Midterm

Jeffrey Zhuohui Liang

3/7/2021

Method

Table 1: Data summary

Name	hrt_data
Number of rows	920
Number of columns	15
Column type frequency:	
factor	9
numeric	6
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
diagnosis_heart_disease	0	1.00	FALSE	2	pre: 509, abs: 411
location	0	1.00	FALSE	4	cle: 303, hun: 294, va: 200, swi: 123
sex	0	1.00	FALSE	2	mal: 726, fem: 194
chest_pain_type	0	1.00	FALSE	4	asy: 496, non: 204, aty: 174, typ: 46
fasting_blood_sugar	90	0.90	FALSE	2	fas: 692, fas: 138
resting_ecg	2	1.00	FALSE	3	nor: 551, lef: 188, ST-: 179
exercise_induced_angina	55	0.94	FALSE	2	no: 528, yes: 337
peak_exercise_st_segment	309	0.66	FALSE	3	Fla: 345, Up-: 203, Dow: 63
thalassemia	486	0.47	FALSE	3	nor: 196, rev: 192, fix: 46

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
age	0	1.00	53.51	9.42	28.0	47	54.0	60.0	77.0
resting_blood_pressure	59	0.94	132.13	19.07	0.0	120	130.0	140.0	200.0
serum_cholesterol	30	0.97	199.13	110.78	0.0	175	223.0	268.0	603.0
max_heart_rate_achieved	55	0.94	137.55	25.93	60.0	120	140.0	157.0	202.0
st_depression_exercise	62	0.93	0.88	1.09	-2.6	0	0.5	1.5	6.2
num_major_vessels_flouro	611	0.34	0.68	0.94	0.0	0	0.0	1.0	3.0

```
# split into training set
train_index = createDataPartition(hrt_data$diagnosis_heart_disease,p=0.8,list = F)

Y_tr = hrt_data$diagnosis_heart_disease[train_index]

Y_ts = hrt_data$diagnosis_heart_disease[-train_index]

options(na.action = "na.pass")
X_tr = model.matrix(diagnosis_heart_disease ~., hrt_data,na.action = "na.pass")[train_index,-1]

X_ts = model.matrix(diagnosis_heart_disease ~., hrt_data,na.action = "na.pass")[-train_index,-1]

TRC = caret::trainControl(method = "repeatedcv",repeats=5,
                          summaryFunction = twoClassSummary,
                          classProbs = T)

PPS = c("knnImpute", "center", "scale")
```

Modeling

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)

logistic_model =
  train(
    X_tr,
    Y_tr,
    method = "glmnet",
    tuneGrid = expand.grid(alpha = seq(0,1,length=6),
                           lambda = exp(seq(
                             6, to = -6, length = 50
                           ))),
    family = "binomial",
    preProcess = PPS,
    metric = "ROC",
    trControl = TRC
  )

stopCluster(cl)

p_logistics =
  ggplot(logistic_model) +
  scale_x_continuous(trans = "log")+
  labs(title = "Lasso Logistics")
```

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)
```

```
mars_model =
  train(X_tr,
        Y_tr,
        method = "earth",
        tuneGrid = expand.grid(degree = 1:3,
                               nprune = 5:20),
        preProcess = PPS,
        trControl = TRC,
        metric = "ROC")

stopCluster(cl)

p_mars = ggplot(mars_model) +labs(title ="MARS")
```

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)

knn_model =
  train(X_tr,
        Y_tr,
        method = "knn",
        tuneGrid = expand.grid(k = seq(10,60,2)),
        preProcess = PPS,
        trControl = TRC,
        metric = "ROC")

stopCluster(cl)

p_knn = ggplot(knn_model)+labs(title = "KNN")
```

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)

lda_model =  train(
  X_tr,
  Y_tr,
  method = "lda",
  preProcess = PPS,
  trControl = TRC,
  metric = "ROC"
)

stopCluster(cl)
```

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)

qda_model =  train(
  X_tr,
  Y_tr,
```

```
  method = "qda",
  preProcess = PPS,
  trControl = TRC,
  metric = "ROC"
)

stopCluster(cl)
```

```
set.seed(123123)
cl = parallel::makePSOCKcluster(5)
doParallel::registerDoParallel(cl)

nb_model =
  train(
    X_tr,
    Y_tr,
    method = "nb",
    tuneGrid = expand.grid(
      usekernel = c(T, F),
      fL = 1,
      adjust = seq(.2, 3, by = .2)
    ),
    preProcess = PPS,
    trControl = TRC,
    metric = "ROC"
  )
stopCluster(cl)
```

```
coef(logistic_model$finalModel,logistic_model$bestTune$lambda) %>%
  as.vector() %>%
  tibble(term = c("Intercept",colnames(X_tr)),
         coefficient = .) %>%
  knitr::kable(caption = "Coefficient of Lasso Logistic Regression")
```

Table 4: Coefficient of Lasso Logistic Regression

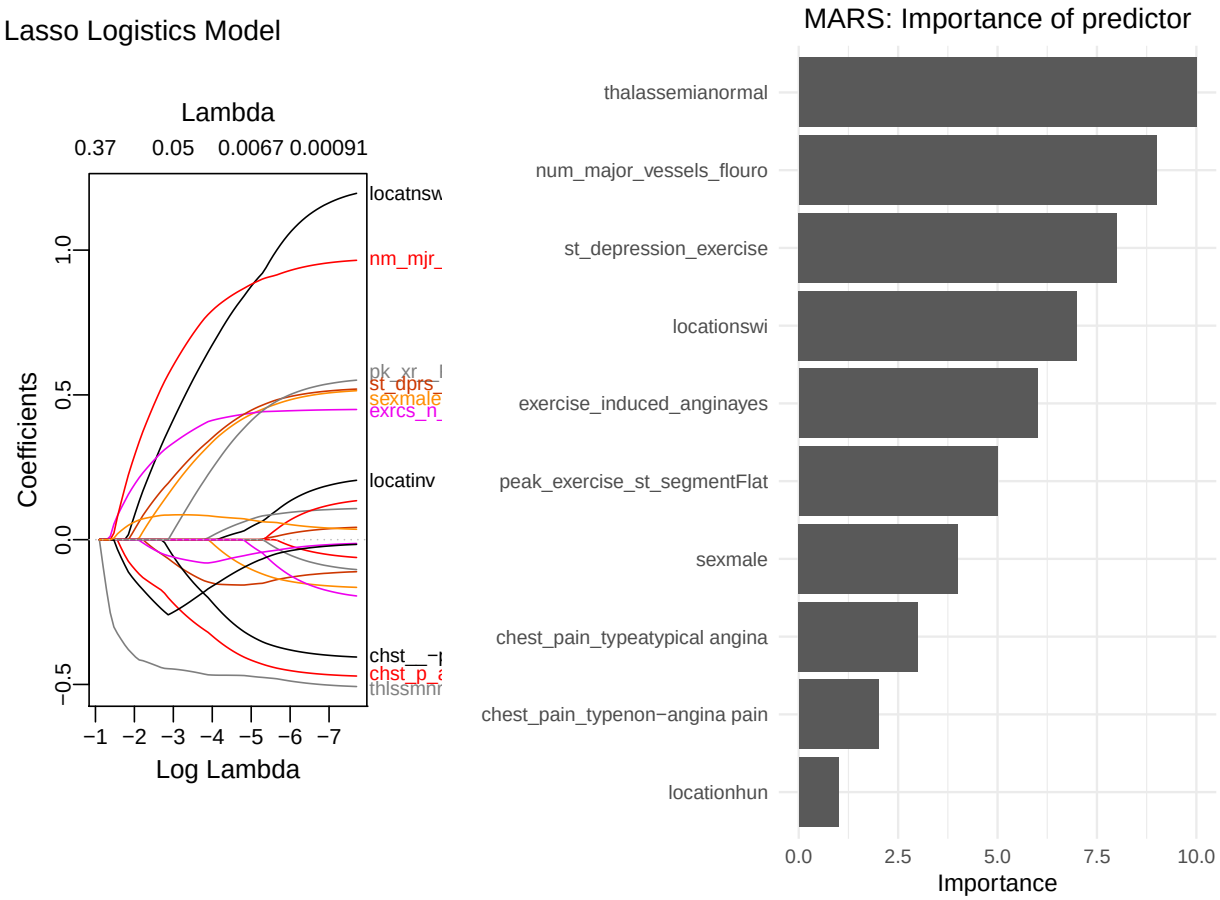
term	coefficient
Intercept	0.420
locationhun	-0.156
locationswi	0.789
locationva	0.019
age	0.048
sexmale	0.396
chest_pain_typeatypical angina	-0.385
chest_pain_typenon-angina pain	-0.288
chest_pain_typetypical angina	-0.068
resting_blood_pressure	0.000
serum_cholesterol	0.000
fasting_blood_sugarfasting blood sugar > 120 mg/dl	0.000
resting_ecgnormal	0.000
resting_ecgST-T wave abnormality	0.000
max_heart_rate_achieved	-0.063

term	coefficient
exercise_induced_anginayes	0.427
st_depression_exercise	0.409
peak_exercise_st_segmentFlat	0.342
peak_exercise_st_segmentUp-sloaping	-0.117
num_major_vessels_flouro	0.846
thalassemianormal	-0.468
thalassemiareversible defect	0.075

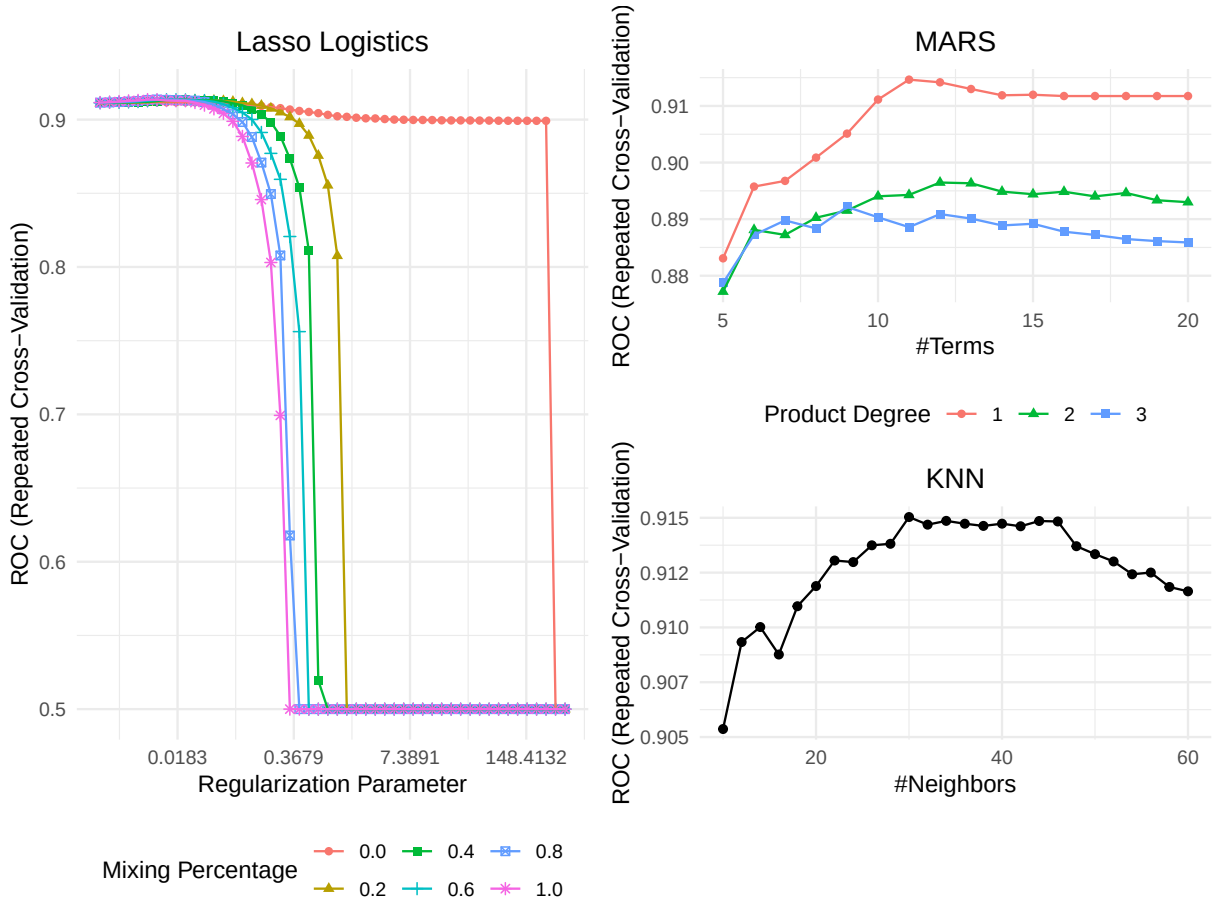
```
p_mv = vip::vip(mars_model$finalModel) + labs(title = "MARS: Importance of predictor")

p_ll = ggplotify::as.ggplot(~plot_glmnet(logistic_model$finalModel))
p_ll = p_ll + labs(title = "Lasso Logistics Model") +
  xlim(c(0.1,1))+
  ylim(c(0.1,1))

p_ll | p_mv
```




```
p_logistics | (p_mars/p_knn)
```



Performance comparison

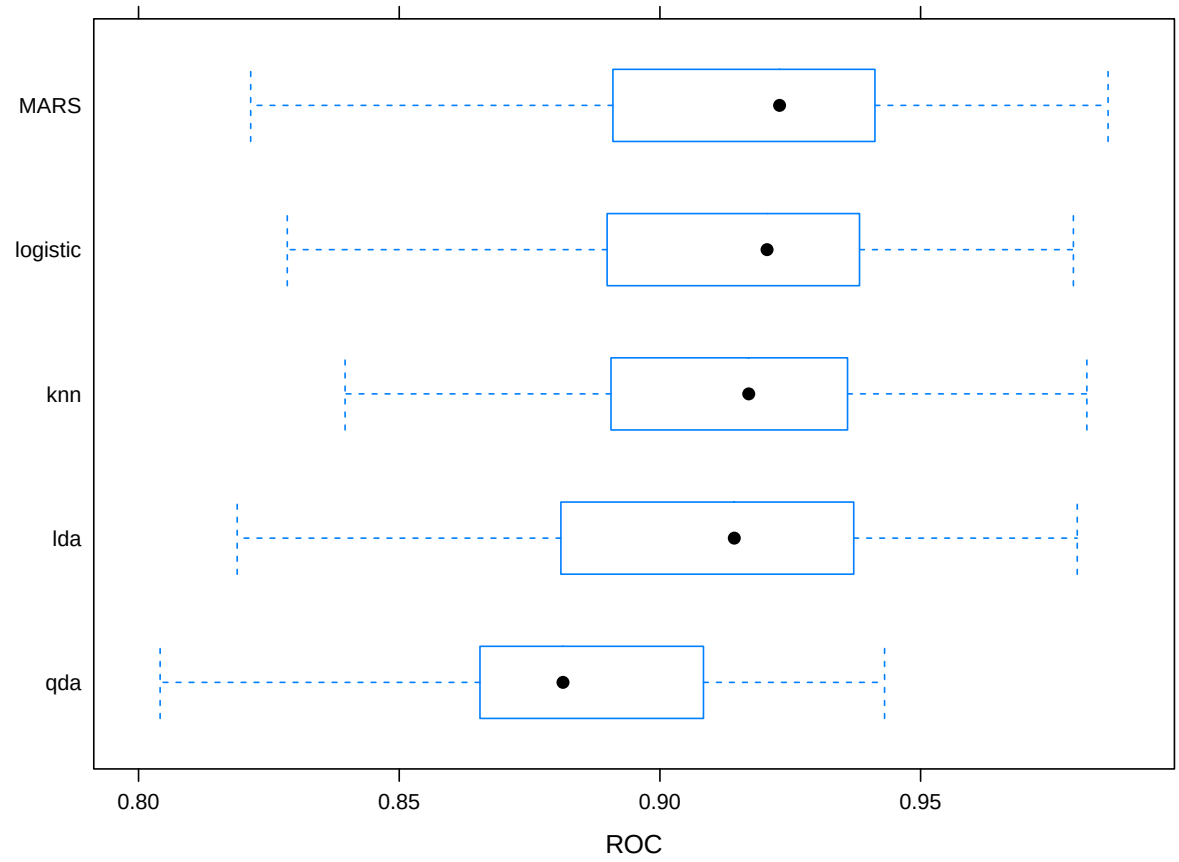
```
rsmp = resamples(
  list(
    logistic = logistic_model,
    MARS = mars_model,
    knn = knn_model,
    lda = lda_model,
    qda = qda_model
  ),
  metric = c("ROC", "Kappa")
)

summary(rsmp)
```

```
##
## Call:
## summary.resamples(object = rsmp)
##
## Models: logistic, MARS, knn, lda, qda
## Number of resamples: 50
```

```
##
## ROC
##           Min. 1st Qu. Median  Mean 3rd Qu.  Max. NA's
## logistic 0.829   0.891  0.921 0.914   0.938 0.979    0
## MARS     0.822   0.892  0.923 0.915   0.941 0.986    0
## knn      0.840   0.891  0.917 0.915   0.936 0.982    0
## lda      0.819   0.881  0.914 0.910   0.937 0.980    0
## qda      0.804   0.866  0.881 0.881   0.908 0.943    0
##
## Sens
##           Min. 1st Qu. Median  Mean 3rd Qu.  Max. NA's
## logistic 0.545   0.758  0.818 0.797   0.848 0.909    0
## MARS     0.545   0.758  0.818 0.797   0.848 0.909    0
## knn      0.545   0.735  0.788 0.782   0.848 0.909    0
## lda      0.576   0.758  0.818 0.802   0.848 0.939    0
## qda      0.606   0.758  0.812 0.799   0.848 0.909    0
##
## Spec
##           Min. 1st Qu. Median  Mean 3rd Qu.  Max. NA's
## logistic 0.732   0.829  0.854 0.862   0.902 0.976    0
## MARS     0.756   0.826  0.878 0.862   0.902 0.951    0
## knn      0.805   0.875  0.902 0.900   0.927 0.976    0
## lda      0.732   0.829  0.864 0.863   0.902 0.976    0
## qda      0.683   0.800  0.829 0.830   0.877 0.927    0
```

```
bwplot(rsmp,metric = "ROC")
```




```
ROC =
  expand.grid(
    test_X = list(X_ts),
    test_Y = list(Y_ts),
    model = list(logistic_model, mars_model, knn_model, lda_model, qda_model)
  ) %>%
  mutate(
    pred = map2(model, test_X,  ~ predict(.x, newdata = .y, type = "prob")[, 2]),
    roc = map2(test_Y, pred,  ~ pROC::roc(.x, .y))
  ) %>%
  pull(roc)

auc = c()

for (i in 1:5){
  auc = append(auc,ROC[[i]]$auc[1])
  plot(ROC[[i]],col = i, add = T * (i>1), legacy.axes = T * (i==1))
}

model_name =
  c("lasso logistic","MARS","KNN","LDA","QDA")

legend("bottomright",
```

```
      legend = paste0(model_name,"~",round(auc,3)),col=1:5,lwd=2)
```