Classification: Prediction protein localization sites

[redacted]

CE5310 Class Project

Introduction

Classification problem appeared in a board variety of real world problems. The classification result can be binary or multiclass. Binary classification problems have only two outcomes for each instance. Multiclass classification is to classify instances to three or more classes. For example, in medical studies, given the genomic data (DNA sequencing or DNA methylation levels) of one individual, researchers want to be able to tell whether the cancer patience of one of three cancer subtypes. In imaging recognition, given the image, we want the computers to put the image into correct category so the next steps can be done.

Dataset

The dataset for the multiclass classification problem in this project is obtained from UCI machine learning repository (https://archive.ics.uci.edu/ml/datasets/Yeast). It is first created by Kenta Nakai from Institue of Molecular and Cellular Biology, Osaka University, Japan. This

The eight explanatory variables plus the index variable are:

1. Sequence Name: Accession number for the SWISS-PROT database

2. mcg: McGeoch's method for signal sequence recognition.

3. gvh: von Heijne's method for signal sequence recognition.

4. alm: Score of the ALOM membrane spanning region prediction program.

5. mit: Score of discriminant analysis of the amino acid content of the N-terminal region (20 residues long) of mitochondrial and non-mitochondrial proteins.

6. erl: Presence of "HDEL" substring (thought to act as a signal for retention in the endoplasmic reticulum lumen). Binary attribute.

7. pox: Peroxisomal targeting signal in the C-terminus.

8. vac: Score of discriminant analysis of the amino acid content of vacuolar and extracellular proteins.

9. nuc: Score of discriminant analysis of nuclear localization signals of nuclear and non-nuclear proteins.

Methods

There are many existing methods for multiclass classifications. In the project, fuzzy rule based

Logistic model is widely used in modelling problems with a binary response variable. In the basic form, a logistic function is used. Like other forms of regression analysis, logistic regression makes use of one or more predictor variables that may be either continuous or categorical. Logistic regression is used for predicting dependent variables that take membership in one of a limited number of categories (treating the dependent variable in the binomial case as the outcome of a Bernoulli trial) rather than a continuous variable. To make the linear regression assumptions still hold, we need to apply a transformation on the response variable, so that it can be changed to a continuous one. To do that, binomial logistic regression calculates the odds of the event happening for different levels of each independent variable, and then takes its logarithm to create a continuous criterion as a transformed version of the dependent variable. The so-called logit transformation is defined as $\log \frac{\pi}{1-\pi} = \pi'$.

After the transformation, we estimate the model parameters using maximum likelihood estimation.

Ridge Regression

Ridge Regression is a technique for analyzing multiple regression data that suffer from multicollinearity. When there are too many predictors, multicollinearity probably will be a problem. It is similar to the least squares estimation, where we want to minimize the residual

adding a degree of bias to the regression estimates, ridge regression reduces the standard errors. Ridge regression is the more popular compared to principle component regression.

In ridge regression, the first step is to standardize the variables (both dependent and independent) by subtracting their means and dividing by their standard deviations. As far as standardization is concerned, all ridge regression calculations are based on standardized variables. When the final regression coefficients are displayed, they are adjusted back into their original scale. However, the ridge trace is in a standardized scale.

Ridge regression may be given a Bayesian interpretation. If we assume that each regression coefficient has expectation zero and variance 1/k, then ridge regression can be shown to be the Bayesian solution. Also, it can be shown that the ridge regression solution is achieved by adding rows of data to the original data matrix. These rows are constructed using 0 for the dependent variables and the square root of k or zero for the independent variables. One extra row is added for each independent variable. The idea that manufacturing data yields the ridge regression results has caused a lot of concern and has increased the controversy in its use and interpretation.

LASSO

The lasso (least absolute shrinkage and selection operator) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. Like ridge regression, a penalty term is

to be less than a fixed value, which forces certain coefficients to be set to zero, effectively choosing a simpler model that does not include those coefficients. This idea is similar to ridge regression, in which the sum of the squares of the coefficients is forced to be less than a fixed value, though in the case of ridge regression, this only shrinks the size of the coefficients, it does not set any of them to zero.

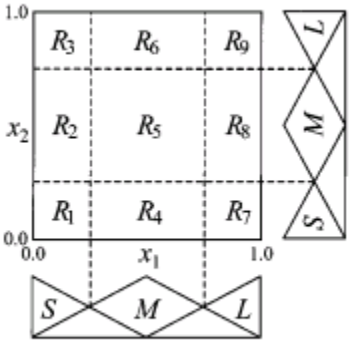
Just as ridge regression can be interpreted as linear regression for which the coefficients have been assigned normal prior distributions, lasso can be interpreted as linear regression for which the coefficients have Laplace prior distributions. The Laplace distribution is sharply peaked at zero (its first derivative is discontinuous) and it concentrates its probability mass closer to zero than does the normal distribution.

Fuzzy rule based classification

Fuzzy rule-based systems (FRBSs) are based on the fuzzy concept proposed by Zadeh in 1965. They are one of the most popular methods in pattern recognition and machine learning. These systems feature a good performance while providing interpretable models by using linguistic labels in the antecedents of their rules. Fuzzy Rule-Based Classification Systems (FRBCSs) have been successfully applied to a wide variety of domains, including bioinformatics, medical problems, or financial applications, among others.

Ishibuchi's classification technique using weight factor

When fuzzy IF–THEN rules have no certainty grades, a new pattern Xp is classified by the single winner rule Rj defined by $\mu_j{}^*(x_p) = \max\{\mu_j(x_p) : j = 1, 2, ..., N\}$. Each fuzzy IF–THEN rule has its own decision area in which new patterns are classified by that rule. The decision area of each rule is illustrated in the following figures.



Decision area of each fuzzy IF–THEN rule.

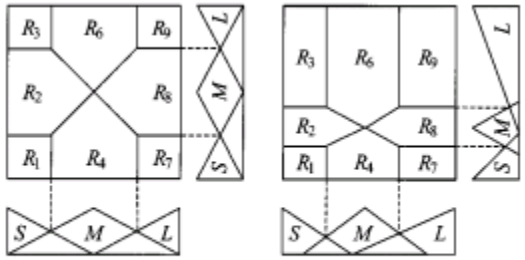


Fig. 4. Decision area of each fuzzy IF–THEN rule in the case of incomplete fuzzy rule tables.

Here the nine fuzzy IF–THEN rules generated by three antecedent linguistic values (i.e., S: small, M: medium, and L: large) on each axis of the two-dimensional pattern space [0,1] [0,1]. Each of the nine cells (or patches) corresponds to the decision area of each fuzzy IF–THEN rule.

Each fuzzy IF-THEN rule consists of antecedent linguistic values and a single consequent class with certainty grades (weights). The antecedent part is determined by a grid-type fuzzy partition from the training data. The consequent class is defined as the dominant class in the fuzzy subspace corresponding to the antecedent part of each fuzzy IF-THEN rule and the certainty grade is calculated from the ratio among the consequent class. A class of the new instance is

Wang and Mendel's method

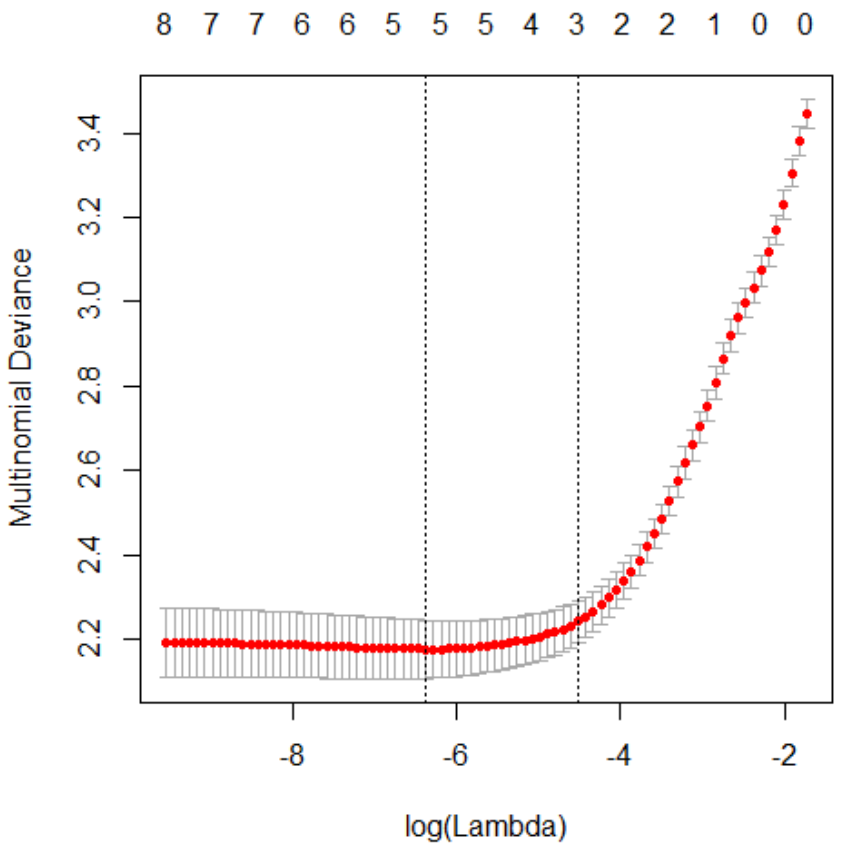
L. X. Wang and J. M. Mendel's proposed a method in 1992. For the learning process, first divide equally the input and output spaces of the given numerical data into fuzzy regions as the database. In this case, fuzzy regions refer to intervals for each linguistic term. Therefore, the length of fuzzy regions represents the number of linguistic terms. For example, the linguistic term "hot" has the fuzzy region [1, 3]. We can construct a triangular membership function having the corner points a = 1, b = 2, and c = 3 where b is a middle point that its degree of the membership function equals one. Then fuzzy IF-THEN rules are generated covering the training data, using the database from Step 1. First, we calculate degrees of the membership function for all values in the training data. For each instance in the training data, we determine a linguistic term having a maximum degree in each variable. Then, we repeat the process for each instance in the training data to construct fuzzy rules covering the training data. The next step is to determine a degree for each rule. Degrees of each rule are determined by aggregating the degree of membership functions in the antecedent and consequent parts. In this case, we are using the product aggregation operators. At last, we need to obtain a final rule base after deleting redundant rules. Considering degrees of rules, we can delete the redundant rules having lower degrees. The outcome is a Mamdani model. In the prediction phase, there are four steps: fuzzification, checking the rules, inference, and defuzzification.

technique. However, since it is based on the FRBCS model, Chi's method only takes class labels on each data to be consequent parts of fuzzy IF-THEN rules. In other words, we generate rules as in Wang and Mendel's technique (WM) and then we replace consequent parts with their classes. Regarding calculating degress of each rule, they are determined by antecedent parts of the rules. Redudant rules can be deleted by considering their degrees. Lastly, we obtain fuzzy IF-THEN rules based on the FRBCS model.

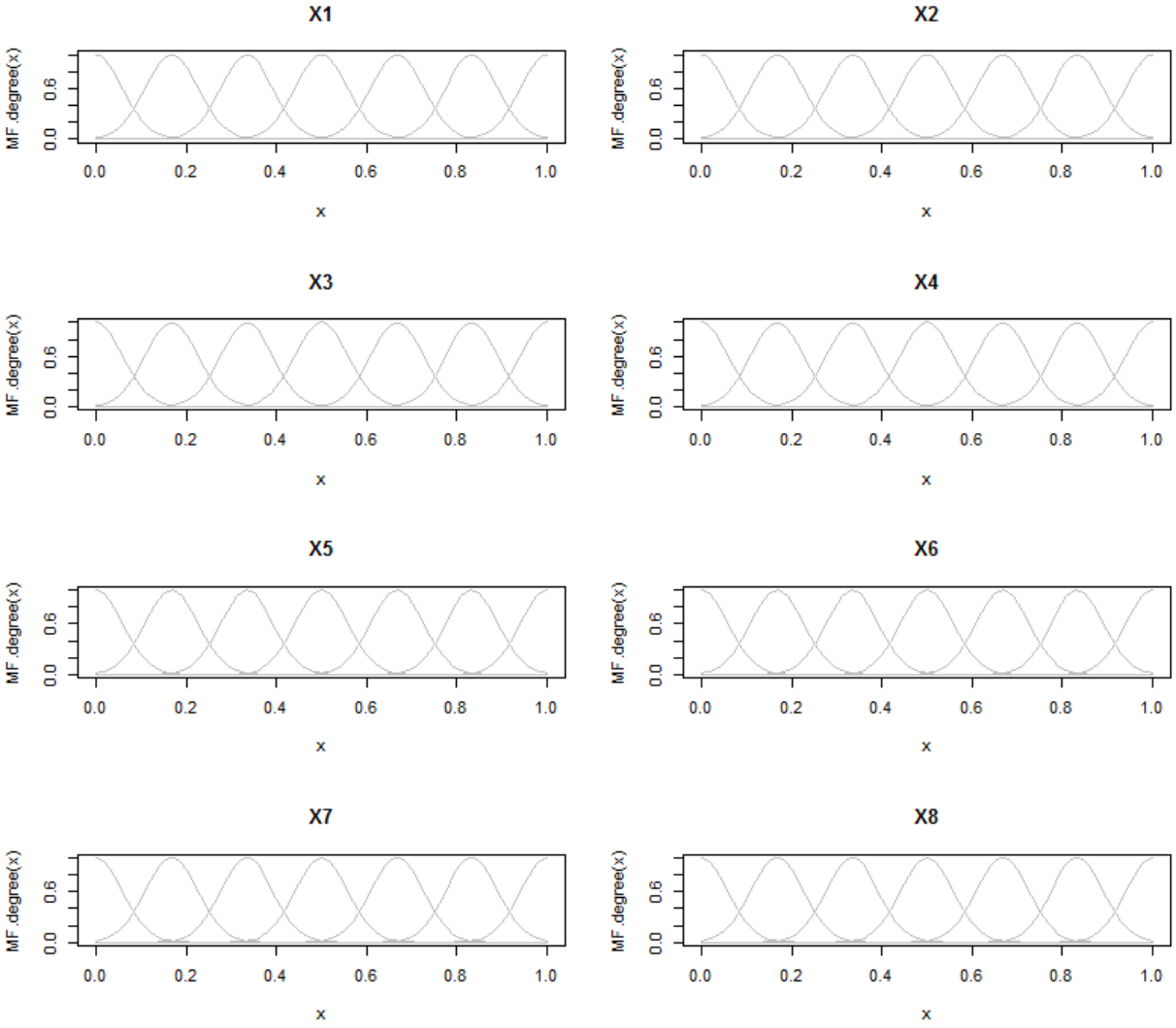
Analysis Procedures

First, we separated the data to test (25%) and training (75%). For the training data, apply different classification methods. Cross validation was used to determine the best parameters, at last the final model was fitted.

For both lasso and ridge regression, we chose the minimum lambda value obtained from the cross validation process as our penalty term coefficient.



For the fuzzy rule based classification systems, the number of labels (linguistic terms) is 7. The membership function is Guassian. The type of the defuzzification method is the weighted average method. The type of conjunction operator (t-norm) is the standard type (minimum). The type of disjunction operator (s-norm) is maximum. ZADEH is the implication function that was



Results and discussion

```
> table(lasso.pred,test[,11])

lasso.pred CYT ME3 MIT NUC
       CYT  86  10  16  47
       ME3   1  42   1   4
```

```
fuzzy.pred CYT ME3 MIT NUC
         1  20   4   8   5
         2   0  23   0   1
         3   2   3  18   2
         4  97  26  25  91
> table(fuzzy.pred2.chi,test[,11])

fuzzy.pred2 CYT ME3 MIT NUC
          1  65   8  21  52
          2   1  39   1   4
          3   5   2  22   4
          4  48   7   7  39
```

From the above confusion matrices, we can study the performance of the 4 different classification methods. Lasso and ridge regression acted similarly and are the best, while Ishibuchi's method performed the worst. In Ishibuchi's paper, the method was developed for a 2-class classification problem. It might be the reason that this method did not perform as well as others.

In addition, it will be interesting to explore how different parameter setups would influence certain classification problems.

Reference

Nakai, Kenta, and Minoru Kanehisa. "Expert system for predicting protein localization sites in gram - negative bacteria." Proteins: Structure, Function, and Bioinformatics 11.2 (1991): 95-110.

Chen, Scott Shaobing, and David L. Donoho. "Application of basis pursuit in spectrum estimation." Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on. Vol. 3. IEEE, 1998.

H. Ishibuchi and T. Nakashima, "Effect of rule weights in fuzzy rule-based classification systems", IEEE Transactions on Fuzzy Systems, vol. 1, pp. 59 - 64 (2001).

Z. Chi, H. Yan, T. Pham, "Fuzzy algorithms with applications to image processing and pattern recognition", World Scientific, Singapore (1996).

A. Gonzalez and R. Perez, "Selection of relevant features in a fuzzy genetic learning algorithm", IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 31, no. 3, pp. 417 - 425 (2001).

```
setwd("C:/Users/ysgz7/Downloads/Courses/CE5310/TCGA-PANCAN-HiSeq-801x20531")

install.packages("frbs")

library(frbs)

install.packages("caret")

library(caret)

install.packages("glmnet")

library(glmnet)

#yeast=read.csv(file= "C:/Users/ysgz7/Downloads/Courses/CE5310/yeast.csv",header = F)

#colnames(yeast)=c("ID","X1","X2","X3","X4","X5","X6","X7","X8","class")

#yeast.data=yeast[yeast$class=='CYT',]

#yeast.data2=yeast[yeast$class=='NUC',]

#yeast.data3=yeast[yeast$class=='MIT',]

#yeast.data4=yeast[yeast$class=='ME3',]

#data=rbind(yeast.data,yeast.data2,yeast.data3,yeast.data4)

#seperating the test and train data
```

```
index=sample(nrow(data),nrow(data)*.75)

train=data[index,]

test=data[-index,]

#class=read.csv(file= "C:/Users/ysgz7/Downloads/Courses/CE5310/TCGA-PANCAN-HiSeq-
801x20531/labels.csv",header = TRUE)

#data=read.csv(file= "C:/Users/ysgz7/Downloads/Courses/CE5310/TCGA-PANCAN-HiSeq-
801x20531/data.csv",header = TRUE)

#library(foreign)

#autism=read.arff(file="C:/Users/ysgz7/Downloads/Autism-Screening-Child-Data Plus
Description/Autism-Child-Data.arff")



#genes=as.matrix(data[,-1])

#levels=cbind(class[,1],unclass(class[,2]))

#genes=merge(data,class,by="X")



#use lasso for classification
```

```
Y=train[,11]

cv.lasso <- cv.glmnet(X, Y, alpha = 1, family = "multinomial")

plot(cv.lasso)

# Build the model

lasso.model=glmnet(X, Y, alpha = 1, family = "multinomial",lambda = cv.lasso$lambda.min)

ridge.model=glmnet(X, Y, alpha = 0, family = "multinomial",lambda = cv.lasso$lambda.min)



#predict

lasso.pred=predict (lasso.model ,s=cv.lasso$lambda.min ,newx=as.matrix(test[,3:10]),type =

"class")

table(lasso.pred,test[,11])



ridge.pred=predict (ridge.model ,s=cv.lasso$lambda.min ,newx=as.matrix(test[,3:10]),type =

"class")

table(ridge.pred,test[,11])
```

```
                type.snorm = "MAX", type.implication.func = "ZADEH")

range.data.input <- apply(data[, 3:10], 2, range)

unclass=unclass(train[,11])

train2=cbind(train[3:10],unclass[1:974])

fuzzy.model <- frbs.learn(train2, range.data.input,method.type, control)

fuzzy.pred = predict(fuzzy.model,test[,3:10])

table(fuzzy.pred,test[,11])

#plot the membership function

plotMF(fuzzy.model)

#chis method

fuzzy.model2 <- frbs.learn(train2, range.data.input,method.type="FRBCS.CHI", control)

fuzzy.pred2 = predict(fuzzy.model2,test[,3:10])

table(fuzzy.pred2,test[,11])

plotMF(fuzzy.model2)
```

```
control3 <- list(popu.size = 30, num.class = 4, num.labels = 4, persen_cross = 0.9,
                 max.gen = 200, persen_mutant = 0.3,
                 name="sim-0")
fuzzy.model3 <- frbs.learn(train2, range.data.input, method.type="GFS.GCCL", control3)


fuzzy.pred3 = predict(fuzzy.model3,test[,3:10])
table(fuzzy.pred3,test[,11])
plotMF(fuzzy.model3)
```