Thesis Simulation Single Run for Chapter 4

Dasha Asienga

2024-03-25

Contents

This file is intended to run the Seldonian framework on the parent simulation data set as a single run, before scaling it into multiple trials.

Reading in the Data

First, let's read in the parent simulation data set.

```
compas_sim_path <- "/home/dasienga24/Statistics-Senior-Honors-Thesis/Data Sets/COMPAS/compas_sim.csv"
compas_sim_parent <- read.csv(compas_sim_path)
```

Logistic Regression

First, we'll want to run the logistic regression as our baseline model and obtain the 3 key performance measures: convergence, accuracy, and discrimination.

Fit the logistic regression model.

```
lr <- glm(is_recid ~ age + prior_offense,
            data = compas_sim_parent,
            family = binomial(logit))
```

Convergence

Obtain convergence as an object. We would expect LR to always converge.

```r
lr_converged <- lr[["converged"]]
lr_converged
```

```
## [1] TRUE
```

Accuracy

Conditional on convergence, obtain the accuracy as an object, which in this case is 60.76%.

```r
lr_accuracy <- count(round(lr[["fitted.values"]]) == lr[["y"]])/nrow(compas_sim_parent)
lr_accuracy
```

```
## n_TRUE
## 0.6076
```

Discrimination

Conditional on convergence, obtain the discrimination statistic as an object, which in this case is 0.2784 or 27.84%.

```r
preds <- predict(lr, newdata = compas_sim_parent, type="response")

compas_sim_parent <- compas_sim_parent %>%
  mutate(preds = preds,
         prediction = round(preds, 0),
         pred_risk = ifelse(prediction == 0, 'Low', 'High'))

discrimination <- compas_sim_parent %>%
  dplyr::select(race, pred_risk, is_recid) %>%
  group_by(race, is_recid) %>%
  mutate(total = n()) %>%
  group_by(pred_risk, race, total) %>%
  summarise("reoffended" = count(is_recid == 1),
            "did_not_reoffend" = count(is_recid == 0)) %>%
  pivot_longer(cols = c("reoffended", "did_not_reoffend"),
               names_to = "recidivism") %>%
  pivot_wider(
    id_cols = c("pred_risk", "recidivism", "total"),
    names_from = "race",
    values_from = value
  ) %>%
  rename("Black" = `African-American`,
         "White" = `Caucasian`) %>%
  mutate(Black = round(100 * Black / total, 2),
         White = round(100 * White / total, 2)) %>%
  dplyr::select(-total) %>%
  group_by(pred_risk, recidivism) %>%
  summarize(Black = max(Black, na.rm = TRUE),
            White = max(White, na.rm = TRUE)) %>%
  filter((pred_risk == "High" & recidivism == "did_not_reoffend") |
           (pred_risk == "Low" & recidivism == "reoffended")
  )
```

```
lr_disc_stat <- sum(abs(discrimination$White - discrimination$Black))/100
lr_disc_stat
```

```
## [1] 0.2784
```

The results from the logistic regression are now easily saveable and retrievable, which will be useful when we scale the process.

Seldonian Framework