Addendum: Introduction to Data

4. Filtering and Summarizing Data

Let's suppose that you are interested in understanding summary statistics for only counties with an economic specialization in recreation. There are several ways that we can filter the data to focus our analysis on this sub-sample.

4.1 Create Sub-Sample Data Frame

First, we can filter the data and save it as a separate data set. In the code that follows, we filter out the observations where `EconTopology` is equal to `"Recreation"` and save it to a new data frame we call `rent.rec.df`.

```
rent.rec.df <- filter(countyrent.df, EconTopology=="Recreation")
```

To test for equality, we use the `==` operator, as in `EconTopology=="Recreation"`.

Now we can look at descriptive statistics for some of our variables:

```
summary(rent.rec.df$MedianRent)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   782.9   928.2  1241.7  1320.0  1436.5  3156.3
```

```
summary(rent.rec.df$UrbanInfluence)
```

```
##          In large metro area of 1+ million residents
##                                                   10
## In small metro area of less than 1 million residents
##                                                   16
##       Micropolitan area adjacent to large metro area
##                                                    1
##       Micropolitan area not adjacent to a metro area
##                                                    0
```

4.2 Filter and Pipe

Instead of saving a separate data frame (like `rent.rec.df` above), we can filter our data frame, then *pipe* the result to another function. The pipe operator is given by `%>%`. What is does is it takes the result of the function call that comes before the operator and passes that as an input to the function call that comes after the operator.

In the example below, we filter out counties with an economic specialization in recreation, then we use the `summarise()` function to calculate the means for `MedianRent` and `PovertyPerc`, saving these values as `MeanRent` and `MeanPoverty`

```
filter(countyrent.df, EconTopology=="Recreation") %>%
  summarise(MeanRent=mean(MedianRent), MeanPoverty=mean(PovertyPerc))
```

```
##   MeanRent MeanPoverty
## 1  1319.96    12.32593
```

4.3 Group and Pipe

Perhaps we are interested in counties that specialize in recreation, but we would also like to compare those results to other types of counties. Instead of filtering, we can group data by `EconTopology`, and calculate summary statistics for each group.

In the code below, we compute summary statistics for `MedianRent` for each type of county as defined in `EconTopology`. We again use pipes (`%>%` operator) to pipe the data frame to the `group_by()` function, then pipe that result to the summary function.

```
countyrent.df %>%
  group_by(EconTopology) %>%
  summarise(MeanRent=mean(MedianRent))
```

```
## # A tibble: 5 x 2
##                   EconTopology MeanRent
##                          <fctr>    <dbl>
## 1              Nonspecialized 1183.925
## 2            Mining dependent 1105.758
## 3     Manufacturing dependent 1151.169
## 4 Federal/State Gov dependent 1062.537
## 5                  Recreation 1319.960
```

Now we can compare average rent in counties specializing in recreation to other types of economic specialization. We can see from the table above, that (at least in our sample) counties that specialize in recreation have on average higher rent than other types of counties.

We will conclude with making a bar chart to visualize the results above. The code below uses the `ggplot()` function. The code below to create the chart is beyond the scope of this tutorial, so it is okay if you do not understand it.

```
# First create the summary statistics, save result in renttable
countyrent.df %>%
  group_by(EconTopology) %>%
  summarise(MeanRent=mean(MedianRent)) ->
  renttable

# Call ggplot() with data frame renttable
ggplot(renttable, aes(x=EconTopology, y=MeanRent)) + geom_col()
```