

Thesis Simulation Document for Chapter 4

Dasha Asienaga

2024-03-13

Contents

Data Generation Mechanism	1
Reading in the Data	1
Data Subsetting	1
Generating the Parent Simulation Data Set	4
Examining Distributions of the Recidivism in the Parent Data Set	5
Assessing Baseline Predictive Performance of the Parent Data Set	6

This file is intended to contain all the code and information to set up the simulation study and supplement Chapter 4.

Data Generation Mechanism

We're interested in creating a data set that has 50-50 class balance, even across the demographic group, and also has better predictive performance than the COMPAS tool. For this set-up, we will only use 2 variables from the COMPAS data set: 1 continuous variable and 1 categorical variable.

Reading in the Data

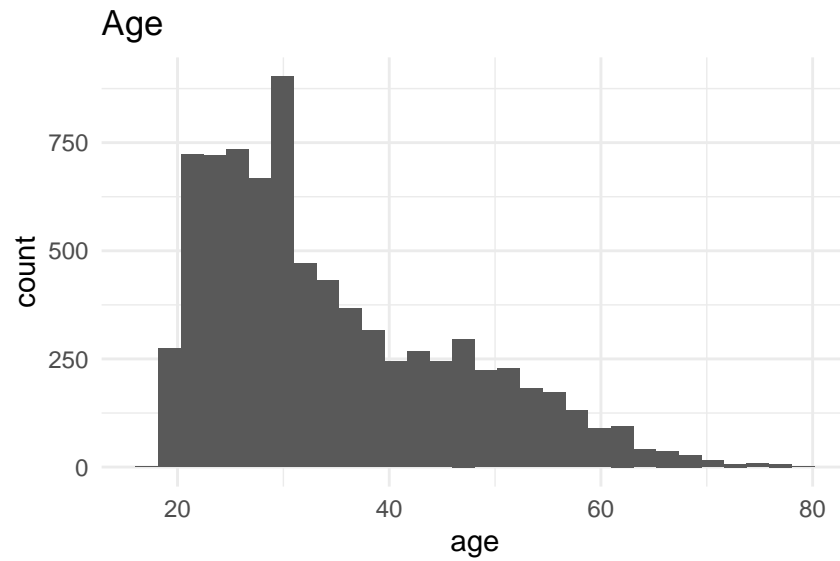
First, let's read in the data.

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

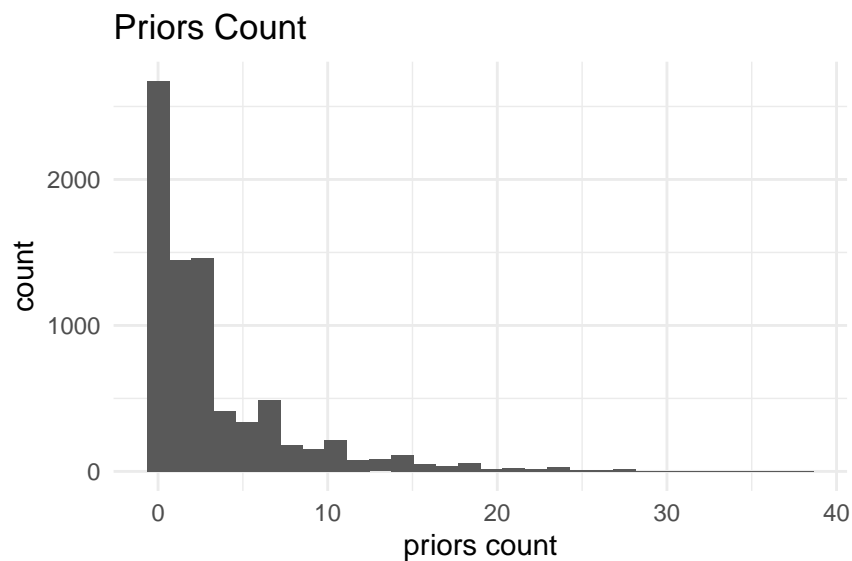
Data Subsetting

Next, let's plot the distributions of the continuous variables to choose which one we'll proceed with.

```
compas_sim %>%
  ggplot(mapping = aes(x = age)) +
  geom_histogram() +
  theme_minimal() +
  labs(title = "Age")
```



```
compas_sim %>%
  ggplot(mapping = aes(x = priors_count)) +
  geom_histogram() +
  theme_minimal() +
  labs(title = "Priors Count",
       x = "priors count")
```



Because age has more variation, we'll use it as our continuous variable. We'll convert `priors_count` into a categorical variable.

```
compas_sim <- compas_sim %>%
  mutate(prior_offense = ifelse(priors_count > 0, 1, 0)) %>%
  dplyr::select(c(race, prior_offense, age, is_recid))
```

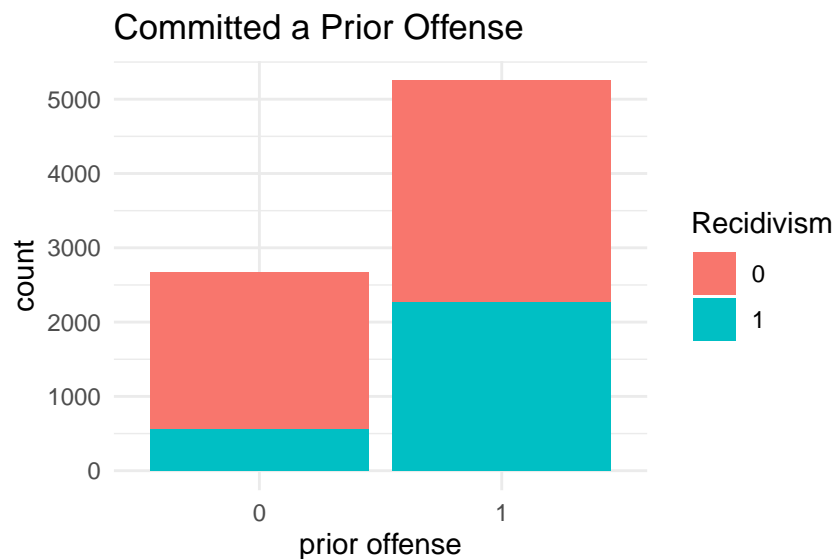
age seems to be a useful predictor for recidivism.

```
favstats(age ~ is_recid, data = compas_sim)
```

```
##   is_recid min Q1 median Q3 max      mean      sd    n missing
## 1         0  18  26     33 44  80 35.92591 12.24397 5102         0
## 2         1  18  24     29 37  78 32.25797 10.57842 2822         0
```

Whether a defendant has committed a prior offense or not appears to be a useful predictor for recidivism as well.

```
compas_sim %>%
  ggplot(mapping = aes(x = as.factor(prior_offense), fill = as.factor(is_recid))) +
  geom_bar() +
  theme_minimal() +
  labs(title = "Committed a Prior Offense",
       fill = "Recidivism",
       x = "prior offense")
```



We'll proceed with these 2 variables – age and prior_offense for the simulation study. A glimpse of the data is shown below.

```
compas_sim <- compas_sim %>%
  mutate(prior_offense = as.factor(prior_offense),
         is_recid = as.factor(is_recid))

glimpse(compas_sim)
```

```
## Rows: 7,924
## Columns: 4
## $ race      <chr> "African-American", "African-American", "Caucasian", "Ca~
## $ prior_offense <fct> 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0,~
## $ age       <int> 34, 24, 41, 39, 20, 26, 27, 23, 37, 22, 41, 47, 31, 25, ~
## $ is_recid   <fct> 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, ~
```

```
head(compas_sim)
```

##		race	prior_offense	age	is_recid
## 1	African-American		0	34	1
## 2	African-American		1	24	1
## 3	Caucasian		1	41	1
## 4	Caucasian		0	39	0
## 5	Caucasian		0	20	0
## 6	Caucasian		0	26	0

Generating the Parent Simulation Data Set

We want a setting with 50-50 class balance for each combination of race and recidivism status. To achieve that, we'll perform sample observations with replacement. Let's create a data set with 1250 observations in each of these 4 groups, hence, 5000 observations total.

First, let's subset these 4 groups.

```
compas_b_y <- compas_sim %>%  
  filter(race == "African-American" & is_recid == 1)  
  
compas_b_n <- compas_sim %>%  
  filter(race == "African-American" & is_recid == 0)  
  
compas_w_y <- compas_sim %>%  
  filter(race == "Caucasian" & is_recid == 1)  
  
compas_w_n <- compas_sim %>%  
  filter(race == "Caucasian" & is_recid == 0)
```

Next, let's randomly sample 1250 observations from each of these groups.

```
compas_b_y_balanced <- compas_b_y[sample(nrow(compas_b_y), 1250, replace = TRUE),]  
compas_b_n_balanced <- compas_b_n[sample(nrow(compas_b_n), 1250, replace = TRUE),]  
compas_w_y_balanced <- compas_w_y[sample(nrow(compas_w_y), 1250, replace = TRUE),]  
compas_w_n_balanced <- compas_w_n[sample(nrow(compas_w_n), 1250, replace = TRUE),]
```

Finally, let's union all these together into a single data set.

```
compas_sim_balanced <- rbind(compas_b_y_balanced,  
                             compas_b_n_balanced,  
                             compas_w_y_balanced,  
                             compas_w_n_balanced)
```

Let's also shuffle the data set row orderings to aid the machine learning algorithms later.

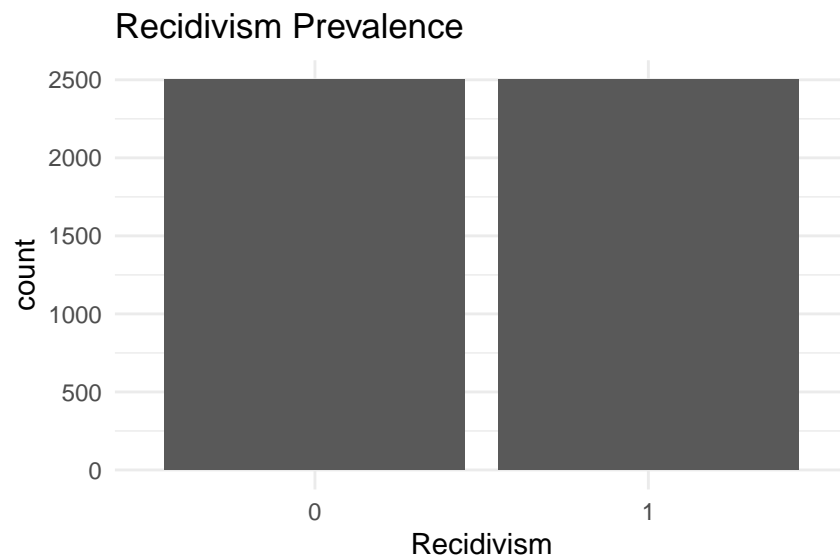
```
compas_sim_balanced <- compas_sim_balanced[sample(nrow(compas_sim_balanced),
                                                  5000,
                                                  replace = FALSE),]
```

The parent data set is now ready.

Examining Distributions of the Recidivism in the Parent Data Set

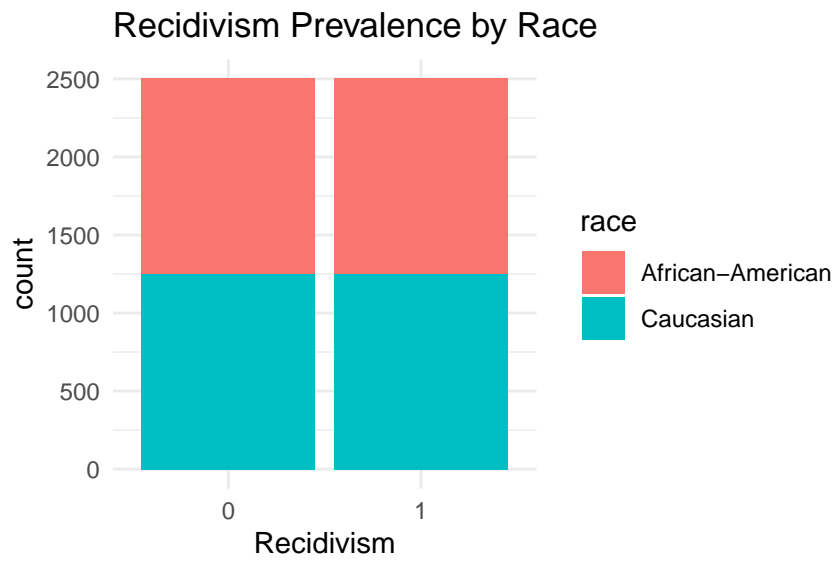
The bar plot below shows that we've achieved perfect class balance.

```
compas_sim_balanced %>%
  ggplot(mapping = aes(x = is_recid)) +
  geom_bar() +
  theme_minimal() +
  labs(x = "Recidivism",
       title = "Recidivism Prevalence")
```



The bar plot below reveals that the balance is preserved by race as well.

```
compas_sim_balanced %>%
  ggplot(mapping = aes(x = is_recid, fill = race)) +
  geom_bar() +
  theme_minimal() +
  labs(x = "Recidivism",
       title = "Recidivism Prevalence by Race")
```



Assessing Baseline Predictive Performance of the Parent Data Set