Wrangle

Shukry Zablah

05 December, 2018

Contents

Imports

```
library(dplyr)
library(tidyr)
library(caTools)
```

Load Data

```
newPIMA <- readRDS(file = "../data/PIMA_original.Rds")
```

Modify Data

Create Variable Names

```
names(newPIMA) <- c("pregnancies", "glucoseConcentration", "bloodPressure", "skinThickness", "insulin",
```

Make the Response a Factor

```
newPIMA <- newPIMA %>%
  mutate(hasDiabetes = as.factor(hasDiabetes))
```

Mark Missing Values

In this dataset, there are missing values marked as 0. An example of how we can identify this by noting that a value of 0 for blood pressure does not make sense.

```
newPIMA <- newPIMA %>%
  mutate(glucoseConcentration = ifelse(glucoseConcentration == 0, NA_integer_, glucoseConcentration)) %
  mutate(bloodPressure = ifelse(bloodPressure == 0, NA_integer_, bloodPressure)) %>%
  mutate(skinThickness = ifelse(skinThickness == 0, NA_integer_, skinThickness)) %>%
  mutate(insulin = ifelse(insulin == 0, NA_integer_, insulin)) %>%
  mutate(bmi = ifelse(bmi == 0, NA_integer_, bmi))
```

Impuation of Missing Values

```
PIMA_noNAs <- newPIMA %>%
  drop_na()

dim(PIMA_noNAs)
```

```
## [1] 392   9
```

There are 392 observations that don't have missing values. That's half of the data. We have to find a way to replace the missing values by sensive alternatives. For now, all work is done assuming we ignore the missing values.

Test/Train Split

```
set.seed(100)

split <- with(PIMA_noNAs,
              sample.split(hasDiabetes, SplitRatio = 0.75))

PIMA_train <- subset(PIMA_noNAs, split == TRUE)
PIMA_test <- subset(PIMA_noNAs, split == FALSE)
```

Save Files

```
saveRDS(newPIMA, file = "../data/PIMA_wrangled.Rds")
saveRDS(PIMA_noNAs, file = "../data/PIMA_noNAs.Rds")
saveRDS(PIMA_train, file = "../data/PIMA_train.Rds")
saveRDS(PIMA_test, file = "../data/PIMA_test.Rds")
```