Clinical Statistics Methods Forum Data Challenge

Eric Polley & Prabin Thapa

September 20th, 2016

Introduction

The 2016 Clinical Statistics Methods Forum Data Challenge
The goal is to construct a predictor of therapeutic drug dose given baseline clinical measurements. Participants given a training data set with 3,000 patients and will be evaluated on a blinded validation data set with 1,722 patients

Overview

- Working in teams, $n \in (1, 2, \ldots, 20)$, use the training data to construct a predictor for the therapeutic dose given baseline clinical measurements
- Predictors evaluated by mean squared error of predicted dose and true dose
- With validation data set, provide predictions for each individual. A file with the Subject ID and the predicted dose as 2 columns can be email to me
- Don't forget to include your team name
- On Oct. 18th, meet for a midpoint review and group discussion
- On Nov. 15th, each team will be asked to provide a short summary on how they constructed the predictor and I will reveal the performance on the validation data
- Everyone is welcome to attend the discussion (and drink coffee)
- Prizes to be determined

Overview

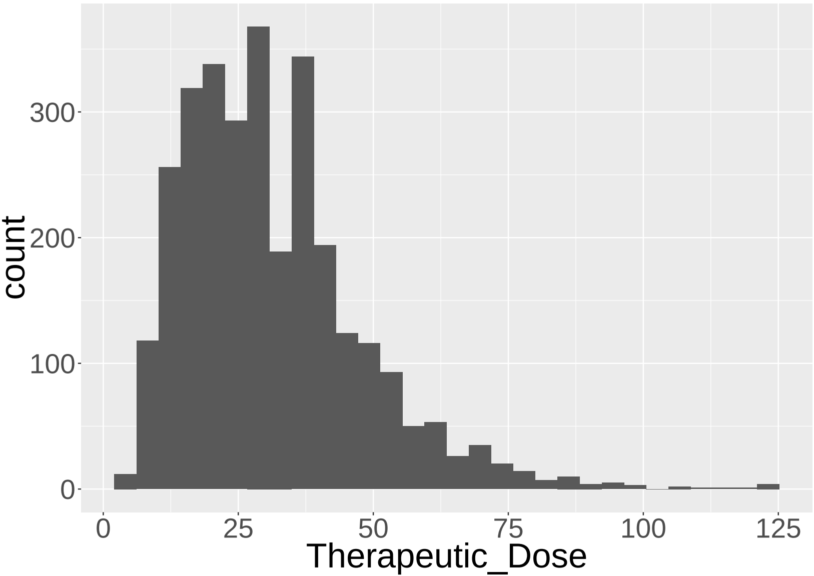
- 2 data sets provided in comma separated format
- TRAIN.CSV includes the baseline variables and outcome of interest (Therapeutic_Dose)
- VALID.CSV includes the baseline variables, need to predict the outcome
- Data available on GitHub: `https://github.com/ecpolley/CSMF_Data_Challenge`

How To Get Data

```r
# link to data on GitHub page if not available
if(file.exists("TRAIN.CSV")) {
  TRAIN <- read.csv("TRAIN.CSV")
} else {
  urlfile <- "https://raw.githubusercontent.com/ecpolley/CSMF_Data_Challenge/master/TRAIN.CSV"
  download.file(urlfile, destfile = "TRAIN.CSV")
  TRAIN <- read.csv("TRAIN.CSV")
}
dim(TRAIN)
```

```
## [1] 3000   44
```

Therapeutic_Dose

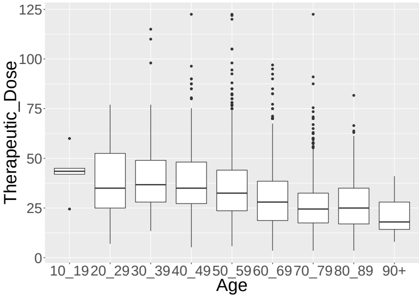


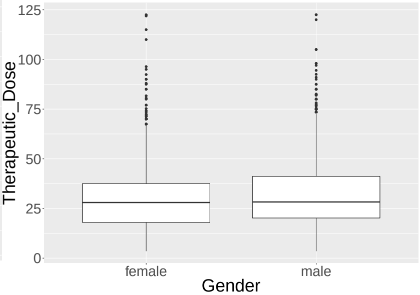
Baseline Variables

```r
str(TRAIN[, 1:8])
```

```
## 'data.frame':    3000 obs. of  8 variables:
##  $ Subject_ID  : Factor w/ 3000 levels "W0126718362",..: 1013 2027 7
##  $ Project.Site: int  4 3 7 1 6 7 16 5 7 3 ...
##  $ Gender      : Factor w/ 2 levels "female","male": 1 2 1 2 2 1 1 2
##  $ Race        : Factor w/ 4 levels "Asian","Black or African Americ
##  $ Ethnicity   : Factor w/ 3 levels "Hispanic or Latino",..: 2 2 3 2
##  $ Age         : Factor w/ 9 levels "10_19","20_29",..: 5 5 6 7 7 8
##  $ Height      : num  155 166 144 170 158 ...
##  $ Weight      : num  59 70 55.3 118.4 84 ...
```

Baseline Variables





Baseline Variables

```r
str(TRAIN[, 9:12])
```

```
## 'data.frame':    3000 obs. of  4 variables:
##  $ Indication_for_Treatment                      : Factor w/ 41 leve
##  $ Diabetes                                      : int  1 NA 0 NA NA
##  $ Congestive_Heart_Failure_and_or_Cardiomyopathy: int  0 0 0 NA NA
##  $ Valve_Replacement                             : int  0 1 0 NA NA
```

Baseline Variables

```r
str(TRAIN[, 13:29])
```

```
## 'data.frame':    3000 obs. of  17 variables:
##  $ Aspirin                                : int  0 NA 0 0 NA 0 1 0 0
##  $ Acetaminophen_or_Paracetamol           : int  0 NA 1 NA NA NA NA
##  $ Simvastatin                            : int  0 NA 0 0 NA 0 0 0 0
##  $ Atorvastatin                           : int  0 NA NA NA NA NA 1
##  $ Fluvastatin                            : int  0 NA NA NA NA NA NA
##  $ Lovastatin                             : int  0 NA NA NA NA NA NA
##  $ Pravastatin                            : int  0 NA NA NA NA NA NA
##  $ Rosuvastatin                           : int  0 NA NA NA NA NA NA
##  $ Cerivastatin                           : int  0 NA NA NA NA NA NA
##  $ Amiodarone                             : int  0 NA 0 0 NA 0 1 0 0
##  $ Carbamazepine                          : int  0 NA NA NA NA NA NA
##  $ Phenytoin                              : int  0 NA NA NA NA NA NA
##  $ Rifampin                               : int  0 NA NA NA NA NA NA
##  $ Sulfonamide_Antibiotics                : int  0 NA NA NA NA NA NA
##  $ Macrolide_Antibiotics                  : int  0 NA NA NA NA NA NA
##  $ Anti_fungal_Azoles                     : int  0 NA NA NA NA NA NA
##  $ Herbal_Medications_Vitamins_Supplements: int  1 NA NA NA NA NA NA
```

Baseline Variables

```r
str(TRAIN[, 30:33])
```

```
## 'data.frame':    3000 obs. of  4 variables:
##  $ Current_Smoker : int  NA NA NA 0 NA 0 1 0 NA NA ...
##  $ GeneA          : Factor w/ 11 levels "*1/*1","*1/*11",..: 1 1 1 1
##  $ GeneB          : Factor w/ 3 levels "A/A","A/G","G/G": 2 1 2 2 1
##  $ NoComorbidities: int  0 0 0 0 1 0 0 0 0 0 ...
```

Baseline Variables

```r
str(TRAIN[, 34:43])
```

```
## 'data.frame':    3000 obs. of  10 variables:
##  $ Biomarker_1 : num  -4.76 -9.35 -15.28 -9.45 -10.71 ...
##  $ Biomarker_2 : num  -1.608 3.454 5.451 -0.612 2.161 ...
##  $ Biomarker_3 : num  1.477 0.463 2.413 2.913 1.268 ...
##  $ Biomarker_4 : num  4.55 3.42 4.18 4.85 5.56 ...
##  $ Biomarker_5 : num  2.41 -1.61 3.74 4.42 2.18 ...
##  $ Biomarker_6 : num  3.3 -1.15 6.26 5.39 3.03 ...
##  $ Biomarker_7 : num  1.885 1.755 1.731 1.582 0.114 ...
##  $ Biomarker_8 : num  -7.21 -7.01 -4.2 -5.01 -5.88 ...
##  $ Biomarker_9 : num  -0.1227 0.2597 -0.0925 -0.1557 -1.0295 ...
##  $ Biomarker_10: num  5.22 8.36 8.26 7.31 7.17 ...
```

Basic Regression Predictor

```r
fit <- lm(Therapeutic_Dose ~ Age + Gender +
            Biomarker_1 + Biomarker_2, data = TRAIN)
fit
```

```
##
## Call:
## lm(formula = Therapeutic_Dose ~ Age + Gender + Biomarker_1 +
##     Biomarker_2, data = TRAIN)
##
## Coefficients:
## (Intercept)     Age20_29     Age30_39     Age40_49     Age50_59
##    42.09257     -2.81562     -2.43856     -3.97531     -6.98273
##    Age60_69     Age70_79     Age80_89       Age90+   Gendermale
##   -12.66090    -16.90053    -16.59322    -21.00917      2.24424
## Biomarker_1  Biomarker_2
##     0.03230     -0.08105
```

How To Get Data

```r
# link to data on GitHub page if not available
if(file.exists("VALID.CSV")) {
  VALID <- read.csv("VALID.CSV")
} else {
  urlfile <- "https://raw.githubusercontent.com/ecpolley/
    CSMF_Data_Challenge/master/VALID.CSV"
  download.file(urlfile, destfile = "VALID.CSV")
  VALID <- read.csv("VALID.CSV")
}
dim(VALID)
```

```
## [1] 1722   43
```

Basic Regression Predictor

```r
pred_lm <- predict(fit, newdata = VALID)
pred_lm_df <- data.frame(ID = VALID$Subject_ID, predict = p
head(pred_lm_df)
```

```
##            ID  predict
## 1 W0151579099 31.46625
## 2 W0151957677 37.27828
## 3 W0151693136 26.53852
## 4 W0135312561 28.54693
## 5 W0151752657 24.78085
## 6 W0150479653 41.08882
```

```r
# write out prediction table
# write.csv(pred_lm_df, "final_predictions.csv")
```

Cross-Validation

Prior to submitting predictions, should evaluate performance

```
## get V-fold CV estimate of MSE
V <- 10
N <- nrow(TRAIN)
MSE_cv <- rep(NA, V)  # placeholder for CV MSE estimates
# list of row ids by V validation splits
validRows <- split(sample(1:N), rep(1:V, length=N))
for(v in seq(V)) {
  tempTRAIN <- TRAIN[-validRows[[v]], ]
  tempVALID <- TRAIN[validRows[[v]], ]
  fit_cv <- lm(Therapeutic_Dose ~ Age + Gender +
            Biomarker_1 + Biomarker_2, data = tempTRAIN)
  pred_cv <- predict(fit_cv, newdata = tempVALID)
  MSE_cv[v] <- mean((pred_cv - tempVALID$Therapeutic_Dose)
}
```

Cross-Validation

```
# get V-fold CV estimate of MSE
summary(MSE_cv)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   215.9   223.4   238.3   243.1   260.0   278.1
```

```
# CV R-squared
1 - mean(MSE_cv)/var(TRAIN$Therapeutic_Dose)
```

```
## [1] 0.0933959
```

Questions?