

Question 1 [50 points]

```
data(midwest)
midwest_modified<-midwest %>% select(county,state,popdensity,
                                     popwhite,popblack,
                                     popamerindian,popasian,
                                     popother,inmetro)
```

The data for this question comes from a modified version of the `midwest` dataset from the `ggplot` library.

```
str(midwest_modified)
```

```
tbl_df [437 x 9] (S3: tbl_df/tbl/data.frame)
 $ county      : chr [1:437] "ADAMS" "ALEXANDER" "BOND" "BOONE" ...
 $ state       : chr [1:437] "IL" "IL" "IL" "IL" ...
 $ popdensity  : num [1:437] 1271 759 681 1812 324 ...
 $ popwhite    : int [1:437] 63917 7054 14477 29344 5264 35157 5298 16519 13384 146506 ...
 $ popblack    : int [1:437] 1702 3496 429 127 547 50 1 111 16 16559 ...
 $ popamerindian: int [1:437] 98 19 35 46 14 65 8 30 8 331 ...
 $ popasian    : int [1:437] 249 48 16 150 5 195 15 61 23 8033 ...
 $ popother    : int [1:437] 124 9 34 1139 6 221 0 84 6 1596 ...
 $ inmetro     : int [1:437] 0 0 0 1 0 0 0 0 0 1 ...
```

```
midwest_modified %>% slice(1:5) %>%
  select(county:popblack)
```

```
# A tibble: 5 x 5
  county    state popdensity popwhite popblack
  <chr>    <chr>      <dbl>    <int>    <int>
1 ADAMS    IL          1271.    63917    1702
2 ALEXANDER IL           759     7054    3496
3 BOND     IL           681.    14477     429
4 BOONE    IL          1812.    29344     127
5 BROWN    IL           324.     5264     547
```

```
midwest_modified %>% slice(1:5) %>%
  select(county,popamerindian:popother)
```

```
# A tibble: 5 x 4
  county    popamerindian popasian popother
  <chr>          <int>    <int>    <int>
1 ADAMS           98      249     124
2 ALEXANDER        19      48      9
3 BOND            35      16     34
4 BOONE           46     150    1139
5 BROWN           14       5      6
```

The dataset contains population data from midwest counties in five states in the United States from an unspecified year. There are identifying variables for both the `county` (the name) and the `state` (the postal abbreviation). The variable `popdensity` is a measure of density (population per unspecified area units). The variable `inmetro` is equal to 1 if the county is classified as a metropolitan area and 0 otherwise. The other variables contain counts of population size within self-identified racial classifications.

CONTINUED ON NEXT PAGE

- (a) [5 pts] Write a line of code that will generate the following `tibble` (or `data.frame`) containing the highest population density from each state:

```
# A tibble: 5 x 2
  state Highest_Pop_Den
  <chr>         <dbl>
1 IL           88018.
2 IN           34659.
3 MI           60334.
4 OH           54313.
5 WI           63952.
```

- (b) [5 pts] Write a line of code that adds a new column to the `midwest_modified` `tibble` called `Metro` where the elements of that column are equal to a string “Metro” if `inmetro` is equal to 1 and “NonMetro” if `inmetro` is equal to 0. The first five rows are given below for the `county`, `state`, `inmetro` and `Metro` columns:

```
# A tibble: 5 x 4
  county    state inmetro Metro
  <chr>    <chr>   <int> <chr>
1 ADAMS    IL         0 NonMetro
2 ALEXANDER IL         0 NonMetro
3 BOND     IL         0 NonMetro
4 BOONE    IL         1 Metro
5 BROWN    IL         0 NonMetro
```

- (c) [5 pts] Write a line of code that will generate the following `tibble` (or `data.frame`) containing the highest population density from each state for metropolitan and non-metropolitan counties separately, using the modified `tibble` from part (b).

```
dens_table
```

```
# A tibble: 10 x 3
# Groups:   state [5]
  state Metro Highest_Pop_Den
  <chr> <chr>         <dbl>
1 IL    Metro       88018.
2 IL    NonMetro    2309.
3 IN    Metro      34659.
4 IN    NonMetro    3090.
5 MI    Metro      60334.
6 MI    NonMetro    2251.
7 OH    Metro      54313.
8 OH    NonMetro    5484.
9 WI    Metro      63952.
10 WI   NonMetro    2344.
```

CONTINUED ON NEXT PAGE

- (d) [5 pts] Assume the tibble from part (c) is called `dens_table` as above. Now write a line of code that produces a tibble which arranges the data above so that we have separate columns for “Metro” and “NonMetro”, as below:

```
# A tibble: 5 x 3
# Groups:   state [5]
  state Metro NonMetro
  <chr>  <dbl>   <dbl>
1 IL    88018.   2309.
2 IN    34659.   3090.
3 MI    60334.   2251.
4 OH    54313.   5484.
5 WI    63952.   2344.
```

Now we will work with only a modified version of the population counts for each county.

- (e) [5 pts] Write a line of code to add a new variable to the data frame named `HighDens` which is equal to “High” if the population density for the county is higher than 1500 and “Not High” if the population density for the county is lower than 1500. Below are the first 5 rows of the data for the `county`, `popdensity` and `HighDens` columns:

```
# A tibble: 5 x 3
  county popdensity HighDens
  <chr>      <dbl> <chr>
1 ADAMS      1271. NotHigh
2 ALEXANDER    759 NotHigh
3 BOND        681. NotHigh
4 BOONE      1812. High
5 BROWN       324. NotHigh
```

Then we will compute the total number of people in each combination of `state`, `inmetro` and `HighDens` using the code below:

```
pop_xtabs<-xtabs(
  I(popwhite+popblack+popamerindian+popasian+popother)~
    state+Metro+HighDens,data=midwest_modified)
pop_xtabs
```

```
, , HighDens = High
```

```
      Metro
state  Metro NonMetro
IL  9323624  405933
IN  3728008  689565
MI  7697643  354081
OH  8811604  1078957
WI  3004347  386892
```

```
, , HighDens = NotHigh
```

```
      Metro
state  Metro NonMetro
IL   250175  1450870
IN   234438   892148
MI         0  1243573
OH    98555   857999
WI   326825  1173705
```

CONTINUED ON NEXT PAGE

(f) [5 pts] What will the code `pop_xtabs["IL",1,2]` return as output?

(g) [5 pts] Using only the `pop_xtabs` object above, write a line of code to find the total number of people in areas high density (i.e. `HighDens` is “High”) as below:

```
High  NotHigh
35480654  6528288
```

(h) [10 pts] Using only the `pop_xtabs` object above, write a line of code that computes the total population in the combination of `State` and `HighDens` to return the output below:

```
HighDens
state    High NotHigh
IL 9729557 1701045
IN 4417573 1126586
MI 8051724 1243573
OH 9890561  956554
WI 3391239 1500530
```

(i) [5 pts] Using only the `pop_xtabs` object above, write a line of code (or multiple lines of code) that computes the percentage of individuals in `High` and `Low` density in each state as below:

```
HighDens
state    High    NotHigh
IL 85.11850 14.881500
IN 79.67977 20.320233
MI 86.62148 13.378518
OH 91.18149  8.818511
WI 69.32541 30.674588
```

END OF QUESTION 1