Simple Models

Laura Cosgrove

5/1/2019

The class of "simple", i.e., interpretable and inflexible, classification models can be thought to include:

- Logistic Regression
- Linear Discriminant Analysis

```
knitr::opts_chunk$set(eval = FALSE)

library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------------------------- tid
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## -- Conflicts ------------------------------------------------------------------------------ tidyverse_
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(RANN)
```

Variable Selection (Linear Combos, Zero Variance, Multicollinearity)

```
cog_data <- readRDS("./data/cog_data.RDS") %>% select(-mmse, -year_round, -sub_cort_gray_vol)

##Linear Combos
```

```r
findLinearCombos(cog_data[3:16] %>% drop_na())

colnames(cog_data[3:16]) #drop 9, cortex vol, and 14, cortical_white_matter_vol

cog_data <- cog_data %>%
  select(-cortex_vol, -cortical_white_matter_vol)

##Near Zero Variance
nearZeroVar(cog_data[3:14], saveMetrics= TRUE) #we good


## Correlation
desc_cor <- cor(cog_data[3:14] %>% drop_na())
summary(desc_cor[upper.tri(desc_cor)])

highlyCorDescr <- findCorrelation(desc_cor, cutoff = .80)
colnames(cog_data[3:14])
# Drops: total_gray_vol, supra_tentorial_vol, lh_cortical_white_matter_vol, rh_cortex_vol: makes sense.
# Retains intra_cranial_vol, lh_cortex_vol, and rh_cortical_white_matter_vol

cog_data <- cog_data %>%
  select(-total_gray_vol, -supra_tentorial_vol, -lh_cortical_white_matter_vol, -rh_cortex_vol) #from do

desc_cor2 <- cor(cog_data[3:10] %>% drop_na())
highlyCorDescr <- findCorrelation(desc_cor2, cutoff = .80)  #none over .8
summary(desc_cor2[upper.tri(desc_cor2)]) #still some high correlation

write_rds(cog_data, "./data/cog_data_preproc.RDS")
```

Preprocessing

First, divide into training and test:

```r
cog_data <- readRDS("./data/cog_data_preproc.RDS")

set.seed(1)
train_index <- createDataPartition(cog_data$cdr, p = 2/3, list = FALSE, times = 1)

cog_train <- cog_data[train_index,]
cog_test  <- cog_data[-train_index,]
```

Imputation and Centering/scaling

```r
skimr::skim(cog_train[3:10])
#note: all vars positive,
#note: <10% missing: perfect to impute
set.seed(12)
preProc_fn <- preProcess(cog_train[3:10],
           method = c("center", "scale", "knnImpute"),
          k = 5,
```

```
          knnSummary = mean,
          verbose = TRUE)

cog_train[3:10] <- predict(preProc_fn, cog_train[3:10])
cog_test[3:10] <- predict(preProc_fn, cog_test[3:10])

#Write RDS
write_rds(cog_train, "./data/cog_train_preproc.RDS")
write_rds(cog_test, "./data/cog_test_preproc.RDS")
```

More `caret` data preparation:

```
# Using caret
ctrl1 <- trainControl(method = "repeatedcv",
                      repeats = 5,
                      summaryFunction = twoClassSummary, #because we're in the two-class setting
                      classProbs = TRUE) #because need predicted class probabilities to get ROC curve

#Read RDS
cog_train <- readRDS("./data/cog_train_preproc.RDS")
cog_test <- readRDS("./data/cog_test_preproc.RDS")
```

Logistic Regression

```
set.seed(12)
logit_fit <- train(x = cog_train[3:10],
                   y = cog_train$cdr,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl1)

#library(recipes)

#recipe(cdr ~ age + prod)

names(cog_train)
logit_fit_int <- train(cdr ~ age + protective_e2 + risk_e4 + intra_cranial_vol*lh_cortex_vol + intra_cra
                   data = cog_train,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl1)

logit_fit_int2 <- train(cdr ~ age + protective_e2 + risk_e4 + intra_cranial_vol*lh_cortex_vol + intra_c
                   data = cog_train,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl1)


logit_fit #Resampled AUC: 0.7998271

summary(logit_fit$finalModel)
```

```r
logit_fit_int


summary(logit_fit_int$finalModel)
logit_fit$finalModel %>% broom::glance()
logit_fit_int$finalModel %>% broom::glance()
logit_fit_int2$finalModel %>% broom::glance()
summary(logit_fit_int2$finalModel)


train_pred_prob  <- predict(logit_fit, type = "prob")
```

```r
##Model Fit (for my practice)
broom::glance(logit_fit$finalModel)
dev <- broom::glance(logit_fit$finalModel) %>%
  pull(deviance)

pval = 1 - pchisq(dev, 655) #DOF = 665 (49 rows with NA) - 9 predictors - 1
pval #FTR, model is acceptable.

#Against Null
null_dev <- broom::glance(logit_fit$finalModel) %>%
  pull(null.deviance)

test_stat = null_dev - dev
pval = 1 - pchisq(test_stat, df = 9) #DOF = 664 - 655
pval #Reject, go with the larger model

##Interaction vs no##
dev2 <- logit_fit_int$finalModel %>% broom::glance() %>%
  pull(deviance)

test_stat = dev - dev2
pval = 1 - pchisq(test_stat, df = 2) #DOF = 654 - 656
pval #Reject, go with the larger model

#but AIC is better

logit_fit_int2

dev3 <- logit_fit_int2$finalModel %>% broom::glance() %>%
  pull(deviance)

test_stat = dev2 - dev3
pval = 1 - pchisq(test_stat, df = 1) #DOF = 1
pval #FTR, go with the smaller model
```

Performance on test data

```r
test_pred  <- predict(logit_fit_int, newdata = cog_test, type = "raw")

confusionMatrix(data = test_pred,
```

```r
                reference = cog_test$cdr,
                positive = "Dementia")

test_pred_prob  <- predict(logit_fit_int, newdata = cog_test, type = "prob")

roc_logit_test <- roc(cog_test$cdr, test_pred_prob$Dementia)

plot(roc_logit_test, legacy.axes = TRUE, print.auc = TRUE)
plot(smooth(roc_logit_test), col = 4, add = TRUE)
```

Risk allele e4, intra__cranial__vol, lh__cortex__vol, and rh__cortical__white__matter__vol are all significant. Unfortunately, intracranial volume really shouldn't carry any information.

Linear Discriminant Analysis

```r
set.seed(13)
lda_fit <- train(x = cog_train[3:10],
                 y = cog_train$cdr,
                 method = "lda", #no tuning parameter
                 metric = "ROC",
                 trControl = ctrl1)

lda_fit #Resampled AUC: 0.7999751

lda_fit$finalModel

train_pred_prob_lda  <- predict(lda_fit, type = "prob")
```

Here, we see that intracranial volume and lh__cortex__vol have the largest discriminating values on the data; rh__cortical__white__matter__vol and risk__e4 are in the second tier of importance. This dovetails nicely with our logistic regression results.

Performance on test data

```r
test_pred_lda  <- predict(lda_fit, newdata = cog_test, type = "raw")

confusionMatrix(data = test_pred_lda,
                reference = cog_test$cdr,
                positive = "Dementia")

test_pred_prob_lda  <- predict(lda_fit, newdata = cog_test, type = "prob")

roc_lda_test <- roc(cog_test$cdr, test_pred_prob_lda$Dementia)

plot(roc_lda_test, legacy.axes = TRUE, print.auc = TRUE)
plot(smooth(roc_lda_test), col = 4, add = TRUE)
```

Logistic vs. LDA:

```
res <- resamples(list(logistic = logit_fit_int,
                      lda = lda_fit)
                      )
ggplot(res) + labs(title = "AUC Performance on Resampled Data") + theme_minimal()
```

```
saveRDS(logit_fit_int, "./data/logit_int.RDS")
saveRDS(logit_fit, "./data/logit.RDS")
saveRDS(lda_fit, "./data/lda.RDS")
```