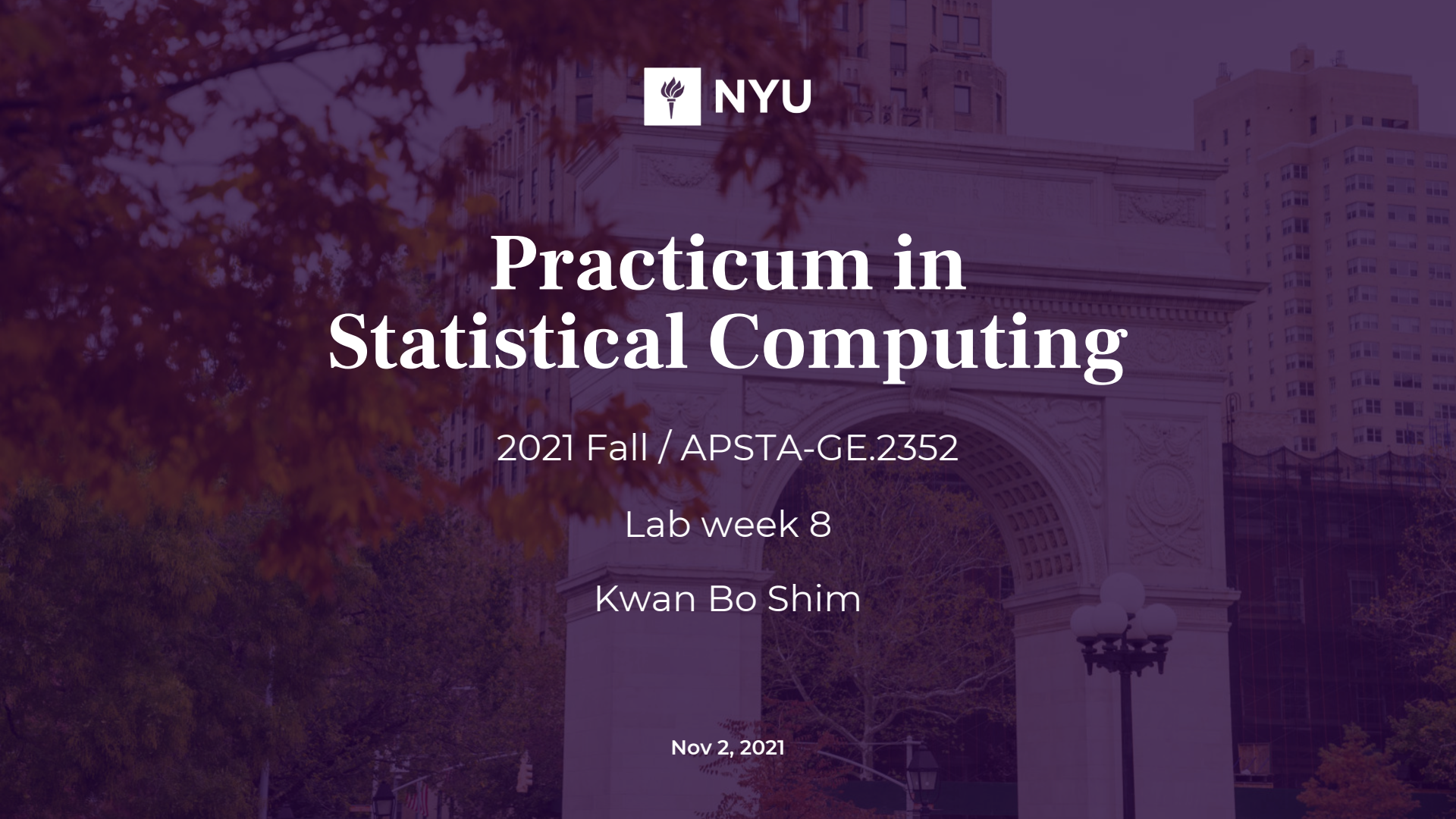


NYU

Practicum in Statistical Computing

2021 Fall / APSTA-GE.2352

Lab week 8

Kwan Bo Shim

Nov 2, 2021

PART 01

Week 8

PART 01

Week 8

1. Poll -> end of the class
2. Attendance
3. Lab_report 2 due date

NYU

PART 01

Week 8

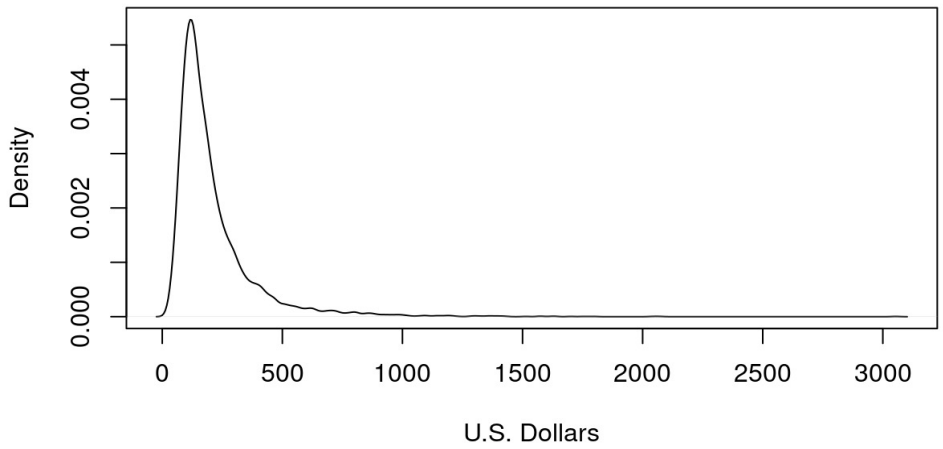
Exploration of multivariate data

- **VLSS** (Vietnam Living Standards Survey)

NYU

Week 8

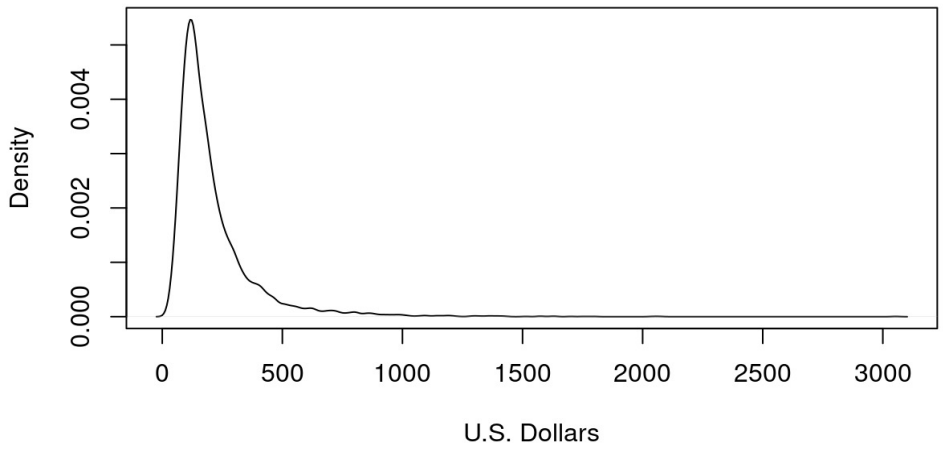
```
# Now we are familiar (bored) with this plot
plot(density(household$Dollars), xlab = "U.S.
Dollars", main = " ")
```



NYU

Week 8

```
# Now we are familiar (bored) with this plot
plot(density(household$Dollars), xlab = "U.S.
Dollars", main = " ")
# How about plotting conditionally?
```



NYU

PART 02

Indexing

Indexing

- An element of a matrix or data frame is accessed using the bracket notation, `x[i,j]` where x is the name of the matrix or data frame, i is the row number, and j is the column number.

```
# access the first housefold's per capita expenditure
household[1, 1]
# access all the variables for the first household
household[1, ]
# access every household's per capita expenditure
household[ , 1]
# access the first household's per capita expenditure
household$Dollars[1]
```

```
household[ , 1]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ , 2]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ , 3]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ 1 , ]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ 2 , ]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ 3 , ]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
household[ 3 , 3]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

household$Dollars[1]

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

household$Region[10]

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
# how about
household[,3][3]
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
# how about
household[1,1][1] ?
```

	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

```
# how about
household[7,12][4] ?
```

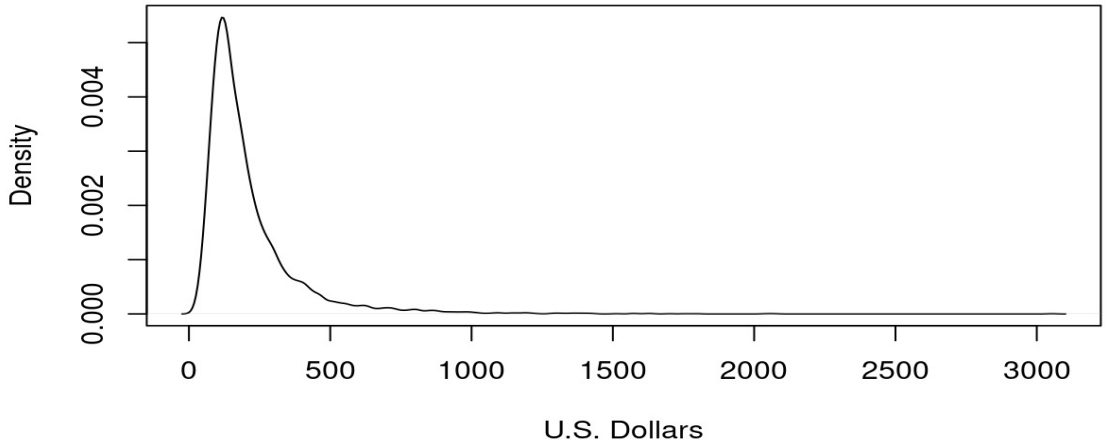
	Dollars <dbl>	Area <chr>	Region <int>
1	184.33	Rural	5
2	62.73	Rural	5
3	119.13	Rural	5
4	76.61	Rural	5
5	97.46	Rural	5
6	132.09	Rural	5
7	99.86	Rural	5
8	79.97	Rural	5
9	88.60	Rural	5
10	161.03	Rural	5

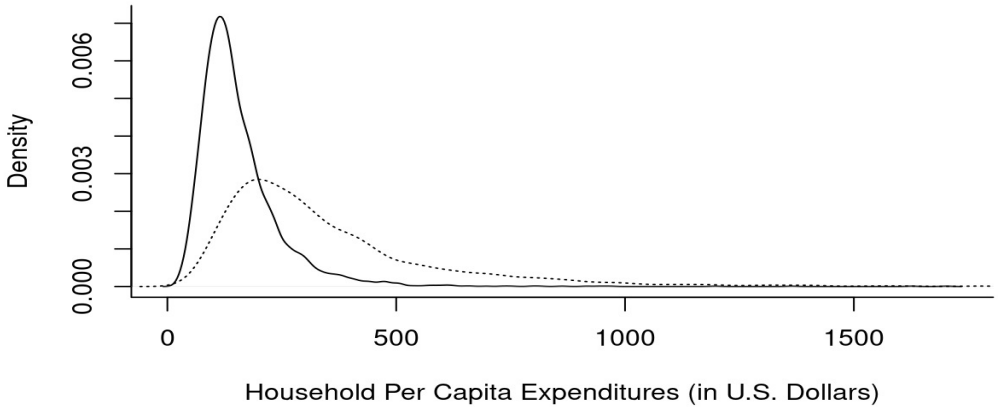
Indexing using logical expression

- The logical expression in the indexing is essentially asking if the value of **household$Area** “is equal to” **Rural**. When the lefthand side is a vector, it evaluates this question for each element sequentially, returning a logical (boolean) vector.

```
household$Dollars[household$Area == "Rural"]
```

```
household$Dollars[household$Area == "Urban"]
```


Density
0.000
0.002
0.004
0
500
1000
1500
2000
2500
3000
U.S. Dollars


Density
0.000
0.003
0.006
0
500
1000
1500
Household Per Capita Expenditures (in U.S. Dollars)

NYU

PART 02

Numerical summary

mean, median

```
# marginal median household per capita expenditure
median(household$Dollars)

# marginal mean household per capita expenditure
mean(household$Dollars)

# Cannot compute conditionally by themselves
```

conditional computation

```
tapply(X = household$Dollars,
       INDEX = household$Area,
       FUN = mean)
```

```
-> results
Rural    Urban
157.4192 348.6887
```

Standard error of the mean

- Another measure of variation that often gets reported in the educational and behavioral sciences is the ***standard error of the mean***, which represents sampling variability of the sample mean estimator, and is computed as follows (recall the Central Limit Theorem):

- **SE = SD / sqrt(N)**

Skewness

- Skewness is a numerical measure that helps summarize a distribution's departure from symmetry about its mean.
- A complete symmetric distribution has a skewness value of zero.
- The skewness for a normal distribution is zero, and any symmetric data should have a skewness near zero.
- The e1071 package provides a function called skewness(), which computes the skewness value for a sample distribution based on three common algorithms.
- This function is supplied with the argument type=2 to compute G1, a slightly modified version of skewness that is a better population estimate.
- G1 is the adjusted Fisher-Pearson coefficient of skewness

Skewness

$$G_1 = \frac{\sqrt{N(N-1)}}{N-2} \frac{\sum_{i=1}^{N} (Y_i - \bar{Y})^3 / N}{s^3}$$

Source: https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm

Skewness

```
# load e1071 package
install.packages('e1071')
library(e1071)

# skewness measure for the marginal distribution of
household per capita expenditure
skewness(household$Dollars, type = 2)

# skewness measure for the distribution of household per
capita expenditure conditioned on area
tapply(X = household$Dollars, INDEX = household$Area,
FUN = skewness, type = 2)
```

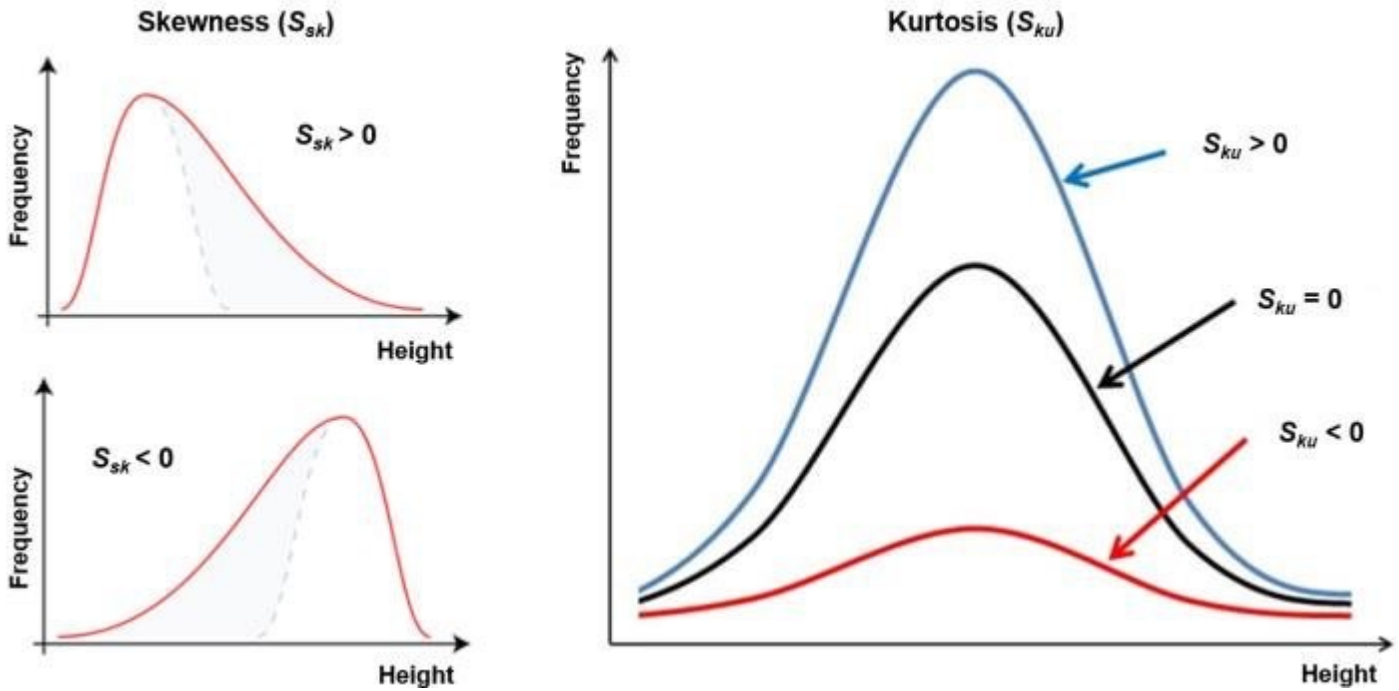
Kurtois

- Kurtosis is often used as a numerical summarization of the "peakedness" of a distribution, referring to the relative concentration of scores in the center, tail, and shoulders.

```
# Kurtosis measure for the marginal distribution of
household per capita expenditure
kurtosis(household$Dollars, type = 2)

# Kurtosis for the distribution of household per capita
expenditure conditioned on area
tapply(X = household$Dollars, INDEX = household$Area,
FUN = kurtosis, type = 2)
```

Skewness & Kurtois



NYU

Image source: https://www.researchgate.net/figure/Illustration-of-the-skewness-and-kurtosis-values-and-how-they-correlate-with-the-shape-of_fig1_298415862

PART 03

Publication

Plots for publication

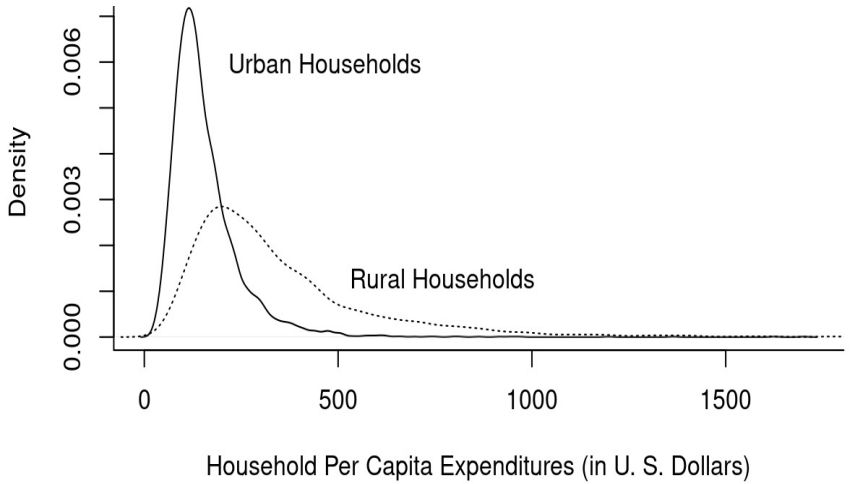
- The book mentions that the APA manual (American Psychological Association, 2009) provides suggestions for presenting descriptive statistics for groups of individuals.
- The suggestion is that information should be presented in the text when there are three or fewer groups and in a table when there are more than three groups.
- There are many statisticians who prefer visual representations in most situations.
- *You must remain flexible.*

The **text() function** is used to add text to an existing plot.

```
d.rural <- density(rural.households)
d.urban <- density(urban.households)

plot(d.rural, main = " ", xlab = "Household
Per Capita Expenditures (in U. S.
Dollars)", lty = "solid", bty = "l")
lines(d.urban, lty = "dotted")

text(x = 182, y = 0.0059, labels = "Urban
Households", pos = 4)
text(x = 495, y = 0.0012, labels = "Rural
Households", pos = 4)
```

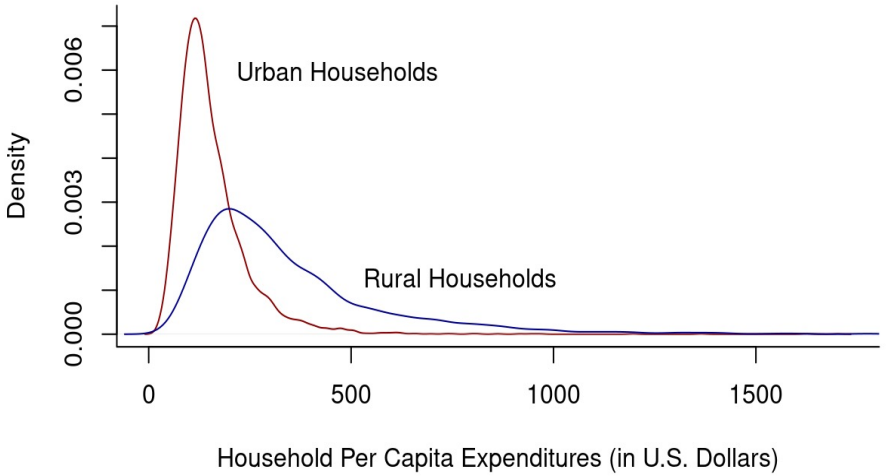


- One way to specify color is to use the RGB color model
- RGB colors are specified using the **rgb() function**.
- This function requires a numerical value for each of the three arguments, red=, green=, and blue=.

```
d.rural <- density(rural.households)
d.urban <- density(urban.households)

plot(d.rural, main = " ", xlab = "Household Per
Capita Expenditures (in U.S. Dollars)", bty = "l",
col = rgb(red = 139, green = 0, blue = 0,
maxColorValue = 255), lty = "solid")
lines(d.urban, col = rgb(red = 0, green = 0, blue =
139, maxColorValue = 255), lty = "solid")

text(x = 182, y = 0.0059, labels = "Urban
Households", pos = 4)
text(x = 495, y = 0.0012, labels = "Rural
Households", pos = 4)
```



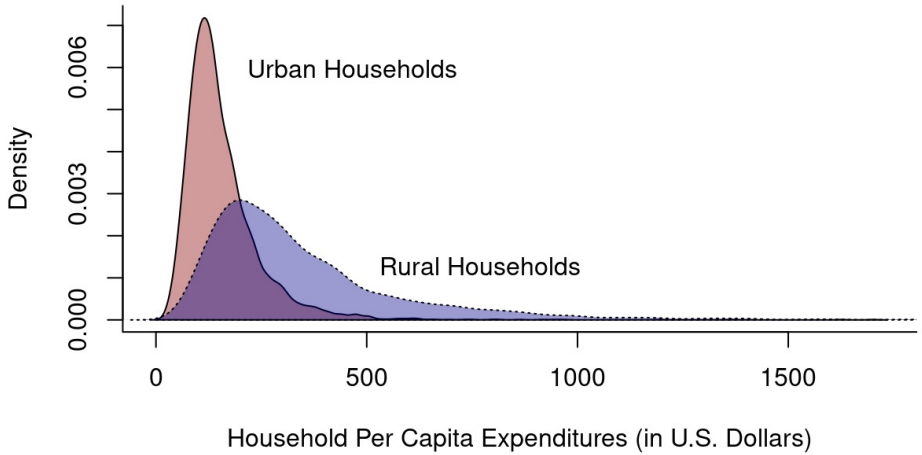
NYU

- The **polygon() function** can be used to shade the densities being plotted.

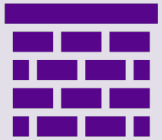
```
plot(d.rural, main = " ", xlab = "Household Per
Capita Expenditures (in U.S. Dollars)", bty =
"l", type = "n")

polygon(d.rural, col = rgb(red = 139, green = 0,
blue = 0, alpha = 100, maxColorValue = 255), lty
= "solid")
polygon(d.urban, col = rgb(red = 0, green = 0,
blue = 139, alpha = 100, maxColorValue = 255),
lty = "dotted")

text(x = 182, y = 0.0059, labels = "Urban
Households", pos = 4)
text(x = 495, y = 0.0012, labels = "Rural
Households", pos = 4)
```



PART 03



Robust estimate

Robust estimation

- The sample mean, variance, and standard deviation can be inordinately influenced by outliers that may be present in the sample data.
- Because of this, in some distributions - such as skewed distributions - the sample mean and variance are not good representatives of the typical score and variation in the population.

- **Robust estimates** reduce the effects of the tails of a sample distribution and outliers by either trimming or recoding the distribution before the estimates are computed. An advantage of a robust estimate is that its associated standard error will typically be smaller than its conventional counterpart.

Trimming

- One strategy for reducing the effects of the tails of a distribution is simply to remove them.

```
# marginal 20% trimmed mean
mean(household$Dollars, tr = 0.2)

# conditional 20% trimmed means
tapply(X = household$Dollars,
       INDEX = household$Area,
       FUN = mean, tr=.2)
```

Winsorization

- It is also possible to compute a robust estimate for the variation in a data set.
- A robust estimate of variation can be obtained by recoding extreme observations to be less extreme. This recoding is known as Winsorizing.

```
# install and load package
Install.packages('WRS2')
Library(WRS2)

# marginal 20% Winsorized variance
winvar(household$Dollars, tr = 0.2)

# conditional 20% Winsorized variances
tapply(X = household$Dollars,
       INDEX = household$Area,
       FUN = winvar, tr=.2)
```

Winsorization

```
# install and load package
library(WRS2)

# example list
listx <- c(1,2,1,2,3,4,3,5,6,4,3,2,1,2,3,4,5,100,200,300)

# marginal 20% Winsorized variance
var(listx)
winvar(listx, tr = 0.2)
```

```
[1] 6263.103
[1] 1.621053
```

End of class Poll

We really appreciate the feedback!