Credit score approval prediction

Jan Alexander Jensen

2020/4/16

Handling nominal variable with dummy variable

Use library ‘fastDummies’ to handle the nominal variable $OCCUPATION_TYPE.

```r
library(fastDummies)
library(janitor)

dummy_var <- function(data){

  # Since the library 'fastDummies' tansforms all factor variable into dummy variables,
  # we will convert our target "y" (factor) into a character variable
  # to avoid it being transformed to dummy variable.

  data$y <- as.numeric(as.character(data$y))

  # transform all factor variables to dummy variables,
  # and removes the original variables that were used to generate the dummy variables.
  data_dummy <- fastDummies::dummy_cols(data, remove_selected_columns=TRUE)

  # column name convention fix (mlr3 name convention - space to underscore)
  data_dummy <- clean_names(data_dummy)


  data_dummy <- as.data.frame(sapply(data_dummy, as.numeric))
  data_dummy$y <- as.factor(data_dummy$y)

  dummy_var <- data_dummy
}

# --------------------------------------------
# ----------------------- handle missing data
# ----------------------- OCCUPATION_TYPE
# -------------------------------------------


dl_dummy_data <- read.csv2("../../credit_card_prediction/dl_na_data.csv", header = TRUE)
dl_dummy_data <- dummy_var(dl_dummy_data)

mf_dummy_data <- read.csv2("../../credit_card_prediction/mf_na_data.csv", header = TRUE)
mf_dummy_data <- dummy_var(mf_dummy_data)

mice_dummy_data <- read.csv2("../../credit_card_prediction/mice_na_data.csv", header = TRUE)
mice_dummy_data <- dummy_var(mice_dummy_data)

# row, column, NA
cat("dl_dummy_data\t", dim(dl_dummy_data), any(is.na(dl_dummy_data)), "\n")
cat("mf_dummy_data\t", dim(mf_dummy_data), any(is.na(mf_dummy_data)), "\n")
cat("mice_dummy_data\t", dim(mice_dummy_data), any(is.na(mice_dummy_data)), "\n")
```

```r
# store data with dummy variable
# write_csv2(dl_dummy_data,"credit_card_prediction/dummy_data/dl_dummy_data.csv")
# write_csv2(mf_dummy_data,"credit_card_prediction/dummy_data/mf_dummy_data.csv")
# write_csv2(mice_dummy_data,"credit_card_prediction/dummy_data/mice_dummy_data.csv")
```

```
## dl_dummy_data     25134 55 FALSE
## mf_dummy_data     36457 55 FALSE
## mice_dummy_data   36457 55 FALSE
```

Load all data for training (one-hot, dummy, IV)

```r
library(mlr3)

# function to load data into task and define target
dataToTask <- function(path, id, sep=';', header=TRUE){
  dt <- read.csv2(path, sep = sep, header = header)
  dt <- as.data.frame(sapply(dt, as.numeric))
  dt$y <- as.factor(dt$y)
  dataToTask <- TaskClassif$new(id = id, backend = dt, target = "y")
}

dl_dummy_task <-
  dataToTask("../../credit_card_prediction/dummy_data/dl_dummy_data.csv", "dl_dummy")
dl_oh_task <-
  dataToTask("../../credit_card_prediction/oh_data/dl_oh_data.csv", "dl_oh")
dl_iv_task <-
  dataToTask("../../credit_card_prediction/iv_data/dl_iv_data.csv", "dl_iv")

mf_dummy_task <-
  dataToTask("../../credit_card_prediction/dummy_data/mf_dummy_data.csv", "mf_dummy")
mf_oh_task <-
  dataToTask("../../credit_card_prediction/oh_data/mf_oh_data.csv", "mf_oh")
mf_iv_task <-
  dataToTask("../../credit_card_prediction/iv_data/mf_iv_data.csv", "mf_iv")

mice_dummy_task <-
  dataToTask("../../credit_card_prediction/dummy_data/mice_dummy_data.csv", "mice_dummy")
mice_oh_task <-
  dataToTask("../../credit_card_prediction/oh_data/mice_oh_data.csv", "mice_oh")
mice_iv_task <-
  dataToTask("../../credit_card_prediction/iv_data/mice_iv_data.csv", "mice_iv")

# combine all tasks into one list
dl <- list(dummy=dl_dummy_task, oh=dl_oh_task, iv=dl_iv_task)
mf <- list(dummy=mf_dummy_task, oh=mf_oh_task, iv=mf_iv_task)
mice <- list(dummy=mice_dummy_task, oh=mice_oh_task, iv=mice_iv_task)


# tasks[["<type>"]][["<code>"]], tasks$<type>$<code>
# ex. tasks[["dl"]][["dummy"]], tasks$dl$dummy
tasks <- list(dl=dl, mf=mf, mice=mice)

# remove unused variables (save memory)
```

```r
rm(dl, mf, mice)
rm(dl_dummy_task, mf_dummy_task, mice_dummy_task)
rm(dl_oh_task, mf_oh_task, mice_oh_task)
rm(dl_iv_task, mf_iv_task, mice_iv_task)

# print task ids and data size
for(t in tasks){
  for(c in t){
    cat(c$id, dim(c$data()), "\n")
  }
}
```

```
## dl_dummy 25134 47
## dl_oh 25134 55
## dl_iv 25134 33
## mf_dummy 36457 47
## mf_oh 36457 55
## mf_iv 36457 33
## mice_dummy 36457 47
## mice_oh 36457 55
## mice_iv 36457 33
```

KNN

```r
library(mlr3)
library("mlr3learners")

# train one model with fixed seed
train_model <- function(task, learner, resampling){
  set.seed(2020)
  print(task$id)
  train_model <- resample(task, learner, resampling, store_models = TRUE)
  print(train_model)
}

# train all task with one learner
train_all <- function(tasks, learner, resampling){

  models <- list()
  miss_name <- c('dl', 'mf', 'mice')
  code_name <- c('dummy', 'oh', 'iv')

  for(missing in miss_name){
    for(coding in code_name){
      name <- paste0(missing, "_", coding)
      task <- tasks[[missing]][[coding]]
      models[[name]] <- train_model(task, learner, resampling)
    }
  }
  train_all <- models
}

# evaluate multiple models with AUC
```

```r
evaluate_models <- function(models){
  for(m in models){
    name <- m$task$id
    auc <- m$aggregate(msr("classif.auc"))[[1]]
    max_auc <- max(m$score(msr("classif.auc"))[,9])
    print(sprintf("%10s: %.4f (max: %.4f)", name, auc, max_auc))
    #cat(paste0(name, ": ", auc, "\t(max: ", max_auc, ")\n"))
  }
}

resampling = rsmp("cv", folds = 5)
learner <- lrn("classif.kknn", id = "knn", predict_type = "prob", k=15, distance=2, scale=FALSE)
models <- train_all(tasks, learner, resampling)
```

```
## [1] "dl_dummy"
## INFO  [19:38:36.288] Applying learner 'knn' on task 'dl_dummy' (iter 1/5)
## INFO  [19:38:38.973] Applying learner 'knn' on task 'dl_dummy' (iter 2/5)
## INFO  [19:38:40.348] Applying learner 'knn' on task 'dl_dummy' (iter 3/5)
## INFO  [19:38:41.550] Applying learner 'knn' on task 'dl_dummy' (iter 4/5)
## INFO  [19:38:42.864] Applying learner 'knn' on task 'dl_dummy' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: dl_dummy
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "dl_oh"
## INFO  [19:38:44.121] Applying learner 'knn' on task 'dl_oh' (iter 1/5)
## INFO  [19:38:45.248] Applying learner 'knn' on task 'dl_oh' (iter 2/5)
## INFO  [19:38:46.365] Applying learner 'knn' on task 'dl_oh' (iter 3/5)
## INFO  [19:38:47.512] Applying learner 'knn' on task 'dl_oh' (iter 4/5)
## INFO  [19:38:48.814] Applying learner 'knn' on task 'dl_oh' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: dl_oh
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "dl_iv"
## INFO  [19:38:49.965] Applying learner 'knn' on task 'dl_iv' (iter 1/5)
## INFO  [19:38:50.937] Applying learner 'knn' on task 'dl_iv' (iter 2/5)
## INFO  [19:38:51.896] Applying learner 'knn' on task 'dl_iv' (iter 3/5)
## INFO  [19:38:52.867] Applying learner 'knn' on task 'dl_iv' (iter 4/5)
## INFO  [19:38:53.831] Applying learner 'knn' on task 'dl_iv' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: dl_iv
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mf_dummy"
## INFO  [19:38:54.873] Applying learner 'knn' on task 'mf_dummy' (iter 1/5)
## INFO  [19:38:57.725] Applying learner 'knn' on task 'mf_dummy' (iter 2/5)
## INFO  [19:39:00.443] Applying learner 'knn' on task 'mf_dummy' (iter 3/5)
## INFO  [19:39:03.290] Applying learner 'knn' on task 'mf_dummy' (iter 4/5)
## INFO  [19:39:05.886] Applying learner 'knn' on task 'mf_dummy' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mf_dummy
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mf_oh"
## INFO  [19:39:08.561] Applying learner 'knn' on task 'mf_oh' (iter 1/5)
## INFO  [19:39:11.312] Applying learner 'knn' on task 'mf_oh' (iter 2/5)
## INFO  [19:39:14.057] Applying learner 'knn' on task 'mf_oh' (iter 3/5)
## INFO  [19:39:16.833] Applying learner 'knn' on task 'mf_oh' (iter 4/5)
## INFO  [19:39:19.592] Applying learner 'knn' on task 'mf_oh' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mf_oh
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mf_iv"
## INFO  [19:39:22.591] Applying learner 'knn' on task 'mf_iv' (iter 1/5)
## INFO  [19:39:25.024] Applying learner 'knn' on task 'mf_iv' (iter 2/5)
## INFO  [19:39:27.419] Applying learner 'knn' on task 'mf_iv' (iter 3/5)
## INFO  [19:39:29.777] Applying learner 'knn' on task 'mf_iv' (iter 4/5)
## INFO  [19:39:32.550] Applying learner 'knn' on task 'mf_iv' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mf_iv
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mice_dummy"
## INFO  [19:39:34.973] Applying learner 'knn' on task 'mice_dummy' (iter 1/5)
## INFO  [19:39:37.580] Applying learner 'knn' on task 'mice_dummy' (iter 2/5)
## INFO  [19:39:40.219] Applying learner 'knn' on task 'mice_dummy' (iter 3/5)
## INFO  [19:39:42.844] Applying learner 'knn' on task 'mice_dummy' (iter 4/5)
## INFO  [19:39:45.486] Applying learner 'knn' on task 'mice_dummy' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mice_dummy
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mice_oh"
## INFO  [19:39:48.153] Applying learner 'knn' on task 'mice_oh' (iter 1/5)
## INFO  [19:39:50.801] Applying learner 'knn' on task 'mice_oh' (iter 2/5)
## INFO  [19:39:53.515] Applying learner 'knn' on task 'mice_oh' (iter 3/5)
## INFO  [19:39:56.443] Applying learner 'knn' on task 'mice_oh' (iter 4/5)
## INFO  [19:39:59.181] Applying learner 'knn' on task 'mice_oh' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mice_oh
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
## [1] "mice_iv"
## INFO  [19:40:01.975] Applying learner 'knn' on task 'mice_iv' (iter 1/5)
## INFO  [19:40:04.301] Applying learner 'knn' on task 'mice_iv' (iter 2/5)
## INFO  [19:40:06.608] Applying learner 'knn' on task 'mice_iv' (iter 3/5)
## INFO  [19:40:08.925] Applying learner 'knn' on task 'mice_iv' (iter 4/5)
## INFO  [19:40:11.465] Applying learner 'knn' on task 'mice_iv' (iter 5/5)
## <ResampleResult> of 5 iterations
## * Task: mice_iv
## * Learner: knn
## * Warnings: 0 in 0 iterations
## * Errors: 0 in 0 iterations
```

```
evaluate_models(models)
```

```
## [1] "  dl_dummy: 0.7445 (max: 0.7839)"
## [1] "     dl_oh: 0.7445 (max: 0.7839)"
## [1] "     dl_iv: 0.7445 (max: 0.7841)"
## [1] "  mf_dummy: 0.7520 (max: 0.7884)"
## [1] "     mf_oh: 0.7520 (max: 0.7884)"
## [1] "     mf_iv: 0.7520 (max: 0.7885)"
## [1] "mice_dummy: 0.7519 (max: 0.7884)"
## [1] "   mice_oh: 0.7519 (max: 0.7884)"
## [1] "   mice_iv: 0.7520 (max: 0.7886)"
```