MARSHALL JOURNAL OF MEDICINE
Serving the Mountain State, Appalachia and Beyond
mds.marshall.edu/mjm

Volume 4 | Issue 4

Article 8

2018

Predicting Adverse Outcomes in End Stage Renal Disease: Machine Learning Applied to the United States Renal Data System

Zeid Khitan, Alexis D. Jacob, Courtney Balentine, Adam N. Jacob, Juan R. Sanabria, and Joseph I. Shapiro

DOI: http://dx.doi.org/10.18590/mjm.2018.vol4.iss4.8
Author Footnote: Acknowledgments: The authors wish to acknowledge support from the National Institutes of Health (HL109015, HL071556 and HL105649 to J.I.S) as well as the Brickstreet Foundation and the Huntington Foundation. The authors also wish to acknowledge access to the USRDS dataset furnished by the National Institutes of Health (NIDDK).

Recommended Citation

Khitan, Zeid; Jacob, Alexis D.; Balentine, Courtney; Jacob, Adam N.; Sanabria, Juan R.; and Shapiro, Joseph I. (2018) "Predicting Adverse Outcomes in End Stage Renal Disease: Machine Learning Applied to the United States Renal Data System," *Marshall Journal of Medicine*: Vol. 4: Iss. 4, Article 8.
DOI: http://dx.doi.org/10.18590/mjm.2018.vol4.iss4.8
Available at: https://mds.marshall.edu/mjm/vol4/iss4/8

References with DOI

1. Lowrie EG, Lazarus JM, Mocelin AJ, Bailey GL, Hampers CL, Wilson RE, et al. Survival of patients undergoing chronic hemodialysis and renal transplantation. N Engl J Med. 1973;288(17):863-7. https://doi.org/10.1056/nejm197304262881701

2. Foley RN, Murray AM, Li S, Herzog CA, McBean AM, Eggers PW, et al. Chronic kidney disease and the risk for cardiovascular disease, renal replacement, and death in the United States Medicare population, 1998 to 1999. J Am Soc Nephrol. 2005;16(2):489-95. https://doi.org/10.1681/asn.2004030203

3. Port FK, Orzol SM, Held PJ, Wolfe RA. Trends in treatment and survival for hemodialysis patients in the United States. Am J Kidney Dis. 1998;32(6 Suppl 4):S34-8.

4. Block GA, Klassen PS, Lazarus JM, Ofsthun N, Lowrie EG, Chertow GM. Mineral metabolism, mortality, and morbidity in maintenance hemodialysis. J Am Soc Nephrol. 2004;15(8):2208-18. https://doi.org/10.1097/01.asn.0000133041.27682.a2

5. Teng M, Wolf M, Lowrie E, Ofsthun N, Lazarus JM, Thadhani R. Survival of patients undergoing hemodialysis with paricalcitol or calcitriol therapy. N Engl J Med. 2003;349(5):446-56. https://doi.org/10.1056/nejmoa022536

6. Szczech LA, Reddan DN, Klassen PS, Coladonato J, Chua B, Lowrie EG, et al. Interactions between dialysisrelated volume exposures, nutritional surrogates and mortality among ESRD patients. Nephrol Dial Transplant. 2003;18(8):1585-91. https://doi.org/10.1093/ndt/gfg225

7. Lowrie EG, Li Z, Ofsthun N, Lazarus JM. Body size, dialysis dose and death risk relationships among hemodialysis patients. Kidney Int. 2002;62(5):1891-7. https://doi.org/10.1046/j.1523-1755.2002.00642.x

8. Chertow GM, Johansen KL, Lew N, Lazarus JM, Lowrie EG. Vintage, nutritional status, and survival in hemodialysis patients. Kidney Int. 2000;57(3):1176-81. https://doi.org/10.1046/j.1523-1755.2000.00945.x

9. Jacob AN, Khuder S, Malhotra N, Sodeman T, Gold JP, Malhotra D, et al. Neural network analysis to predict mortality in end-stage renal disease: application to United States Renal Data System. Nephron Clin Pract. 2010;116(2):c148-58. https://doi.org/10.1159/000315884

10. Description of the USRDS and its data base. Am J Kidney Dis. 1991;18(5 Suppl 2):17-20.

11. USRDS research studies. Am J Kidney Dis. 1991;18(5 Suppl 2):105-10.

12. Gullo CA, McCarthy MJ, Shapiro JI, Miller BL. Predicting Medical Student Success on Licensure Exams. Med Sci Educ. 2015;25:447-53. https://doi.org/10.1007/s40670-015-0179-6

13. Tirelli T, Gamba M, Pessani D. Support vector machines to model presence/absence of Alburnus alburnus alborella (Teleostea, Cyprinidae) in North-Western Italy: comparison with other machine learning techniques. C R Biol. 2012;335(10-11):680-6. https://doi.org/10.1016/j.crvi.2012.09.001

14. Chen T, Cao Y, Zhang Y, Liu J, Bao Y, Wang C, et al. Random forest in clinical metabolomics for phenotypic discrimination and biomarker selection. Evid Based Complement Alternat Med. 2013;2013:298183. https://doi.org/10.1155/2013/298183

15. Khondoker MR, Bachmann TT, Mewissen M, Dickinson P, Dobrzelecki B, Campbell CJ, et al. Multi-factorial analysis of class prediction error: estimating optimal number of biomarkers for various classification rules. J Bioinform Comput Biol. 2010;8(6):945-65. https://doi.org/10.1142/s0219720010005063

16. Tsiliki G, Munteanu CR, Seoane JA, Fernandez-Lozano C, Sarimveis H, Willighagen EL. RRegrs: an R package for computer-aided model selection with multiple regression models. J Cheminform. 2015;7:46. https://doi.org/10.1186/s13321-015-0094-2

17. Liu R, Li X, Zhang W, Zhou HH. Comparison of Nine Statistical Model Based Warfarin Pharmacogenetic Dosing Algorithms Using the Racially Diverse International Warfarin Pharmacogenetic Consortium Cohort Database. PLoS One. 2015;10(8):e0135784. https://doi.org/10.1371/journal.pone.0135784

18. Khitan Z, Shapiro AP, Shah PT, Sanabria JR, Santhanam P, Sodhi K, Abraham NG, Shapiro JI. Predicting adverse outcomes in chronic kidney disease using machine learning methods: data from the modification of diet in renal disease. Marshall Journal of Medicine. 2017;3(4):67-79. https://doi.org/10.18590/mjm.2017.vol3.iss4.10

19. Provost F. Machine Learning from imbalanced data sets 101 pages.stern.nyu.edu/~fprovost/Papers/skew.PDF2000

Predicting adverse outcomes in end stage renal disease: machine learning applied to the United States Renal Data System

Zeid Khitan MD[1], Alexis D. Jacob MD[2], Courtney Balentine MD[2], Adam N. Jacob BS[3], Juan R. Sanabria MD[1], Joseph I. Shapiro MD[1]

Author Affiliations:

1. Marshall University Joan C. Edwards School of Medicine, Huntington, West Virginia
2. University of Texas at Southwestern, Dallas, Texas
3. University of Toledo, Toledo, Ohio

The authors have no financial disclosures to declare and no conflicts of interest to report.

Corresponding Author:

Joseph I. Shapiro MD
Marshall University Joan C. Edwards School of Medicine
Huntington, West Virginia
Email: shapiroj@marshall.edu

Abstract

We examined machine learning methods to predict death within six months using data derived from the United States Renal Data System (USRDS). We specifically evaluated a generalized linear model, a support vector machine, a decision tree and a random forest evaluated within the context of K-10 fold validation using the CARET package available within the open source architecture R program. We compared these models with the feed forward neural network strategy that we previously reported on with this data set.

Keywords

hypertension, blood pressure, chronic renal disease, correlation, machine learning

Introduction

Patients with end stage renal disease (ESRD) have an extremely high extra renal morbidity and age adjusted mortality compared with the general population in the United States.[1-3] A number of factors have been identified which predict risk in this patient population, and some of these factors are reasonably powerful at predicting risk.[4-8] We have previously reported on patient records kept within the United States Renal Data System (USRDS);[9] a number of qualitative and quantitative measurements are presented which can be accessed rather easily from the National Institutes of Health.[10,11] In our previous study, we found that a neural network approach was not superior to that obtainable with a logistic linear approach at predicting time to death. However, since that report, advances in machine learning have allowed for the relatively easy application of other approaches which might help clinicians estimate mortality risk in this population. For that reason, the following study was performed.

Methods

Files containing de-identified patient records from the USRDS in 2007 were read in the program SAS (version 9.1), SAS Institute Inc., Cary, NC, and exported in a CSV format. Forty-two variables were selected to be used in the analysis based on their potential clinical significance and their wide availability within the USRDS as we had previously reported.[9]

All analysis was performed using the open source program R. We used a generalized linear model as our default.[12] In addition, we examined the utility of a support vector machine,[13] decision trees with the RPART package, neural networks (1 hidden layer, feed forward as previously studied(9)), and random forests.[14,15] The CARET package was used for comparison of the mature models employing 10 K- folds and 3 repeats performed on a training set (5% of total) chosen with different randomization seeds to allow for reproducibility.[16] Other packages within R were used for different specific tasks (e.g., NNet for construction of the neural network, randomForest (RFor) for constructing random forests)[17] as we recently demonstrated with the Modification of Diet with Renal Disease (MDRD) dataset.[18]

For these studies, we focused on the categorical output of survival less than six months. This outcome variable was chosen for its clinical relevance to nephrology practice.

Results and Discussion

In the records that were selected for analysis, just over 67 thousand subjects died within the first six months of starting hemodialysis (HD) therapy whereas the remaining 330 thousand subjects survived longer. The data in these two groups are summarized in Table 1. Those that died early tended to have poorer nutrition as evidenced by a lower serum albumin, serum creatinine (SCr) and body mass index (BMI) (all $p<0.01$). They also tended to be significantly older (68.3+/15.0 vs 61.3+/-15.8, $p<0.01$), have a lower prevalence of insulin dependent diabetes ($p<0.01$) and higher EPO dosages ($p<0.01$). The prevalence of ischemic heart disease and prevalence of pulmonary disease were both higher in those dying early (both $p<0.01$). Many of the data were quite similar in the two groups although because of the large numbers involved, statistical significance was noted (Table 1). The high rates of HIV and AIDS reflects the time that these data were obtained; it is quite likely that a more recent data set would have much lower prevalence for HIV and related conditions.

Table 1: Comparison of Early Death (< 6 months) and Others

	Not Dead at 6 months	Dead at 6 months	P value
N	330452	67139	
Hemoglobin	9.79+/-1.67	9.96+/-1.62	P<0.01
Albumin	3.16+/-0.67	2.95+/-0.68	P<0.01
SCr	7.45+/-3.41	6.41+/-3.00	P<0.01
BMI	27.3+/-7.0	26.0+/-6.7	P<0.01
BUN	82.9+/-27.6	83.8+/-29.2	P<0.01
SEX	53%Male	53%Male	NS
RACE	59%White 31% Black 10% other	71%White 22% Black 7% other	P<0.01
AGE	61.3+/-15.8	68.3+/-15.0	P<0.01
DIALYSIS SETTING	91% In Center	93% In Center	P<0.01
DIALYSIS TYPE	93% IHD	96% IHD	P<0.01
INCIDENT ESRD AGE	62.6+/-15.6	69.3+/-14.5	P<0.01
AIDS	19%	17%	P<0.01
HIV	19%	17%	P<0.01
ALCOH	1.3%	2.0%	P<0.01
CANCER	5.1%	10,2%	P<0.01
CARFAIL	31%	42%	P<0.01
CVA	9%	13%	P<0.01
INSULIN	24%	21%	P<0.01
DIABETES PRIMARY DX	46%	41%	P<0.01
DRUG	1.1%	1.0%	P<0.05
DYYSRYTH	5.6%	10.4%	P<0.01
EPO	68%	75%	P<0.01
HYPER	81%	72%	P<0.01
Ischemic Heart Disease	24%	32%	P<0.01
MI	8%	12%	P<0.01
NOAMBGUL	3.3%	9.1%	P<0.01
PERICARD	0.7%	0.7%	NS
PULMON	6.9%	11.7%	P<0.01
PVASC	14%	19%	P<0.01
SMOKE	5.6%	4.8%	P<0.01

*Note that because number of subjects is so high in both groups, confidence intervals around point estimate for prevalence are <<1% for all categorical values.

SCr – serum creatinine, BUN – serum urea nitrogen, ALCOH – alcohol dependency, CANCER – cancer present, CARFAIL- cardiac failure, CVA – cerebrovascular accident, HIV – human immunodeficiency virus positive, AIDS – acquired immunodeficiency syndrome present, DRUG – drug dependency, DYYSRYTH- cardiac arrhythmias, EPO – erythropoietin utilization, HYPER – hypertension present, Ischemic Heart Disease present, MI – history of myocardial infarction, NOAMBGUL- not able to ambulate, PERICARD – pericarditis, PULMON – pulmonary disease present, PVASC – peripheral vascular disease present, SMOKE – active smoker.

Different machine learning approaches yielded somewhat different fits as assessed by ROC curves (Figure 1, Table 2). In general, the best fits were obtained by either the generalized linear model (logistic regression, GLM) or the random forest (RForest) approach with the feed-forward neural network (NNet) just slightly behind. The SVM was next with the decision tree (RPart) least effective. Because the decision tree method was so weak, we did not investigate its predictions further. In contrast to the ROC curves which demonstrated significant differences (Table 2), the accuracy values obtained by the remaining four methods were remarkably similar although statistical inferiority to the linear model was evidenced by both the SVM and the NNet models. Accuracy achieved by the Rforest was similar to that obtained by the GLM. Sensitivity of the GLM was inferior to that obtained by the SVM and RForest methods whereas specificity of the NNet method was the best. Along those lines the NNet method had the highest positive predictive value (PPV) where the RForest had the highest negative predictive value (NPV). These data are all summarized in Table 3.

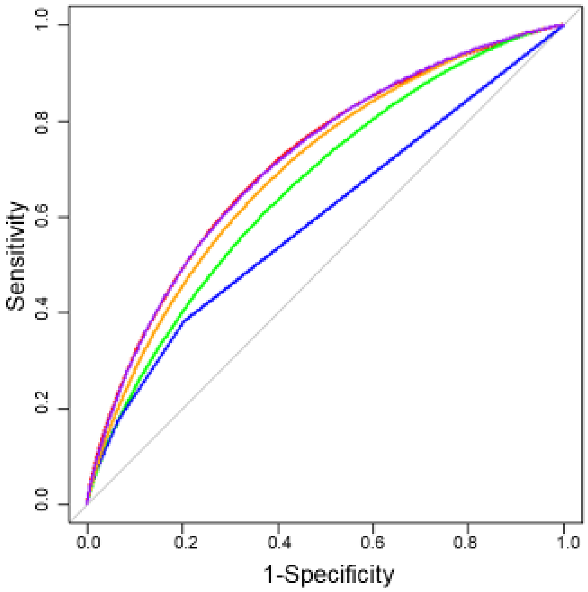

Figure 1: Receiver operator curves (ROC) achieved with generalized linear model (GLM) - red, support vector machine (SVM) – green, decision tree (RPart) – blue, feed forward neural network (NNet) – orange and random forest (RFor) – purple on testing set (95%) after training on training set (5%) with seed 33 used for randomization.

Table 2: ROC areas with the different methods:

	GLM	SVM	RPart	NNet	RForest
Mean	0.7140	0.6546	0.6119	0.6980	0.7152
SD	0.0007	0.0114	0.0087	0.0023	0.0006
P value		P<0.01	P<0.01	P<0.01	

GLM – generalized linear model, SVM – support vector machine, RPart – decision tree, NNet – feed forward neural network with 1 hidden layer, RForest – random forest. P value vs GLM. Each ROC determined for each method with 6 different seed values to generate selection of training and testing sets. Training sets chosen to 5% of the total patient records.

Table 3: Diagnostic accuracy with different methods: Calculated from N=6 seeds.

	GLM	SVM	RPart	NNet	RForest
Accuracy	0.8319+/- 0.0002	0.8311+/- 0.0001**	ND	0.8272+/- 0.0004**	0.8317+/- 0.0001
Kappa	0.073+/- 0.005	0.013+/- 0.008**	ND	0.091+/- 0.006**	0.048+/- 0.004**
Sensitivity	0.989+/- 0.001	0.998+/- 0.001**	ND	0.978+/- 0.002**	0.993+/- 0.001**
Specificity	0.058+/- 0.005	0.010+/- 0.006**	ND	0.084+/- 0.006**	0.037+/- 0.004**
PPV	0.838+/- 0.001	0.832+/- 0.001**	ND	0.840+/- 0.001**	0.835+/- 0.001**
NPV	0.521+/- 0.009	0.491+/- 0.032**	ND	0.440+/- 0.005**	0.526+/- 0.007

Data shown as mean +/- SD of six determinations. PPV – positive predictive value, NPV – negative predictive value. Positive class is “alive > 6 months.” ** p<0.01 vs GLM.

The factors that were most important to the models are shown in Table 4. It is clear that patient age, serum creatinine and serum albumin are important to the different models. Other measurements made it to the top of some of the models but not others. The different models were remarkably consistent with the importance order with which variables were chosen with the different seeds (data not shown).

Table 4: Variable importance among the different methods

	GLM	SVM	NNet	RForest
1	Albumin	Age	Disease Group	Age
2	Disease Group 81%	Incident Age 98%	Age 77%	BMI 92%
3	Non-Ambulatory 78%	SCr 69%	SCr 72%	SCr 86%
4	Hypertension 70%	Albumin 57%	Albumin 67%	BUN 81%
5	EPO 58%	DisGrp 45%	Incident Age 63%	Albumin 74%

Albumin- serum albumin, Incident Age – age of first ESRD treatment, SCr – serum creatinine, BUN – serum urea nitrogen, EPO – erythropoietin use, Hypertension – presence of hypertension.

As the entire data set had a relative paucity of early deaths, we examined how our algorithms performed with a balanced training set constructed from drawing from a subpopulation where the fraction of patients with early (< 6 month) deaths was 50:50. When we did this, all training algorithms had dramatic increases in kappa values (to about 0.2) as well as specificity values (to between 0.60 and 0.65) with marked decreases in sensitivity to be essentially matched to the specificity value obtained with that algorithm. As accuracy also decreased by about 20%, we chose to leave the training dataset unbalanced. Manuscripts addressing the challenge of unbalanced data sets recognize this problem but do not offer a universal solution.[19]

The results we observed were not very surprising based on our previous experience with this data set where we saw that the neural network model did not afford advantages over linear or actuarial strategies at predicting time to death.[9] In the current study, the logistic linear model (as we were predicting a categorical outcome) was, to all intents and purposes, comparable or superior to more sophisticated strategies at predicting early death after the initiation of dialysis therapy. Cross talk between variables clearly wasn't all that important in the determination of this important outcome; evidence strongly supported the contention that a logistic linear model captured most of the information present in this large data set.

In the analysis performed, sensitivity was calculated based on the model's ability to predict survival. Along with the high prevalence of survivors, the positive predictive value was generally in excess of 80%. This seems to be more than high enough to merit a trial of dialytic therapy. In contrast, the negative predictive value of the models hovered around 50%. Frankly, this does not come close to meeting the authors' threshold for futility of care, and it would seem irresponsible to withhold dialytic therapy for such a prediction. However, it seems that such a prediction might be of a precision sufficient to recommend additional vigilance in monitoring. With the ease of implementing the logistic linear model, this seems to be a reasonable approach based on the data used in this study which are readily available from routine clinical records (and usually submitted with the CMS-2728-U3 form).

References

1. Lowrie EG, Lazarus JM, Mocelin AJ, Bailey GL, Hampers CL, Wilson RE, et al. Survival of patients undergoing chronic hemodialysis and renal transplantation. N Engl J Med. 1973;288(17):863-7.
2. Foley RN, Murray AM, Li S, Herzog CA, McBean AM, Eggers PW, et al. Chronic kidney disease and the risk for cardiovascular disease, renal replacement, and death in the United States Medicare population, 1998 to 1999. J Am Soc Nephrol. 2005;16(2):489-95.
3. Port FK, Orzol SM, Held PJ, Wolfe RA. Trends in treatment and survival for hemodialysis patients in the United States. Am J Kidney Dis. 1998;32(6 Suppl 4):S34-8.
4. Block GA, Klassen PS, Lazarus JM, Ofsthun N, Lowrie EG, Chertow GM. Mineral metabolism, mortality, and morbidity in maintenance hemodialysis. J Am Soc Nephrol. 2004;15(8):2208-18.
5. Teng M, Wolf M, Lowrie E, Ofsthun N, Lazarus JM, Thadhani R. Survival of patients undergoing hemodialysis with paricalcitol or calcitriol therapy. N Engl J Med. 2003;349(5):446-56.
6. Szczech LA, Reddan DN, Klassen PS, Coladonato J, Chua B, Lowrie EG, et al. Interactions between dialysis-related volume exposures, nutritional surrogates and mortality among ESRD patients. Nephrol Dial Transplant. 2003;18(8):1585-91.
7. Lowrie EG, Li Z, Ofsthun N, Lazarus JM. Body size, dialysis dose and death risk relationships among hemodialysis patients. Kidney Int. 2002;62(5):1891-7.
8. Chertow GM, Johansen KL, Lew N, Lazarus JM, Lowrie EG. Vintage, nutritional status, and survival in hemodialysis patients. Kidney Int. 2000;57(3):1176-81.
9. Jacob AN, Khuder S, Malhotra N, Sodeman T, Gold JP, Malhotra D, et al. Neural network analysis to predict mortality in end-stage renal disease: application to United States Renal Data System. Nephron Clin Pract. 2010;116(2):c148-58.
10. Description of the USRDS and its data base. Am J Kidney Dis. 1991;18(5 Suppl 2):17-20.
11. USRDS research studies. Am J Kidney Dis. 1991;18(5 Suppl 2):105-10.
12. Gullo CA, McCarthy MJ, Shapiro JI, Miller BL. Predicting Medical Student Success on Licensure Exams. Med Sci Educ. 2015;25:447-53.
13. Tirelli T, Gamba M, Pessani D. Support vector machines to model presence/absence of Alburnus alburnus alborella (Teleostea, Cyprinidae) in North-Western Italy: comparison with other machine learning techniques. C R Biol. 2012;335(10-11):680-6.
14. Chen T, Cao Y, Zhang Y, Liu J, Bao Y, Wang C, et al. Random forest in clinical metabolomics for phenotypic discrimination and biomarker selection. Evid Based Complement Alternat Med. 2013;2013:298183.
15. Khondoker MR, Bachmann TT, Mewissen M, Dickinson P, Dobrzelecki B, Campbell CJ, et al. Multi-factorial analysis of class prediction error: estimating optimal number of biomarkers for various classification rules. J Bioinform Comput Biol. 2010;8(6):945-65.
16. Tsiliki G, Munteanu CR, Seoane JA, Fernandez-Lozano C, Sarimveis H, Willighagen EL. RRegrs: an R package for computer-aided model selection with multiple regression models. J Cheminform. 2015;7:46.
17. Liu R, Li X, Zhang W, Zhou HH. Comparison of Nine Statistical Model Based Warfarin Pharmacogenetic Dosing Algorithms Using the Racially Diverse International Warfarin Pharmacogenetic Consortium Cohort Database. PLoS One. 2015;10(8):e0135784.
18. Khitan Z, Shapiro AP, Shah PT, Sanabria JR, Santhanam P, Sodhi K, Abraham NG, Shapiro JI. Predicting adverse outcomes in chronic kidney disease using machine learning methods: data from the modification of diet in renal disease. Marshall Journal of Medicine. 2017;3(4):67-79.
19. Provost F. Machine Learning from imbalanced data sets 101
pages.stern.nyu.edu/~fprovost/Papers/skew.PDF2000

Appendix:

```
rm(list=ls()) #empty memory
setwd("C:/Users/shapiroj/Dropbox/Current Stuff/work") #set working directory
#load csv file and erase empty columns
library(dplyr)
dat <- read.csv("esrd.csv",stringsAsFactors=FALSE,na.string=c("",NA," ","U","Unk"))
dim(dat)
dat1 = dat[,!apply(is.na(dat), 2, all)]   # automatically get rid of empty cols at the end
#set up outcome variable as “yes”  or “no” for subsequent machine learning
A=NULL
mm=dim(dat1)[1]
for(i in 1:mm){
if(dat1[i,39]<6){
A[i]="yes"
}else{
A[i]="no"
}
}
#make all data used for fitting numeric; essential for most machine learning algorithms
dat2=dat1[,1:38]
for(i in 1:38){
  dat2[,i]=as.numeric(dat2[,i])
}
#reconstitute file z with output1 variable having outcomes as yes or no.
z=cbind(dat2,A)
colnames(z)[39]="output1"
#clean up some variables
z=z[,-c(1,5,38)]
#load additional libraries
library(rJava)
library(ROCR)
library(pROC)
library(rpart)
library(caret)
library(nnet)
library(C50)
library(ggplot2)
library(lattice)
library(randomForest)
library(rminer)
library(xgboost)
library(rBayesianOptimization)  ## Bayesian Optimization
#run simulations and save data
vv=c(2,33,15,19,5) #create vector with different seeds
#loop with different seeds
for(i in 1:5){
```

DOI: http://dx.doi.org/10.18590/mjm.2018.vol4.iss4.8

```
  set.seed(k)
  #split into training and testing subsets based on seed
  ind = sample(2, nrow(z), replace = TRUE, prob = c(0.5, 0.95))
  trainset = z[ind == 1,]
  testset = z[ind == 2,]
  #save files with output data
  vvv=paste0("esrd_10_seed_",k,".txt")
  www=paste0("esrd_10_seed_",k,".png")
  #set up training with CARET for different machine learning methods
  control = trainControl(method = "repeatedcv", number = 10, repeats = 3, classProbs =
  TRUE, summaryFunction = twoClassSummary)
  glm.model = train(output1 ~ ., data = trainset, method = "glm", metric = "ROC", trControl =
control, preProc=c("center","scale"))
  tunGrid_svm=expand.grid(sigma=c(0.015), C=c(1)) #sigma and C fit optimized
  separately based on ROC on training set
  svm.model = train(output1 ~ ., data = trainset, method = "svmRadial",metric = "ROC",
  tuneGrid=tunGrid_svm, trControl = control, preProc=c("center","scale"))
  rpart.model = train(output1 ~ ., data = trainset, method = "rpart", metric = "ROC",
  trControl = control, preProc=c("center","scale"))
  tunGrid=expand.grid(size=c(9),decay=c(0.2)) #number of hidden neurons (size) and
  decay rate optimized separately based on ROC on training set
  nnet.model = train(output1 ~ ., data=trainset, method = "nnet", metric="ROC",
  trace=FALSE, trControl=control, tuneGrid=tunGrid,
  preProc=c("center","scale"))
  tunegrid=expand.grid(.mtry=c(12)) #mtry which is number of branches to random forest
  optimized based on ROC on training set
  rfor.model = train(output1 ~ ., data=trainset, method = "rf", metric="ROC",
  trControl=control,tuneGrid=tunegrid, preProc=c("center","scale"))
  #make predictions based on models
  glm.probs = predict(glm.model, testset[,! names(testset) %in% c("output1")], type = "prob")
  svm.probs = predict(svm.model, testset[,! names(testset) %in% c("output1")], type = "prob")
  rpart.probs = predict(rpart.model, testset[,! names(testset) %in% c("output1")], type = "prob")
  nnet.probs=predict(nnet.model,  testset[,! names(testset) %in% c("output1")], type = "prob")
  rfor.probs=predict(rfor.model,  testset[,! names(testset) %in% c("output1")], type = "prob")
  #make ROC graphs
  png(www)
  glm.ROC = roc(response = testset[, c("output1")], predictor = glm.probs $yes, levels =
  levels(testset[, c("output1")]))
  plot(glm.ROC,add=F, col =" red",main=k)
  svm.ROC = roc(response = testset[, c("output1")], predictor = svm.probs $yes, levels =
  levels(testset[, c("output1")]))
  plot(svm.ROC, add = TRUE, col ="green")
  rpart.ROC = roc(response = testset[, c("output1")], predictor = rpart.probs $yes, levels =
  levels(testset[, c("output1")]))
  plot(rpart.ROC, add = TRUE, col ="blue")
```

```
nnet.ROC=roc(response = testset[, c("output1")], predictor = nnet.probs $yes, levels =
levels(testset[, c("output1")]))
plot(nnet.ROC, add = TRUE, col ="orange")
rfor.ROC=roc(response = testset[, c("output1")], predictor = rfor.probs $yes, levels =
levels(testset[, c("output1")]))
plot(rfor.ROC, add = TRUE, col ="purple")
dev.off() #close ROC graph
sink(vvv) #open text output
#confusion matrices and variable importance lists
glm.pred=predict(glm.model,testset[,!names(testset)%in% c("output1")])
t=table(glm.pred,testset[,c("output1")])
tt=confusionMatrix(glm.pred,testset[,c("output1")])
print("glm.model")
print(tt)
print(glm.ROC)
print(varImp(glm.model))
svm.pred=predict(svm.model,testset[,!names(testset)%in% c("output1")])
t=table(svm.pred,testset[,c("output1")])
tt=confusionMatrix(svm.pred,testset[,c("output1")])
print("svm.model")
print(tt)#
print(svm.ROC)
print(varImp(svm.model))
rpart.pred=predict(rpart.model,testset[,!names(testset)%in% c("output1")])
t= table(rpart.pred,testset[,c("output1")])
tt=confusionMatrix(rpart.pred,testset[,c("output1")])
print(rpart.ROC)
print(varImp(rpart.model))
nnet.pred=predict(nnet.model,testset[,!names(testset)%in% c("output1")])
t= table(nnet.pred,testset[,c("output1")])
tt=confusionMatrix(nnet.pred,testset[,c("output1")])
print("nnet.model")
print(tt)
print(nnet.ROC)
print(varImp(nnet.model))
rfor.pred=predict(rfor.model,testset[,!names(testset)%in% c("output1")])
t=table(rfor.pred,testset[,c("output1")])
tt=confusionMatrix(rfor.pred,testset[,c("output1")])
print("rfor.model")
print(tt)
print(rfor.ROC)
print(varImp(rfor.model))
sink() #close text file
}#end loop
```