose estimate and prediction interval using parametric methods. RFerr was applied in two ways. Since we can choose any predictive model to be M1 in step (1), we built M1 using either regression or random forest. The second-level error model M2 in Step (4) was built using random forest. The two models were denoted as RFerr-reg and RFerr-RF, respectively. Finally, we built a QRF model on the same dataset. All methods used the same set of genes without feature selection.

For benchmark and simulation datasets, RFerr was compared to QRF on all datasets. RFerr was applied with both M1 and M2 built in random forest. All modeling was done in RStudio. M1 in RFerr was implemented using *randomForest* R package under default setting, while the nodesize was set to 10 in M2. All available variables were used for both M1 and M2 for simplicity. The idea to restrict the nodesize to 10 in M2 is to grow a relatively small tree so that sufficient samples can reside in the same nodes for a smoother empirical distribution. QRF was implemented using a R package called *quantregForest* under default setting. The nodesize in quantregForest was defaulted to 10 as well.

In addition, in benchmark datasets, we tested different versions of RFerr using different sets of samples to generate the empirical distribution. Specifically, we tested using OOB samples, in-bag samples or all samples in Step #5 to create prediction interval. They are labeled as RFerr(out), RFerr(in) and RFerr(all) respectively.

4.2.3 Evaluation

Four models were built to create prediction interval on radiation dataset. Leave-one-out cross-validation was used to evaluate the model performances. For benchmark and simulation datasets, we compared the performance of RFerr to QRF under 5-fold cross-validation. The performance metrics we are interested in the coverage under cross-validation and the length of prediction interval. Ideally the (1-α)% PI should have the (1-α)% observations staying within its corresponding intervals. So the miscoverage rate should be close to α%. In addition, a precise PI indicates more certainty about the future prediction, thus a PI with shorter length is more preferred. Miscoverage rate is defined as the percentage of observations whose actual response falls out of the lower and upper PI limits. Interval length is defined as the average difference of the upper PI limits and lower PI limits for all samples in one dataset.

$$\text{Miscoverage rate } = \frac{\mathrm{I}(\mathrm{y} \notin [\mathrm{PI}_{\propto},\ \mathrm{PI}_{1-\propto}])}{\mathrm{N}(\mathrm{y})} \#(4.8)$$

$$\text{Interval length } = \frac{\sum(\mathrm{PI}_{1-\propto}-\mathrm{PI}_{\propto})}{\mathrm{N}(\mathrm{y})} \#(4.9)$$

Because both RFerr and QRF are based on RF, which cause some variability, the modeling processes were repeated 100 times on benchmark datasets to obtain the mean and standard error of the miscoverage rate and precision of prediction interval.

4.3 Results

4.3.1 Radiation Data

A non-constant residual variance from RF predictive models across delivered dose levels was observed in Figure 20. Lower dose samples are overestimated and higher dose samples are underestimated. Variance is obviously larger at middle-to-higher dose range than lower dose range.

To construct a prediction interval for dose estimate, we built four models on the radiation dataset under LOOCV and compared their performance.

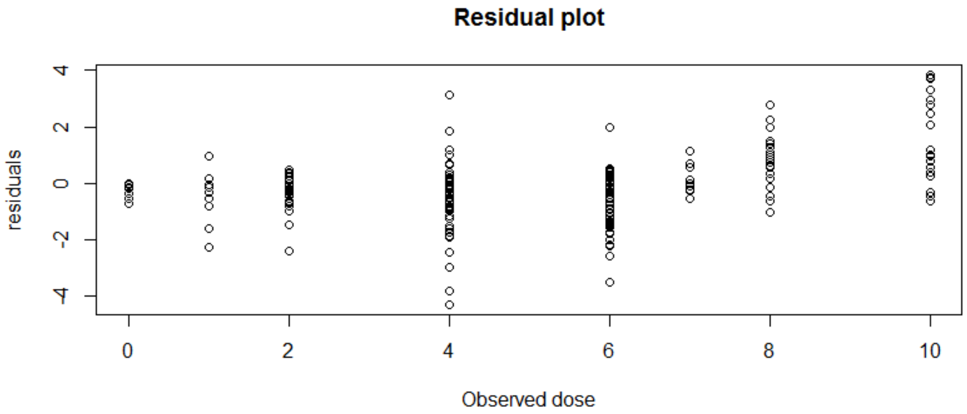

Figure 20 Prediction Errors across Dose. Variance of residuals is not constant across observed dose. Variance is larger for observed dose greater than 4 Gy. Samples of lower dose are all overestimated and samples of higher dose are all underestimated.

Table 22 Miscoverage Rate and Length of PI on Radiation Dataset

Models	Miscoverage rate (%)	Interval length
Regression	11.7 (5.2, 21.9)	3.3
RFerr-reg	4.4 (0.9, 12.3)	3.8
RFerr-RF	4.4 (0.9, 12.3)	3.2
QRF	0 (0, 5.2)	5.4

As shown in Table 22, PI calculated from standard regression model using a parametric method had the highest miscoverage rate. Numbers in parentheses of miscoverage rate is the 95% confidence interval calculated under binomial assumption. A further examination of the residuals revealed that the simple linear regression model may be misspecified since assumptions about residuals were not met (Figure 20). We used RFerr to construct PI with the same regression model as the predictive model. Miscoverage rate of RFerr-reg was 4.4% and the confidence interval crossed 5%. It indicated that PI calculated from RFerr doesn’t require strong assumptions about residuals. McNemar’s test suggested the miscoverage rates by regression method and

RFerr-reg are different (p=0.07). Paired t-test showed the interval length generated by parametric method and RFerr differed significantly (t=-5.12, p<0.001), when we used regression model as the predictive model. RFerr-reg has a correct coverage but the interval is precise. Paired t-test showed the interval length generated by QRF and RFerr also differed significantly (t=11.41, p<0.001), when we used RF as the predictive model. Overall, RFerr-RF has the most precise interval with an accurate coverage rate.

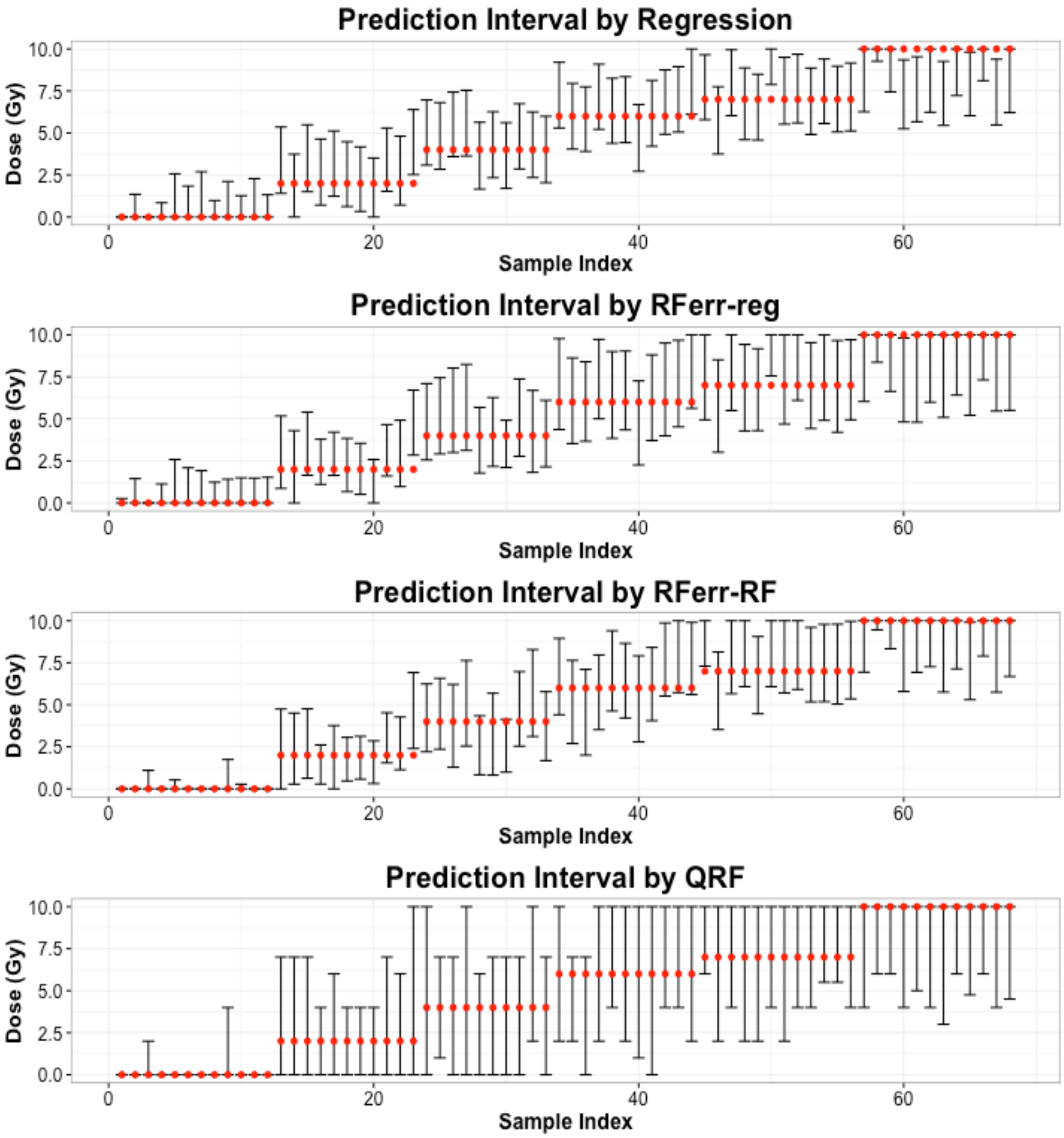

Figure 21 Prediction Interval on Radiation Dataset. (a) presents PIs created by traditional regression methods; (b) present PIs created by RFerr with regression model as the primary model; (c) presents PIs created by RFerr with random forest model as the primary model; (d) presents PIs created by Quantile Regression Forest. For all plots, X-axis represents 68 samples and Y-axis indicates dose. Red dots represent the actual dose delivered. Vertical lines indicate the 95% prediction interval for each sample.

We also used RFerr with a RF model as the predictive model. Miscoverage rate of RFerr-RF was still 4.4% but the average length of PIs decreased to 3.2. Although RFerr was able to construct legitimate PI for a mis-specified model, with a better predictive model, RFerr could generate more precise PI. Lastly, the average length of PIs generated by QRF was 5.4 while the miscoverage rate was 0%. Figure 21 displays the 95% PIs created for each of the samples using each of the four models. As seen in Figure 21 (d), the PI bounds generated by QRF were the same for a group of samples. QRF may behave too conservative on this dataset. To conclude, RFerr generated shorter and more differentiated PIs than QRF, and its miscoverage rates were closer to the prescribed rate than the regression model and QRF.

4.3.2 Benchmark Data

Table 23 Miscoverage Rate and Length of PI on Benchmark Datasets

Dataset	Methods	Mean (SE)	
		Miscoverage rate (%)	Interval length
BostonHousing	QRF	2.1 (0.43)	16.35 (0.22)
	RFerr	4.8 (0.73)	11.45 (0.14)
Ozone	QRF	3.7 (0.83)	18.10 (0.25)
	RFerr	4.7 (1.01)	16.83 (0.22)
BigMac2003	QRF	2.7 (1.52)	62.75 (3.42)
	RFerr	5.0 (1.92)	67.63 (3.41)
Fuel2001	QRF	5.0 (1.79)	492.99 (26.45)
	RFerr	5.0 (2.69)	437.68 (31.89)

The performance comparisons on benchmark datasets are shown in Table 23. Because there was variability in both RFerr and QRF modeling, we repeated the modeling process 100 times. Mean is the average performance from 100 repetitions. SE is the standard deviation of the mean estimates from 100 repetitions. The results showed

RFerr was able to construct an appropriate prediction interval with a more accurate coverage rate and a relatively shorter length than QRF. The 95% PIs created by QRF had miscoverage rates much lower than 5%, except for Fuel2001 dataset. In contrast, the 95% prediction intervals created by RFerr had miscoverage rates all very close to 5%. The interval lengths were all shorter than those by QRF, except for BigMac2003 dataset.

Figure 22 showed RFerr(in) and RFerr(all) both had miscoverage rates higher than 5%, although interval lengths were shorter.

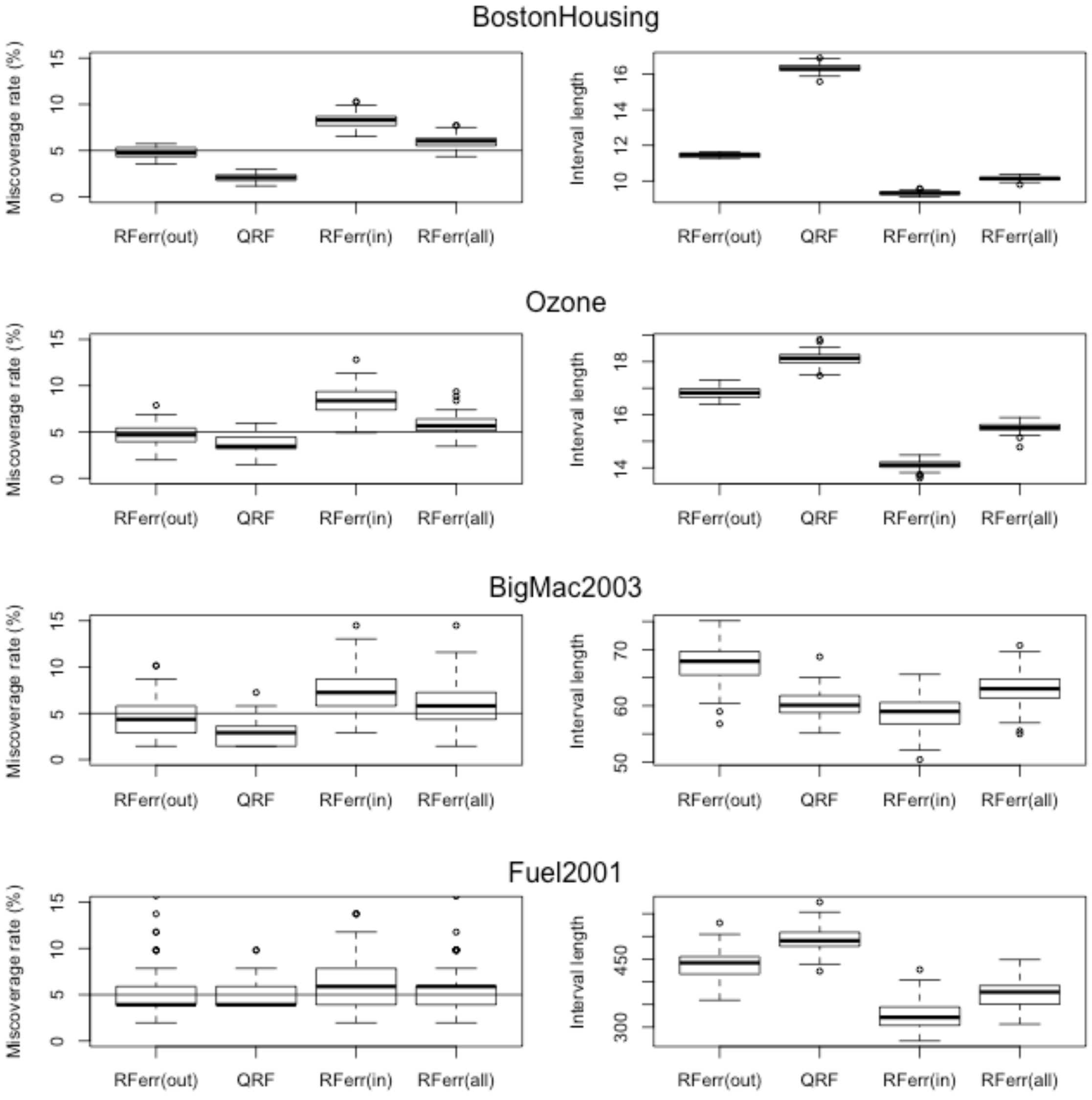

Figure 22 Miscoverage Rate and PI Length on Benchmark Datasets. Boxplots were generated from 100 repetitions. The left-hand side panel displays miscoverage rates and the horizontal line represents the prescribed miscoverage rate (5%). The right-hand side panel displays the lengths of prediction intervals.

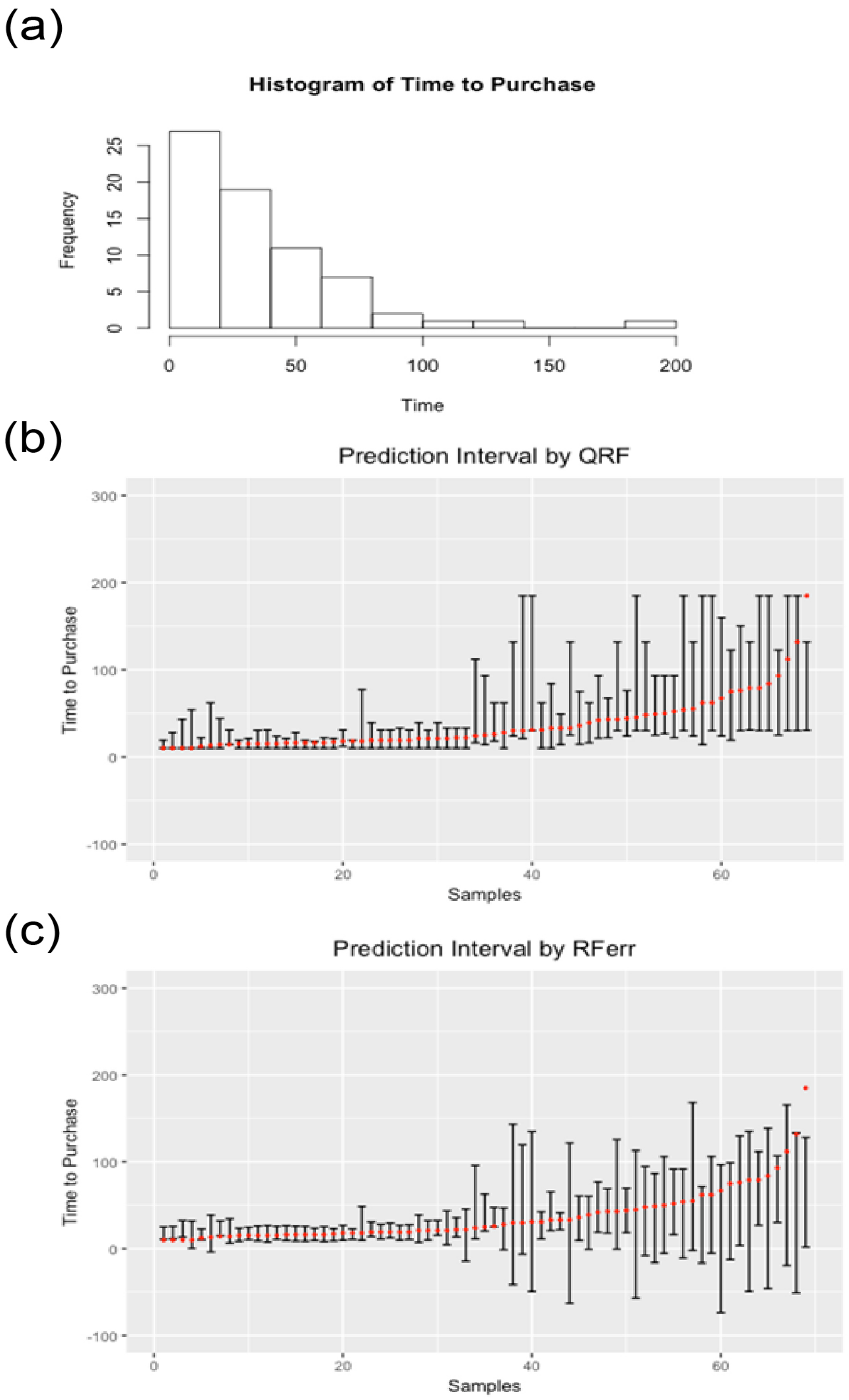

Figure 23 Comparison of QRF and RFerr on BigMac. (a) Distribution of response variable of BigMac dataset is highly skewed. (b) and (c) are PIs generated by QRF and RFerr respectively. Prediction variability is much higher for samples at the right tail. Some lower bound of PIs in (c) fall below 0 so we need to adjust them.

4.3.3 Simulation Results

The results from simulation are shown in Table 24 and Figure 24. Mean and standard error are calculated from 100 simulated datasets. Again, RFerr has a miscoverage rate closer to 5% and the interval length is shorter than QRF.

Table 24 Miscoverage Rate and Length of PI on 100 Simulated Datasets

Methods	Mean (SE)	
	Miscoverage rate (%)	Interval length
QRF	3.56 (0.67)	5.94 (0.22)
RFerr	4.64 (0.73)	4.52 (0.21)

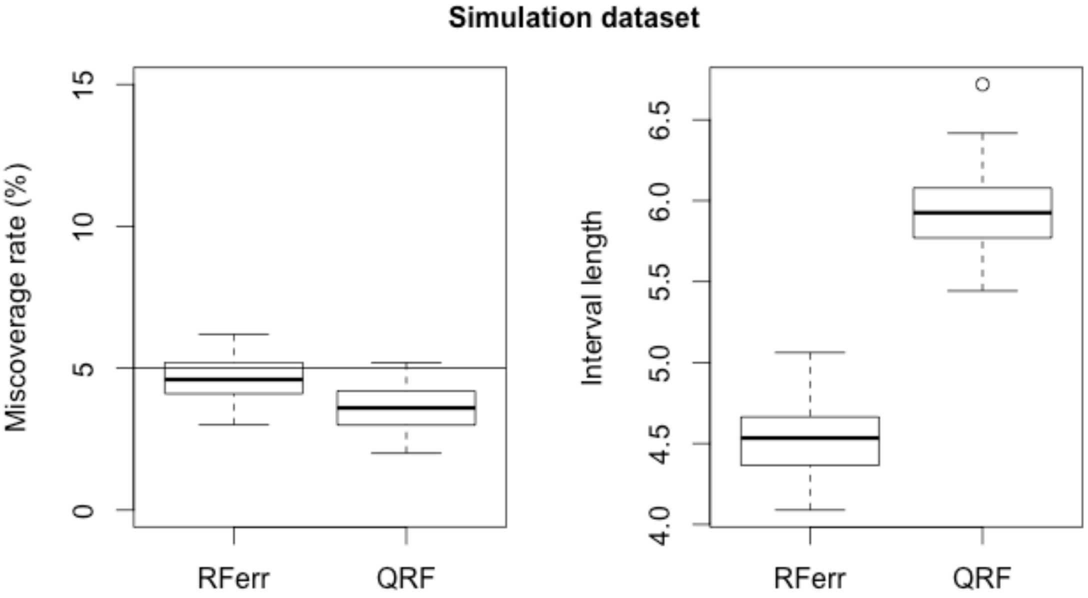

Figure 24 Miscoverage Rate and PI Length on Simulated Datasets. (a) compares the miscoverage rate of 95% prediction interval. The horizontal line represents the prescribed miscoverage rate (5%). Median of the miscoverage rate resulted from RFerr over 100 simulated datasets is closer to 5% than from QRF. (b) compares the length of 95% prediction interval. RFerr overall produces more precise prediction intervals than QRF.

4.4 Discussion and Conclusion

RFerr can create prediction interval for non-linear, non-additive, high-dimensional data. Unlike parametric methods, it doesn't require strong assumptions about the sampling distribution of error. Instead, it works with complex models such as random forest and utilizes the empirical distribution of residuals. RFerr showed a more accurate coverage rate and more precise prediction interval, compared to quantile regression forest, which similarly is based on random forest.

As seen from the results, the prediction intervals generated by QRF are usually wide and sometimes identical for many samples. One possible reason may be the discreteness of the response variable. For example in the radiation dataset, dose is a continuous response variable, but only certain dose levels were administered due to the experimental design. Because QRF uses the empirical distribution of the training data, which consist of few unique discrete values, the quantile statistics from the empirical distribution are bounded by certain values. More importantly, for all the datasets (except Fuel2001), including the radiation, benchmark and simulated datasets, QRF generates PIs with miscoverage rates much lower than 5%. This suggests QRF can be too conservative. On the other hand, RFerr generated more precise prediction intervals on all datasets except for BigMac2003. One possible reason is that the distribution of its response variable in BigMac2003 is highly skewed (Figure 23 a). Figure 23 (b) and (c) compared the PI generated by QRF and RFerr. Note that for samples at the right tail, PIs were generally much wider by both methods because the sample size was limited at the tail. Lower bounds of PI for some samples by RFerr even fell below 0. After adjusting those lower bounds, the length of PI by RFerr became 57.82, which was shorter than QRF.

RFerr provides an accurate PI for all datasets with a miscoverage rate right about 5%. Not only for the discreteness problem, RFerr outperforms QRF when predictors are informative about the variability but not the mean of response variable. Both QRF and

RFerr are based on RF predictive model, however, predicting prediction error is a better idea than predicting response when we need to capture the dispersion of future prediction. More importantly, RFerr is not restricted to create PI for random forest model. Since it models the residuals from any predictive model, such as regression model, RFerr is widely applicable.

Moreover, PI created by RFerr is robust to model misspecification. Unlike QRF, RFerr is more widely applicable and can generate PI for any predictive model other than RF. For radiation dataset, we used either linear regression or RF as the predictive model for dose. Although linear regression model is not a great fit, the coverage of PI by RFerr is still accurate. The length of PI is a little wider than using a better fit RF model. The reason may because that the OOB sample prediction errors simultaneously include all causes of errors in the model predictions, including random variations in the data collection, parameter estimation errors, and errors due to incorrect model specification. The empirical approach does not assume that the prediction is unbiased. Instead, it is based on the empirical analysis of past prediction errors that would have been made by the chosen model.

Empirical prediction interval methods usually compute the prediction errors as residuals and then apply non-parametric methods, such as Chebyshev's inequality and kernel density estimators[75], and semi-parametric methods, such as quantile regression[76], to construct prediction intervals. By using residuals rather than future prediction errors, these methods usually result in shorter PIs with higher miscoverage rates[77]. We illustrated this point by using OOB, in-bag or all samples in RFerr algorithm. It was shown that RFerr(out) generated 95% PIs with miscoverage rates the closest to the desired percentile.

It is because that the OOB sample provides a better representative sample of future observations, its empirical distribution is a better approximation of future prediction errors. Therefore, in the application of RFerr, OOB samples should be used to obtain the prediction interval.

In addition, RFerr can handle multiple imputation and interpolation naturally. In predictive modeling, it is not unusual to deal with missing data. Multiple imputation is a technique to replace each missing value with a set of plausible values that represent the uncertainty about the true value[78]. Usually, each imputed value will be analyzed independently and a pooled estimate will be generated for the sample with missing values. In RFerr, multiple plausible values would be imputed for a missing value and each imputed copy is analyzed separately. For each imputed copy, the empirical conditional distribution of response variable is found by shifting the empirical distribution of residuals by the point estimate of response, and the final distribution of all imputed copies is a mixture of individual empirical response distributions. Prediction interval can be then obtained by taking quantiles from the merged distribution. The prediction interval calculated by RFerr incorporates the variation in imputation for missing data, above and beyond the variation in estimation and data.

Similarly, RFerr works for interpolation, when there are multiple models. When interpolating between two models, two point estimates can be obtained respectively. Two error models are then built and so as the conditional empirical distributions of residuals. After shifting the two residual distributions by two point estimates respectively, we can merge them by a weighted sampling with replacement. Finally, prediction interval can be computed using the quantiles of the merged distribution.

RFerr has some limitations. It requires more computation time than QRF, since it requires an extra model of error to build. Time can be saved though by limiting the tree size or terminal node size for the error model. A sufficient number of terminal node size can guarantee a better coverage of PI. Moreover, RFerr is more complex. Instead of creating prediction interval for the response variable directly, we build an error model and create prediction interval for the residuals. This additional step can be beneficial when the response variable is discrete or there exists some predictors that are predictive of variability. Otherwise, the traditional QRF would work great to generate PI. Lastly, QRFerr can cause some out of bound issue when creating PI. For example, in the radiation dataset, a legitimate value for dose estimate would be greater than 0. And in the BigMac dataset, time to purchase a Mac should also be greater than 0. However, because RFerr generates PI as the difference of a point estimate and a residual estimate, it possibly falls out of the range. In this case, users need to be careful to define the prediction range and adjust the prediction interval accordingly.

To conclude, we proposed a novel method RFerr to construct prediction interval using an error model in random forest. By building a second-level model of residuals, this method can be used with any type of predictive model to create corresponding prediction interval. It is especially useful when the response variable is continuous but restricted to a few unique values, or where there exist variables that are predictive of prediction variability. Because RFerr is based on empirical methods, it doesn’t require assumptions about the residual distribution, nor it requires the point estimate to be unbiased.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

This dissertation proposed two novel methods for feature selection and predictive modeling. RF has its built-in mechanism for feature selection and is well known for its excellent performance in prediction. Therefore, we developed the new methodologies based on RF.

The new method Know-GRRF is an improvement from GRRF, which can incorporate weights to guide the regularization of random forest. We can control the degree of regularization better by defining an exponential function of the regularization parameter. Moreover, model performance is considered in search of the regularization parameter so that model performance is optimized. In addition, the stability test would select features, which are consistently returned from multiple runs so the feature set selected by Know-GRRF is more stable. Know-GRRF was used to incorporate domain knowledge for regularization in biomarker discovery. We also showed its generalizability using intrinsic data characteristics in simulated datasets.

The new method RFerr can generate prediction interval in predictive modeling for any kind of predictive models, and is especially useful for complex models. It is a non-parametric method so it doesn’t require for assumptions in traditional linear regression. In RFerr, we use the idea of QRF but build a model of error instead to examine the empirical distribution of residuals. By using the OOB samples, we can able to get a good representation of future prediction error. We can then use predicted residuals to estimate the dispersion of future prediction.

There are several applications of these methodologies in this dissertation. We firstly applied random forest to a gene expression dataset for predictive modeling. By building day-specific models and nested models, the predictive algorithm has better performance with regard to accuracy. The model shows the strength of RF in dealing with challenges in bioinformatics data, including interaction, non-linearity and heterogeneity. However, the model built with NHP data doesn't predict well for human samples because of the inherent difference in two species. Genes that are dose-responsive in NHP may not respond to radiation in human. Therefore, we use cross-species correlation to guide the selection of biomarkers in NHP model. Specifically, we applied Know-GRRF for cross-species prediction, using the cross-species correlation as domain knowledge. The method shows significant improvement in human sample prediction compared to other methods, including VSURF and GRRF. We also applied Know-GRRF on simulated datasets varying in complexity, using intrinsic data characteristics. The feature sets selected by Know-GRRF generally have higher TPR and lower FPR. Models built with selected features by Know-GRRF also have smaller error rate or MSE.

Moreover, for the biodosimetry, decision makers not only want a point estimate, but also need an interval estimate of the predicted dose, which can reflect the uncertainty of future prediction. However, RF model doesn't provide a prediction interval along with the dose estimate. Therefore, we applied RFerr to generate prediction interval. The interval generated by RFerr has a more accurate coverage and a more precise length than QRF using RF as the predictive model. We also compared RFerr and QRF on some benchmark datasets and simulated datasets. They all suggested a better performance by RFerr.

5.2 Future Work

In predictive modeling, we built day-specific models and nested RF models to predict radiation dose from gene expression. By separating samples into smaller groups, we can remove some noise and target on more homogenous group. And by selecting features for each dose range, we can model a more linear relationship in subset of data. In future work, we can generalize this methodology to deal with non-linearity and heterogeneity. We can use either unsupervised clustering or supervised prediction to separate samples. Testing this method on various datasets of different complexity could suggest when this method would be the most helpful.

Know-GRRF was showed to have good performance using either domain knowledge or intrinsic data characteristics for guided regularization. In future study, we could use both two sources to guide the selection of biomarkers simultaneously. Two parameters can be used to control the relative importance of the domain knowledge in the dataset. If the domain knowledge in not relevant, the algorithm would put more weight on the data characteristics.

Secondly, we can decrease the computation time of Know-GRRF by improving the optimization. We can increase the step size or the tolerance to expedite the convergence. In addition, we can investigate several data points to get an idea of the range of the tuned parameter. We can also use some domain knowledge to set the upper or lower bound of the search space. When domain knowledge is very important, the regularization parameter usually is larger.

In future work, we can also apply Know-GRRF on larger bioinformatics data. The application of Know-GRRF in this dissertation is on an experimental dataset with limited sample size. To test its generalizability, we can apply it on available bioinformatics data from database, using domain knowledge such as mutual information or evolutionary weight for biomarker discovery.

Lastly, to improve RFerr in future study, we can save some computation time by using RF as the predictive model. OOB error from RF can be used as the response variable in the error model directly. Otherwise, LOOCV would be needed to calculate residuals. Moreover, setting a larger nodesize in the error model can decrease the computation time significantly. Because residuals are generally more similar than the responses are, we usually would result in a very deep tree for the error model. Nodesize was defaulted to 10 in the error model of RFerr as suggested by QRF, but it may be worth studying the influence of nodesize on the final performance as well as the computation time.

REFERENCES

1. Moore, J. H., Asselbergs, F. W. & Williams, S. M. Bioinformatics challenges for genome-wide association studies. *Bioinformatics* **26,** 445–455 (2010).

2. Moore, J. H. & Williams, S. M. New strategies for identifying gene-gene interactions in hypertension. *Ann. Med.* **34,** 88–95 (2002).

3. Breiman, L. Random forests. *Mach. Learn.* **45,** 5–32 (2001).

4. Caruana, R. & Niculescu-Mizil, A. An empirical comparison of supervised learning algorithms. in *Proceedings of the 23rd international conference on Machine learning* 161–168 (ACM, 2006).

5. Hastie, T. *et al. The elements of statistical learning*. **2,** (Springer, 2009).

6. Cook, N. R., Zee, R. Y. & Ridker, P. M. Tree and spline based association analysis of gene–gene interaction models for ischemic stroke. *Stat. Med.* **23,** 1439–1453 (2004).

7. Lunetta, K. L., Hayward, L. B., Segal, J. & Van Eerdewegh, P. Screening large-scale association study data: exploiting interactions using random forests. *BMC Genet.* **5,** 32 (2004).

8. Liaw, A. & Wiener, M. Classification and Regression by randomForest. *R News* **2,** 18–22 (2002).

9. Saeys, Y., Inza, I. & Larrañaga, P. A review of feature selection techniques in bioinformatics. *bioinformatics* **23,** 2507–2517 (2007).

10. Genuer, R., Poggi, J.-M. & Tuleau-Malot, C. Variable selection using random forests. *Pattern Recognit. Lett.* **31,** 2225–2236 (2010).

11. Díaz-Uriarte, R. & De Andres, S. A. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics* **7,** 3 (2006).

12. Strobl, C., Boulesteix, A.-L., Zeileis, A. & Hothorn, T. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics* **8,** 25 (2007).

13. Domingos, P. Metacost: A general method for making classifiers cost-sensitive. in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining* 155–164 (ACM, 1999).

14. Jin, G. *et al.* The knowledge-integrated network biomarkers discovery for major adverse cardiac events. *J. Proteome Res.* **7,** 4013–4021 (2008).

15. Zhou, H. & Skolnick, J. A knowledge-based approach for predicting gene-disease associations. *Bioinformatics* btw358 (2016).

16. Deng, H. & Runger, G. Gene selection with guided regularized random forest. *Pattern Recognit.* **46,** 3483–3489 (2013).

17. Meinshausen, N. Quantile regression forests. *J. Mach. Learn. Res.* **7,** 983–999 (2006).

18. Waselenko, J. K. *et al.* Medical management of the acute radiation syndrome: recommendations of the Strategic National Stockpile Radiation Working Group. *Ann. Intern. Med.* **140,** 1037–1051 (2004).

19. Pellmar, T. C. & Rockwell, S. Priority list of research areas for radiological nuclear threat countermeasures. *Radiat. Res.* **163,** 115–123 (2005).

20. Cologne, J. B. & Preston, D. L. Longevity of atomic-bomb survivors. *The Lancet* **356,** 303–307 (2000).

21. Gowns, R. E., Holloway, E. C., Berger, M. E. & Ricks, R. C. Early dose assessment following severe radiation accidents. *Health Phys.* **72,** 513–518 (1997).

22. Alexander, G. A. *et al.* BiodosEPR-2006 Meeting: Acute dosimetry consensus committee recommendations on biodosimetry applications in events involving uses of radiation by terrorists and radiation accidents. *Radiat. Meas.* **42,** 972–996 (2007).

23. Weinstock, D. M. *et al.* Radiologic and nuclear events: contingency planning for hematologists/oncologists. *Blood* **111,** 5440–5445 (2008).

24. Garty, G. *et al.* The RABIT: a rapid automated biodosimetry tool for radiological triage. *Health Phys.* **98,** 209 (2010).

25. Ivey, R. G., Subramanian, O., Lorentzen, T. D. & Paulovich, A. G. Antibody-based screen for ionizing radiation-dependent changes in the mammalian proteome for use in biodosimetry. *Radiat. Res.* **171,** 549–561 (2009).

26. Sharma, M. & Moulder, J. E. The urine proteome as a radiation biodosimeter. in *Radiation Proteomics* 87–100 (Springer, 2013).

27. Laiakis, E. C. *et al.* Development of a metabolomic radiation signature in urine from patients undergoing total body irradiation. *Radiat. Res.* **181,** 350–361 (2014).

28. Gruel, G. *et al.* Broad modulation of gene expression in CD4+ lymphocyte subpopulations in response to low doses of ionizing radiation. *Radiat. Res.* **170,** 335–344 (2008).

29. Mori, M., Benotmane, M. A., Tirone, I., Hooghe-Peters, E. L. & Desaintes, C. Transcriptional response to ionizing radiation in lymphocyte subsets. *Cell. Mol. Life Sci.* **62,** 1489–1501 (2005).

30. Paul, S. & Amundson, S. A. Gene expression signatures of radiation exposure in peripheral white blood cells of smokers and non-smokers. *Int. J. Radiat. Biol.* **87,** 791–801 (2011).

31. Paul, S. *et al.* Gene expression response of mice after a single dose of 137CS as an internal emitter. *Radiat. Res.* **182,** 380–389 (2014).

32. Dressman, H. K. *et al.* Gene expression signatures that predict radiation exposure in mice and humans. *PLoS Med* **4,** e106 (2007).

33. Meadows, S. K. *et al.* Gene expression signatures of radiation response are specific, durable and accurate in mice and humans. *PloS One* **3,** e1912 (2008).

34. Ossetrova, N. I. & Blakely, W. F. Multiple blood-proteins approach for early-response exposure assessment using an in vivo murine radiation model. *Int. J. Radiat. Biol.* **85,** 837–850 (2009).

35. Tucker, J. D. *et al.* Gene expression-based dosimetry by dose and time in mice following acute radiation exposure. (2013).

36. Park, J. G. *et al.* Developing Human Radiation Biodosimetry Models: Testing Cross-Species Conversion Approaches Using an Ex Vivo Model System. *Radiat. Res.* (2017).

37. MacQueen, J. & others. Some methods for classification and analysis of multivariate observations. in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* **1,** 281–297 (Oakland, CA, USA., 1967).

38. Wilkins, R. C. *et al.* Interlaboratory comparison of the dicentric chromosome assay for radiation biodosimetry in mass casualty events. *Radiat. Res.* **169,** 551–560 (2008).

39. Paul, S. & Amundson, S. A. Development of gene expression signatures for practical radiation biodosimetry. *Int. J. Radiat. Oncol. Biol. Phys.* **71,** 1236–1244 (2008).

40. Marchetti, F., Coleman, M. A., Jones, I. M. & Wyrobek, A. J. Candidate protein biodosimeters of human exposure to ionizing radiation. *Int. J. Radiat. Biol.* **82,** 605–639 (2006).

41. Paul, S. *et al.* Prediction of in vivo radiation dose status in radiotherapy patients using ex vivo and in vivo gene expression signatures. *Radiat. Res.* **175,** 257–265 (2011).

42. de Lemos Pinto, M. M. P., Santos, N. F. G. & Amaral, A. Current status of biodosimetry based on standard cytogenetic methods. *Radiat. Environ. Biophys.* **49,** 567–581 (2010).

43. Flegal, F. N., Devantier, Y., McNamee, J. P. & Wilkins, R. C. Quickscan dicentric chromosome analysis for radiation biodosimetry. *Health Phys.* **98,** 276–281 (2010).

44. Rodrigues, M. A., Beaton-Green, L. A. & Wilkins, R. C. Validation of the cytokinesis-block micronucleus assay using imaging flow cytometry for high throughput radiation biodosimetry. *Health Phys.* **110,** 29–36 (2016).

45. Vaurijoux, A. *et al.* Detection of partial-body exposure to ionizing radiation by the automatic detection of dicentrics. *Radiat. Res.* **178,** 357–364 (2011).

46. Ma, S. & Huang, J. Penalized feature selection and classification in bioinformatics. *Brief. Bioinform.* **9,** 392–403 (2008).

47. Helleputte, T. & Dupont, P. Partially supervised feature selection with regularized linear models. in *Proceedings of the 26th Annual International Conference on Machine Learning* 409–416 (ACM, 2009).

48. Kohavi, R. & John, G. H. Wrappers for feature subset selection. *Artif. Intell.* **97,** 273–324 (1997).

49. Guyon, I. & Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **3,** 1157–1182 (2003).

50. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Methodol.* 267–288 (1996).

51. Guyon, I., Weston, J., Barnhill, S. & Vapnik, V. Gene selection for cancer classification using support vector machines. *Mach. Learn.* **46,** 389–422 (2002).

52. Breiman, L., Friedman, J., Stone, C. J. & Olshen, R. A. *Classification and regression trees*. (CRC press, 1984).

53. Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T. & Zeileis, A. Conditional variable importance for random forests. *BMC Bioinformatics* **9,** 307 (2008).

54. Benjamini, Y. & Hochberg, Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B Methodol.* 289–300 (1995).

55. Golub, T. R. *et al.* Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science* **286,** 531–537 (1999).

56. Barzilay, O. & Brailovsky, V. L. On domain knowledge and feature selection using a support vector machine. *Pattern Recognit. Lett.* **20,** 475–484 (1999).

57. Ding, C. & Peng, H. Minimum redundancy feature selection from microarray gene expression data. *J. Bioinform. Comput. Biol.* **3,** 185–205 (2005).

58. Xue, B., Zhang, M., Browne, W. N. & Yao, X. A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evol. Comput.* **20,** 606–626 (2016).

59. Tucker, J. D. *et al.* Accurate gene expression-based biodosimetry using a minimal set of human gene transcripts. *Int. J. Radiat. Oncol. Biol. Phys.* **88,** 933–939 (2014).

60. Riecke, A. *et al.* Gene expression comparisons performed for biodosimetry purposes on in vitro peripheral blood cellular subsets and irradiated individuals. *Radiat. Res.* **178,** 234–243 (2012).

61. Bruserud, Ø. *et al.* Expression of the potential therapeutic target CXXC5 in primary acute myeloid leukemia cells-high expression is associated with adverse prognosis as well as altered intracellular signaling and transcriptional regulation. *Oncotarget* **6,** 2794 (2015).

62. van Riggelen, J., Yetil, A. & Felsher, D. W. MYC as a regulator of ribosome biogenesis and protein synthesis. *Nat. Rev. Cancer* **10,** 301–309 (2010).

63. He, Z. & Yu, W. Stable feature selection for biomarker discovery. *Comput. Biol. Chem.* **34,** 215–225 (2010).

64. Shrestha, D. L. & Solomatine, D. P. Machine learning approaches for estimation of prediction interval for the model output. *Neural Netw.* **19,** 225–235 (2006).

65. Montgomery, D. C., Peck, E. A. & Vining, G. G. *Introduction to linear regression analysis*. (John Wiley & Sons, 2015).

66. Harnett, D. L. & Murphy, J. L. *Introductory statistical analysis*. (Addison-Wesley, 1980).

67. Casella, G. & Berger, R. L. *Statistical inference*. **2,** (Duxbury Pacific Grove, CA, 2002).

68. Stine, R. A. Bootstrap prediction intervals for regression. *J. Am. Stat. Assoc.* **80,** 1026–1031 (1985).

69. Thombs, L. A. & Schucany, W. R. Bootstrap prediction intervals for autoregression. *J. Am. Stat. Assoc.* **85,** 486–492 (1990).

70. Mendez, G. & Lohr, S. Estimating residual variance in random forest regression. *Comput. Stat. Data Anal.* **55,** 2937–2950 (2011).

71. Coulston, J. W., Blinn, C. E., Thomas, V. A. & Wynne, R. H. Approximating prediction uncertainty for random forest regression models. *Photogramm. Eng. Remote Sens.* **82,** 189–197 (2016).

72. Meinshausen, N. Quantile regression forests. *J. Mach. Learn. Res.* **7,** 983–999 (2006).

73. Chen, C., Liaw, A. & Breiman, L. Using random forest to learn imbalanced data. *Univ. Calif. Berkeley* 1–12 (2004).

74. Biau, Gãš., Devroye, L. & Lugosi, Gãą. Consistency of random forests and other averaging classifiers. *J. Mach. Learn. Res.* **9,** 2015–2033 (2008).

75. Wu, J. J. Semiparametric forecast intervals. *J. Forecast.* **31,** 189–228 (2012).

76. Taylor, J. W. & Bunn, D. W. A quantile regression approach to generating prediction intervals. *Manag. Sci.* **45,** 225–237 (1999).

77. Lee, Y. S. & Scholtes, S. Empirical prediction intervals revisited. *Int. J. Forecast.* **30,** 217–234 (2014).

78. Rubin, D. B. *Multiple imputation for nonresponse in surveys*. **81,** (John Wiley & Sons, 2004).