Artist Project 4

Diana

February 17, 2019

This time I am going to create a bar chart that compares the distribution of gender given the different ethnicities

Step 1: Set working directory, load libraries, and read the .csv file

```
## [1] "C:/Users/diana/DataViz"
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
## -- Attaching packages ---------------------------------------------------- tidyverse 1.2.1 --
```

```
## v tibble  1.4.2     v purrr   0.2.5
## v tidyr   0.8.1     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Step 2: Omit NA values and create a dataframe to store the ethnicity and gender info along with getting rid of duplicates

```
dist <- artist %>% distinct(artist, .keep_all = TRUE)
# Get counts for gender for each ethnicity:
dist_gender<- dist %>% group_by(ethnicity, gender) %>% summarize(count=n()) %>% na.omit()


dist_gender
```

```
## # A tibble: 10 x 3
## # Groups:   ethnicity [5]
##    ethnicity gender count
##    <fct>     <fct>  <int>
##  1 asian     man      510
##  2 asian     woman     35
##  3 black     man       62
##  4 black     woman     26
```

```
##  5 hispanic  man      173
##  6 hispanic  woman     20
##  7 other     man       61
##  8 other     woman     29
##  9 white     man     5121
## 10 white     woman    732
```

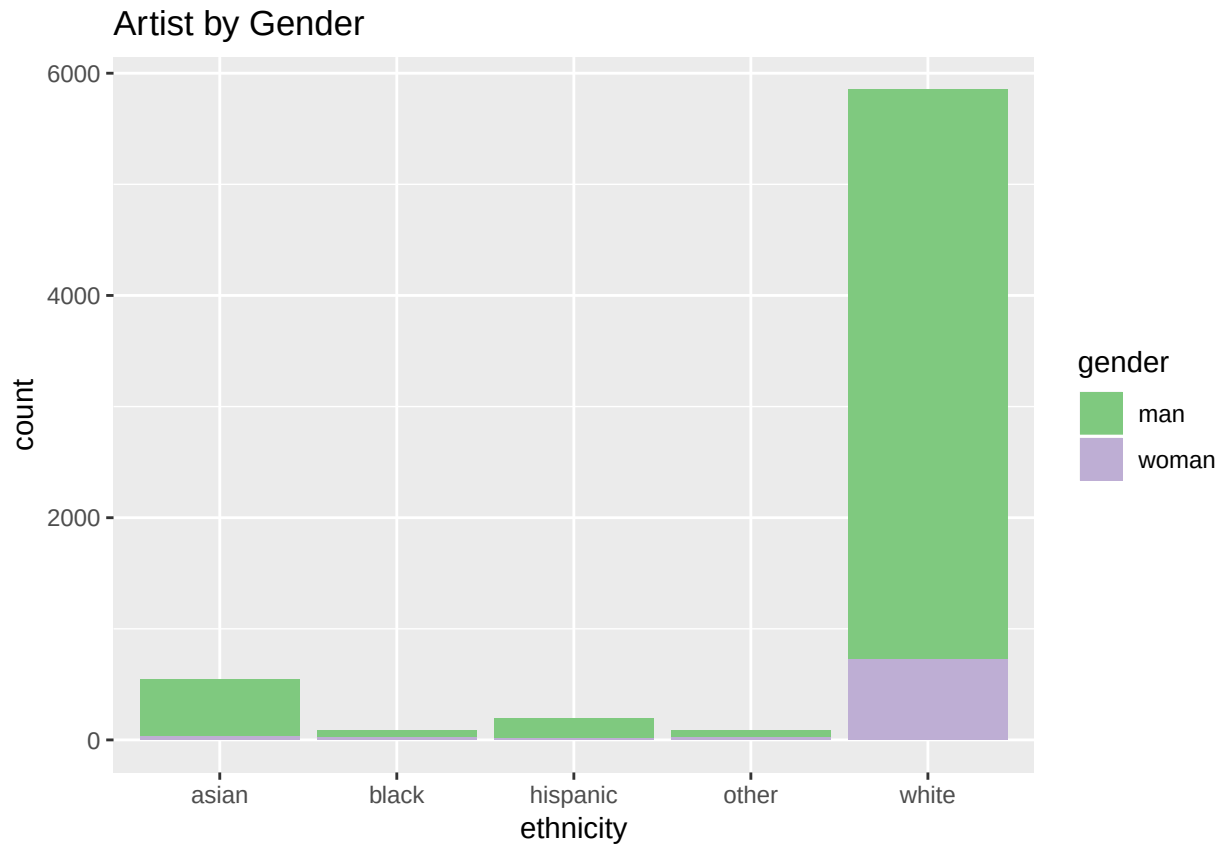
```
library(RColorBrewer)
mycols <- brewer.pal(8, "Accent")[1:5]
mycols  #hex code for colors
```

```
## [1] "#7FC97F" "#BEAED4" "#FDC086" "#FFFF99" "#386CB0"
```

```
p <- scale_fill_manual(values=mycols)
```

Step 3: Create a simple barchart

```
ggplot(dist_gender, aes(x=ethnicity, y=count)) + geom_bar(aes(fill=gender), stat="identity") + p + ggti
```



Step 4 : Create a proportion table

```
counttable <- xtabs(count~ethnicity + gender, data=dist_gender)
counttable
```

```
##           gender
## ethnicity   man woman
##   asian     510    35
##   black      62    26
##   hispanic  173    20
##   other      61    29
##   white    5121   732
```

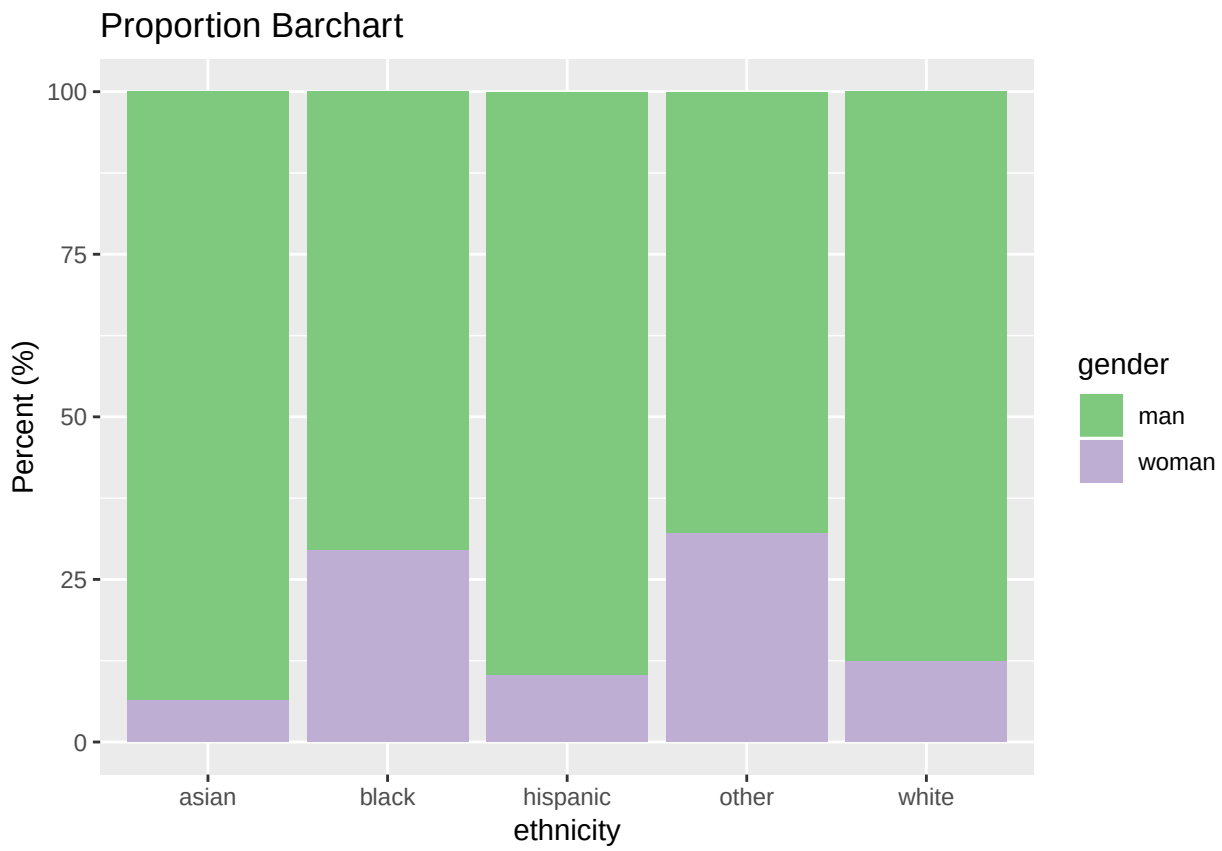
```
# find conditional proportions:
prop.table <- prop.table(counttable, 1)
prop.table
```

```
##           gender
## ethnicity         man      woman
##   asian    0.93577982 0.06422018
##   black    0.70454545 0.29545455
##   hispanic 0.89637306 0.10362694
##   other    0.67777778 0.32222222
##   white    0.87493593 0.12506407
```

```
# recast the table back into a data frame:
plotdata <- as.data.frame(prop.table)
plotdata
```

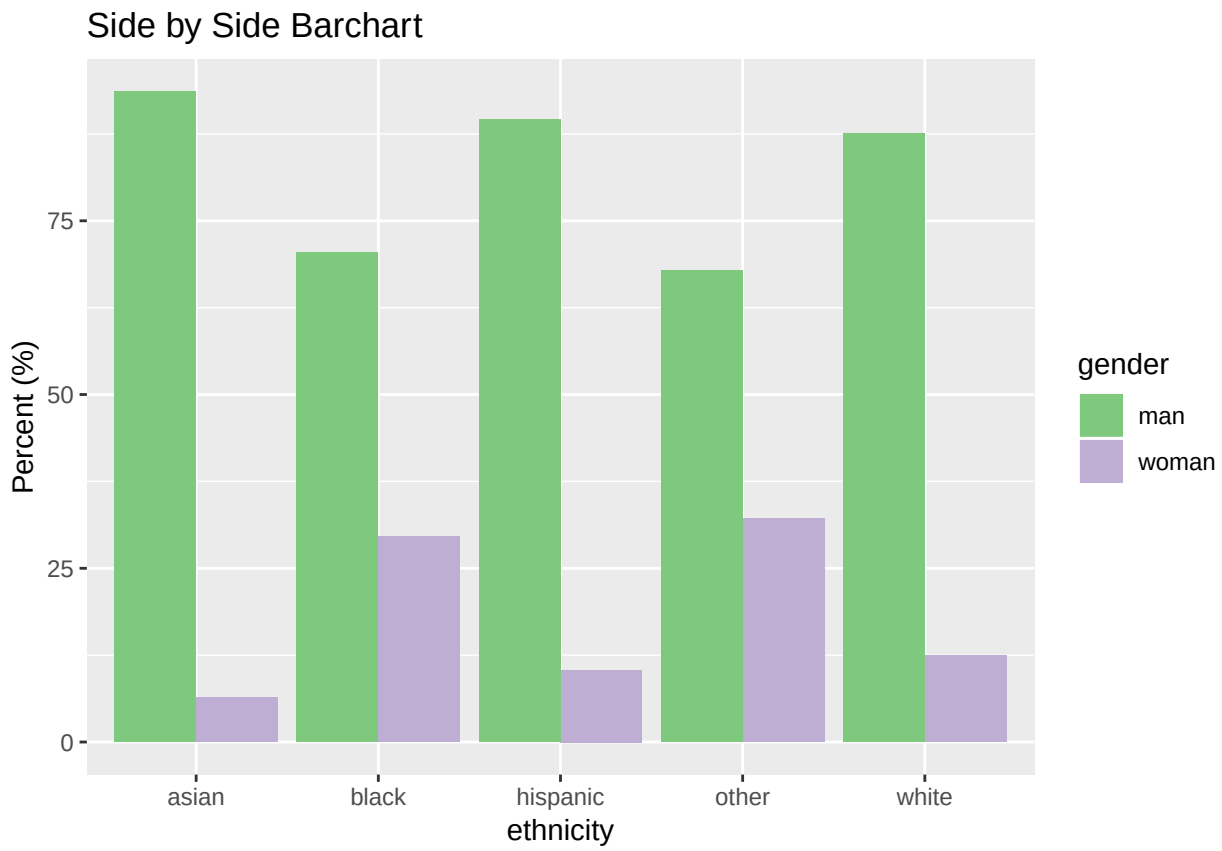
```
##    ethnicity gender       Freq
## 1      asian    man 0.93577982
## 2      black    man 0.70454545
## 3   hispanic    man 0.89637306
## 4      other    man 0.67777778
## 5      white    man 0.87493593
## 6      asian  woman 0.06422018
## 7      black  woman 0.29545455
## 8   hispanic  woman 0.10362694
## 9      other  woman 0.32222222
## 10     white  woman 0.12506407
```

```
ggplot(plotdata, aes(x=ethnicity, y=100*Freq)) +
  geom_bar(aes(fill=gender), stat="identity") +
  ylab("Percent (%)") + p +ggtitle("Proportion Barchart")
```



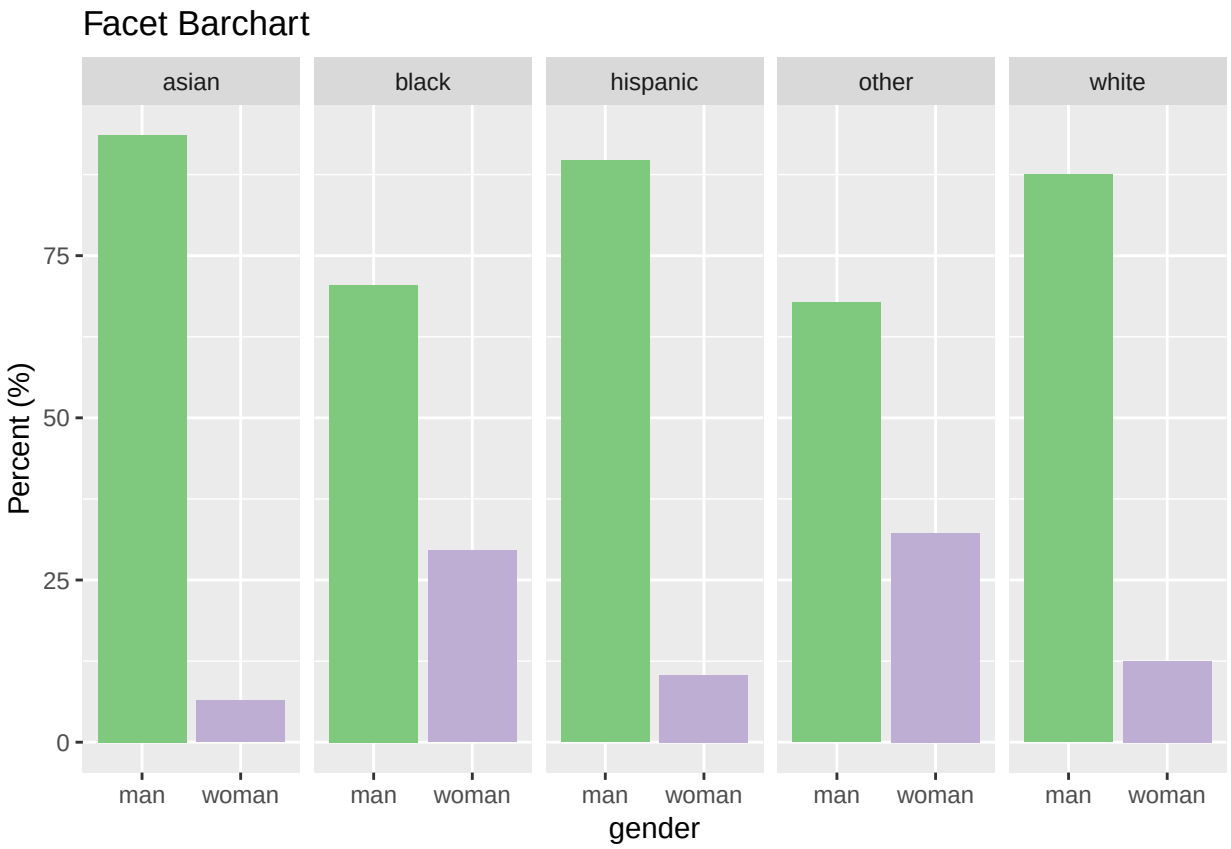
Step 4 : Create a side by side barchart

```
ggplot(plotdata, aes(x=ethnicity, y=100*Freq)) +
  geom_bar(aes(fill=gender), stat="identity", position="dodge") +
  ylab("Percent (%)") + p + ggtitle("Side by Side Barchart")
```



Step 5 : Create facet barchart (for each ethnic)

```
ggplot(plotdata, aes(x=gender, y=100*Freq, fill=gender)) +
  geom_bar(stat="identity", position="dodge") +
  ylab("Percent (%)") +
  facet_wrap(vars(ethnicity), ncol=5) +
  guides(fill="none") + p + ggtitle("Facet Barchart")
```


Facet Barchart
asian
black
hispanic
other
white
Percent (%)
75
50
25
0
man
woman
gender