QTM 220 HW #9

Author

Veronica Vargas

QTM 220 HW #9

```
library(tidyverse)
```

```
Warning: package 'tidyverse' was built under R version 4.3.3
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
✔ dplyr     1.1.3     ✔ readr     2.1.4
✔ forcats   1.0.0     ✔ stringr   1.5.0
✔ ggplot2   3.4.3     ✔ tibble    3.2.1
✔ lubridate 1.9.2     ✔ tidyr     1.3.0
✔ purrr     1.0.2     
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr)
library(ggplot2)
library(cobalt)
```

```
Warning: package 'cobalt' was built under R version 4.3.3
```

```
 cobalt (Version 4.5.5, Build Date: 2024-04-02)
```

```
library(rpart)
```

```
Warning: package 'rpart' was built under R version 4.3.3
```

```
library(rpart.plot)
```

```
Warning: package 'rpart.plot' was built under R version 4.3.3
```

```
library(ipw)
```

```
Warning: package 'ipw' was built under R version 4.3.3
```

```
library(broom)
```

```
Warning: package 'broom' was built under R version 4.3.3
```

Exercise #1 Trees Exercise

```
data(iris)
head(iris)
```

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
```

```
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
```

(a) Variable Type

```
class(iris$Species)
```

```
[1] "factor"
```

```
class(iris$Sepal.Length)
```

```
[1] "numeric"
```

The Species variable is categorical while the Sepal.Length variable is numerical.

(b) Regression Tree

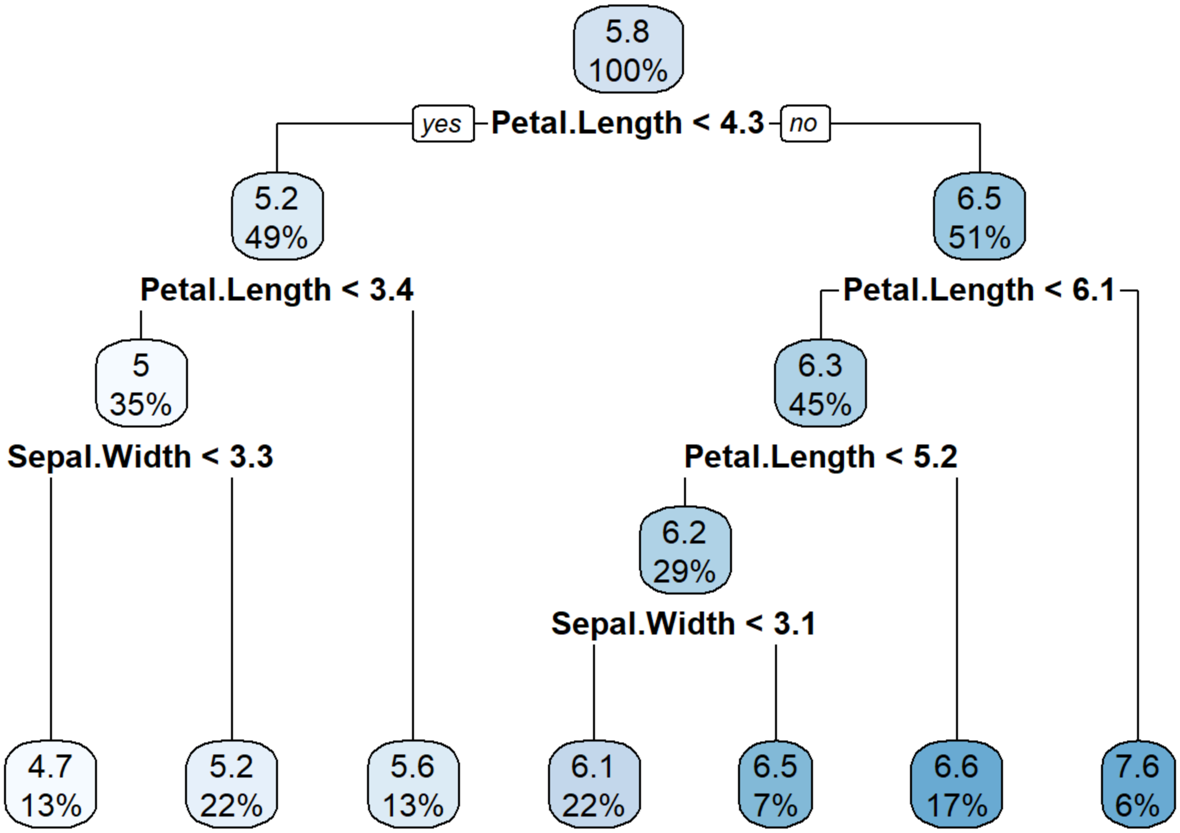
```
base_model <- rpart(Sepal.Length ~.,
      data = iris,
      method = "anova")
base_model
```

```
n= 150

node), split, n, deviance, yval
      * denotes terminal node

 1) root 150 102.1683000 5.843333
   2) Petal.Length< 4.25 73  13.1391800 5.179452
     4) Petal.Length< 3.4 53   6.1083020 5.005660
       8) Sepal.Width< 3.25 20   1.0855000 4.735000 *
       9) Sepal.Width>=3.25 33   2.6696970 5.169697 *
     5) Petal.Length>=3.4 20   1.1880000 5.640000 *
   3) Petal.Length>=4.25 77  26.3527300 6.472727
     6) Petal.Length< 6.05 68  13.4923500 6.326471
      12) Petal.Length< 5.15 43   8.2576740 6.165116
        24) Sepal.Width< 3.05 33   5.2218180 6.054545 *
        25) Sepal.Width>=3.05 10   1.3010000 6.530000 *
      13) Petal.Length>=5.15 25   2.1896000 6.604000 *
     7) Petal.Length>=6.05 9   0.4155556 7.577778 *
```

Since we are looking at a numerical variable, this is a regression tree.

(c) Plotting Tree

```
rpart.plot(base_model)
```



There are seven leaf nodes. The predicted Sepal.Length for the region with the lowest number of observations is 6.5.

(d) Prediction w/ Tree

Looking at the regression tree, the predicted sepal length for a new observation with the corresponding characteristics would be 5.2.

Exercise #2 IPW for Missingness

```
dogs <- read.csv("C:/Users/13015/OneDrive - Emory University/Documents/Fall 2024/QTM
         220/dogs.missing.csv")
head(dogs)
```

```
  Obs  Name Gender Fixed       Color
1   1   Max   Male   Yes  Dark brown
2   2  Isla Female   Yes       Black
3   3 Tyson   Male    No       Black
4   4  Lexi Female   Yes Light brown
5   5         Male   Yes Light brown
6   6  Lola Female   Yes       Black
                                Heritage   Age Weight
1 Designer/deliberate mix (e.g., labradoodles)  7.00     70
2                             Single breed  8.00     13
3                      Mixed breed/unknown  0.33     NA
4 Designer/deliberate mix (e.g., labradoodles)  6.00     24
5                      Mixed breed/unknown  6.00     45
6                      Mixed breed/unknown 14.00     85
```

(a) Missing Weight

```
dogs <- dogs %>%
  mutate(R = ifelse(is.na(Weight), 0, 1))
```

```
## total missing observations
sum(dogs$R == 0)
```

```
[1] 14
```

```
## total missing observations by gender
sum(dogs$R ==0 & dogs$Gender == "Male")
```

```
[1] 9
```

```
sum(dogs$R ==0 & dogs$Gender == "Female")
```

```
[1] 5
```

```
## total missing observations by fixed
sum(dogs$R ==0 & dogs$Fixed == "Yes")
```

```
[1] 7
```

```
sum(dogs$R ==0 & dogs$Fixed == "No")
```

```
[1] 7
```

There are a total of 14 missing observations. There are more missing observations among males than there are among females. Furthermore, the number of missing observations are equal among dogs that were fixed and those that were not fixed.

(b) Missing Values Among Groups

```
mean_weight <- mean(dogs$Weight, na.rm = TRUE)
mean_weight
```

```
[1] 45.91616
```

```
mean(dogs[dogs$Fixed == "Yes", "Weight"], na.rm = TRUE)
```

```
[1] 45.91368
```

```
mean(dogs[dogs$Fixed == "No", "Weight"], na.rm = TRUE)
```

```
[1] 45.97105
```

Since both groups feature the same number of missing values, the estimate should neither be an overestimate or an underestimate when conditioning on whether a dog is fixed or not.

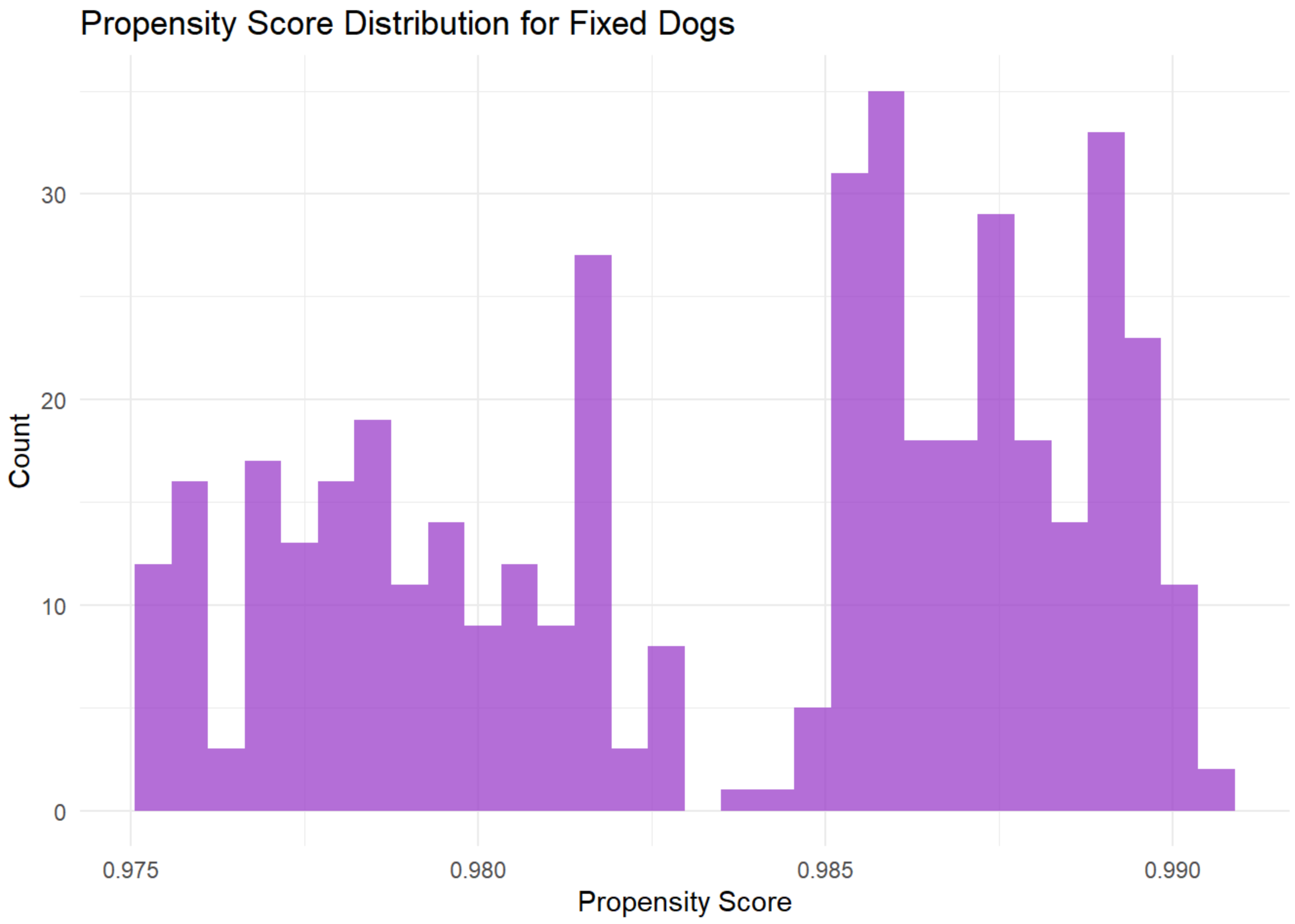
(c) Propensity Scores

```
dogs <- dogs %>%
  mutate(Gender = recode(Gender, "Female" = 0, "Male" = 1)) %>%
  mutate(Fixed = recode(Fixed, "No" = 0, "Yes" = 1))
```

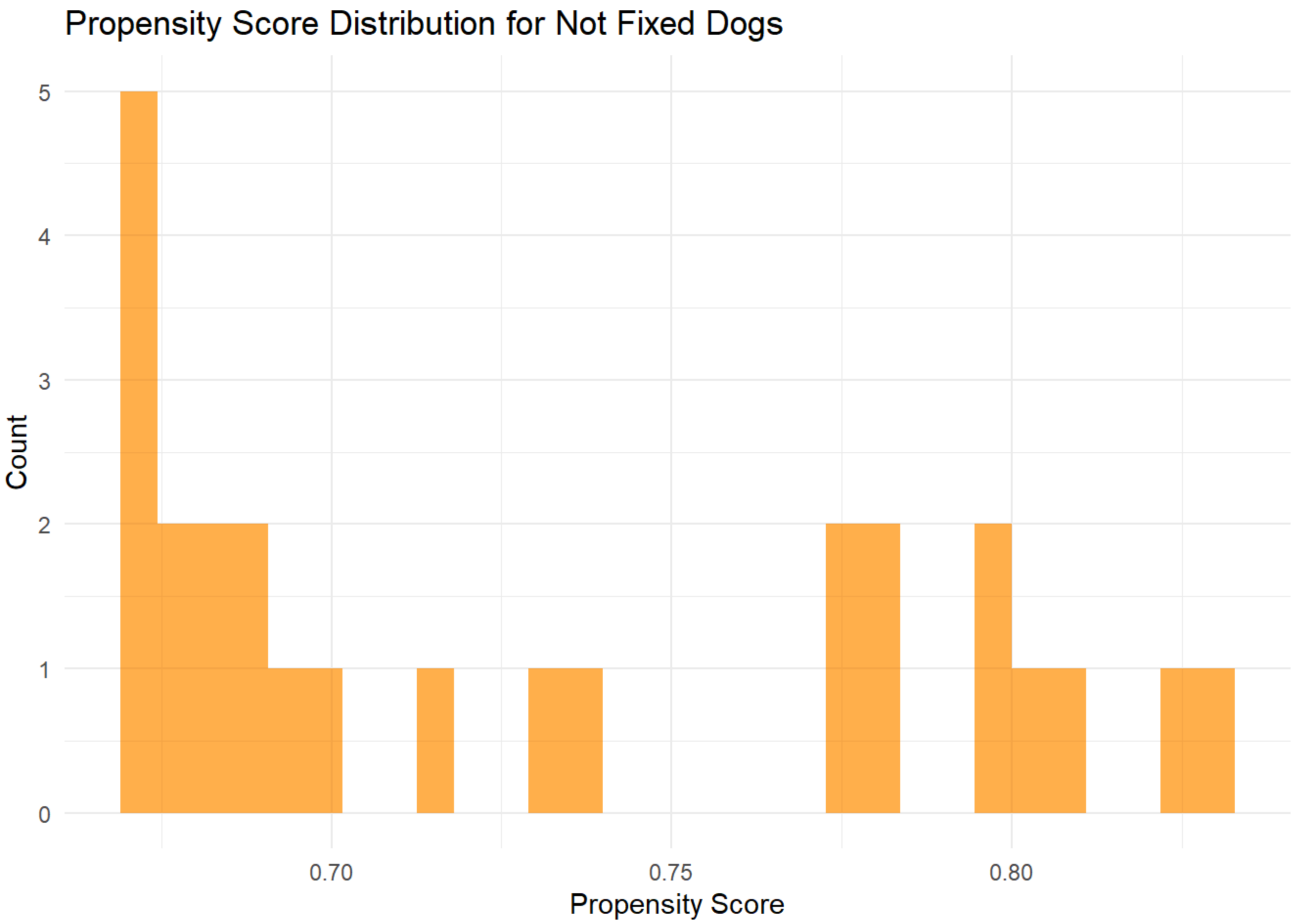
```
model <- glm(R ~ Fixed + Gender + Age,
             family = binomial(link = "logit"),
             data = dogs)

dogs_ipw <- augment_columns(model, dogs,
                            type.predict = "response") %>%
  rename(propensity = .fitted) %>%
  mutate(ipw = (R / propensity) + ((1 - R) / (1 - propensity)))

head(dogs_ipw)
```

```
# A tibble: 6 × 17
    Obs Name   Gender Fixed Color Heritage   Age Weight     R propensity .se.fit
  <int> <chr>   <dbl> <dbl> <chr> <chr>    <dbl>  <dbl> <dbl>      <dbl>   <dbl>
1     1 "Max"       1     1 Dark… Designe…  7        70     1      0.979 0.00951
2     2 "Isla"      0     1 Black Single …  8        13     1      0.988 0.00629
3     3 "Tyso…      1     0 Black Mixed b…  0.33     NA     0      0.673 0.117
4     4 "Lexi"      0     1 Ligh… Designe…  6        24     1      0.987 0.00675
5     5 ""          1     1 Ligh… Mixed b…  6        45     1      0.978 0.00974
6     6 "Lola"      0     1 Black Mixed b… 14        85     1      0.989 0.00711
# ℹ 6 more variables: .resid <dbl>, .hat <dbl>, .sigma <dbl>, .cooksd <dbl>,
#   .std.resid <dbl>, ipw <dbl>
```

```
ggplot(dogs_ipw[dogs_ipw$Fixed == 1, ], aes(x = propensity)) +
  geom_histogram(fill = "darkorchid", bins = 30, alpha = 0.7) +
  labs(title = "Propensity Score Distribution for Fixed Dogs",
       x = "Propensity Score",
       y = "Count") +
  theme_minimal()
```



```
ggplot(dogs_ipw[dogs_ipw$Fixed == 0, ], aes(x = propensity)) +
  geom_histogram(fill = "darkorange", bins = 30, alpha = 0.7) +
  labs(title = "Propensity Score Distribution for Not Fixed Dogs",
       x = "Propensity Score",
       y = "Count") +
  theme_minimal()
```



Yes, the propensity score distributions are different among fixed and not fixed dogs. Fixed dogs feature propensity scores above 0.975 while not fixed dogs have propensity scores below 0.86.

```
summary(dogs_ipw$propensity)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.6722  0.9784  0.9852  0.9692  0.9876  0.9904 
```

The propensity scores across all groups have a propensity score between 0 and 1.

(d) Horvitz-Thomson Estimator

```
dogs_ipw <- dogs_ipw %>%
  mutate(ipw1 = (R / propensity),
         ipw0 = (1 - R)/(1 - propensity))

dogs_ipw2 <- dogs_ipw %>%
  filter(propensity > 0.05 & propensity < 0.95)

ATE <- mean(dogs_ipw2$Weight * dogs_ipw2$ipw1) - mean(dogs_ipw2$Weight * dogs_ipw2$ipw0)
ATE
```

```
[1] NA
```

```
mean(dogs_ipw$Weight * dogs_ipw$ipw1) - mean(dogs_ipw$Weight * dogs_ipw$ipw0)
```

```
[1] NA
```

Using the Horvitz-Thompson estimator, the calculated mean should be similar to the mean calculated earlier because both groups feature the same amount of missing values. That said, for some reason, my estimator is not working and it is giving me an NaN value. This could be due to a cleaning error. Alternatively, fixed dogs only feature propensity scores above 0.95, according to the histograms in the previous question. Since this estimator requires removing these values, it will also not be an equitable comparison to the not fixed group.