Prediction Models

Jingwen Yin jy2786

April 24, 2017

GPA

```
source("../lib/modelFunc.R")
load("../data/categorical.RData")
data.filtered <- read.csv('../data/NAreplaced.csv') #4242 1388
select <- read.csv('../data/Updated_Features/gpa_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$gpa),]
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$gpa, data.train)
colnames(data.train)[1]<-"gpa"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}

y<-train[,1]
model_selection_con(train[,-1], test, y)
```

```
## [1] 2
```

```
##                         Method Test.Error
## 1           Linear Regression     0.4082
## 2                   Full tree     0.4303
## 3                 Pruned tree     0.4617
## 4               Random Forest     0.3799
## 5 Conditional inference trees     0.4476
## 6           Gradient Boosting     0.3858
## 7      Support Vector Machine     0.3918
## 8                       LM+RF     0.3881
## 9                      SVM+RF     0.3824
```

Grit

```
data.filtered <- read.csv('../data/NAreplaced.csv')
select <- read.csv('../data/Updated_Features/grit_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$grit),]
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$grit, data.train)
colnames(data.train)[1]<-"grit"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}

y<-train[,1]
model_selection_con(train[,-1], test, y)
```

```
## [1] 2
```

```
##                          Method Test.Error
## 1           Linear Regression     0.2443
## 2                   Full tree     0.2609
## 3                 Pruned tree     0.2587
## 4               Random Forest     0.2410
## 5 Conditional inference trees     0.2577
## 6           Gradient Boosting     0.2396
## 7      Support Vector Machine     0.2539
## 8                       LM+RF     0.2401
## 9                      SVM+RF     0.2449
```

materialHardship

```
data.filtered <- read.csv('../data/NAreplaced.csv')
select <- read.csv('../data/Updated_Features/materialHardship_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$materialHardship),]
```

```
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$materialHardship, data.train)
colnames(data.train)[1]<-"materialHardship"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}

y<-train[,1]
model_selection_con(train[,-1], test, y)
```

```
## [1] 2
```

```
##                         Method Test.Error
## 1            Linear Regression     0.0181
## 2                    Full tree     0.0199
## 3                  Pruned tree     0.0200
## 4                Random Forest     0.0178
## 5 Conditional inference trees     0.0201
## 6            Gradient Boosting     0.0174
## 7      Support Vector Machine     0.0195
## 8                        LM+RF     0.0195
## 9                       SVM+RF     0.0174
```

eviction

```
data.filtered <- read.csv('../data/NAreplaced.csv')
select <- read.csv('../data/Updated_Features/eviction_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$eviction),]
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$eviction, data.train)
colnames(data.train)[1]<-"eviction"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
```

```
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}

y<-factor(train[,1])
model_selection_cat(train[,-1], test, y)
```

```
##                          Method Test.Error
## 1                           glm     0.0653
## 2                    Full tree     0.0698
## 3                  Pruned tree     0.0592
## 4                Random Forest     0.0577
## 5  Conditional inference trees     0.0592
## 6            Gradient Boosting     0.0592
## 7       Support Vector Machine     0.0592
## 8                          C5.0     0.0592
## 9                           LDA     0.0683
## 10                          KNN     0.0592
```

layoff

```
data.filtered <- read.csv('../data/NAreplaced.csv')
select <- read.csv('../data/Updated_Features/layoff_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$layoff),]
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$layoff, data.train)
colnames(data.train)[1]<-"layoff"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}
```

```r
y<-factor(train[,1])
model_selection_cat(train[,-1], test, y)
```

```
##                          Method Test.Error
## 1                           glm     0.2327
## 2                     Full tree     0.2390
## 3                   Pruned tree     0.2306
## 4                 Random Forest     0.2306
## 5  Conditional inference trees     0.2306
## 6             Gradient Boosting     0.2411
## 7        Support Vector Machine     0.2306
## 8                           C5.0     0.2285
## 9                           LDA     0.2285
## 10                          KNN     0.2516
```

jobTraining

```r
data.filtered <- read.csv('../data/NAreplaced.csv')
select <- read.csv('../data/Updated_Features/jobTraining_features.csv')
select.idx<-colnames(data.filtered) %in% as.character(select$Codes)
data.filtered <- data.filtered[,select.idx]

label <- read.csv('../data/train.csv')
label<-label[!is.na(label$jobTraining),]
Index<-as.numeric(rownames(data.filtered))%in% label$challengeID

data.train<-data.filtered[Index,]
data.train<-as.data.frame(data.train)
data.train<-cbind(label$jobTraining, data.train)
colnames(data.train)[1]<-"jobTraining"
cat.idx<-colnames(data.train) %in% categorical
# create training and test data set
set.seed(123)
train.index <- sample(1:nrow(data.train),800,replace = F)
train <- data.train[train.index,] #800*64
test <- data.train[-train.index,] #214*64
for(i in which(cat.idx)){
  data.train[,i]<-sapply(data.train[,i], factor)
  id <- which(!(test[,i] %in% unique(data.train[,i])))
  test[,i][id]<-sample(unique(data.train[,i]),length(id), replace = TRUE)
}

y<-factor(train[,1])
model_selection_cat(train[,-1], test, y)
```

```
##                          Method Test.Error
## 1                           glm     0.2269
## 2                     Full tree     0.2572
## 3                   Pruned tree     0.2269
## 4                 Random Forest     0.2239
## 5  Conditional inference trees     0.2269
## 6             Gradient Boosting     0.2284
```

```
## 7       Support Vector Machine     0.2269
## 8                         C5.0     0.2405
## 9                          LDA     0.2315
## 10                         KNN     0.2738
```