LECTURE 3A – DESCRIPTIVE AND INFERENTIAL STATISTICS

Descriptive Statistics

See Book Chapters 2 and 7 + Additional Material

DESCRIPTIVE STATISTICS

See Book Chapter 2

Population and Sample

- A population is a well-defined collection of *measurements* on "*subjects*" (people, things, processes, etc.) about which you need to *make a conclusion*.
- Like a population, a **sample** represents measurements taken on a subset of a population (i.e., a subset of the population measurements).
  - Often, a sample is taken because obtaining measurements for the entire population may be tedious, expensive time-consuming or even infeasible. This is the perspective often used in **research** (agricultural, medical, social sciences, political science, etc.)
  - Sometimes, even a smaller sample may involve expensive and difficult data collection (such as in medicine)
- *Traditional (inferential statistics) involves learning (or inferring) about the population, based on a single sample of limited size.*
- With the advent of analytics on large secondary data sets, especially in business applications, the line between a sample and a population is blurred.

Descriptive Statistics vs Inferential Statistics

- A *statistic* is any quantity that is calculated from a *sample.*
  - If we consider the data below as a sample, and we calculate the average of income, that sample average becomes a statistic. So, do the sample standard deviation and the sample median.

- *Descriptive statistics* summarize sample data. The goal is to describe the data using measures such as its frequency distribution, its percentiles, measures of central tendency, dispersion, skewness etc. Sometimes the summary is in the form of graphs. We are also interested in identifying unusual data points or *outliers. There is no formal concept of probability; instead we use relative frequencies.*

- *Inferential statistics* uses statistical theory to draw *inferences (conclusions) about unknown population parameters*, and is used as a tool for decision making and policy development. In inferential statistics, each data point is considered the value of a random variable from an underlying population.

ID	Name	Age	Gender	Education	Credit Score	Income	Net Worth	Sales
001	Adams, John	36	M	HS	350	38,900	65,924	1,535
002	Ramesh, Jyoti	23	F	Bachelors	600	172,000	178,154	2,196
003	Mendez, Nick	67	M	Bachelors	700	218,000	265,209	1,287
004	Mendez, Joan	38	F	PhD	550	182,000	85,277	2,143
005	Ritter, Jake	24	M	Masters	625	434,000	193,760	707
006	Rao, Eric	61	M	PhD	770	82,000	314,953	2,170
007	Blake, Ann	26	F	HS	490	112,000	192,946	1,229
008	Bishop, Marge	44	F	Masters	540	242,000	339,705	520
009	Ahmed, Mo	31	M	Masters	680	111,000	185,767	2,326
010	Shultz, Dante	44	M	Bachelors	280	66,000	97,778	588

Descriptive Statistics – Numerical Data vs Grouped Data

- Sample data may be in the form of numeric data (such as Income) or in the form of grouped data (grouped by categories). The categories may be nominal (Gender) or ordinal (Education).
- We will look at:
  - Describing Numeric (Ungrouped) Data
  - Describing Grouped Data

ID	Name	Age	Gender	Education	Credit Score	Income	Net Worth	Sales
001	Adams, John	36	M	HS	350	38,900	65,924	1,535
002	Ramesh, Jyoti	23	F	Bachelors	600	172,000	178,154	2,196
003	Mendez, Nick	67	M	Bachelors	700	218,000	265,209	1,287
004	Mendez, Joan	38	F	PhD	550	182,000	85,277	2,143
005	Ritter, Jake	24	M	Masters	625	434,000	193,760	707
006	Rao, Eric	61	M	PhD	770	82,000	314,953	2,170
007	Blake, Ann	26	F	HS	490	112,000	192,946	1,229
008	Bishop, Marge	44	F	Masters	540	242,000	339,705	520
009	Ahmed, Mo	31	M	Masters	680	111,000	185,767	2,326
010	Shultz, Dante	44	M	Bachelors	280	66,000	97,778	588

DESCRIPTIVE STATISTICS

Numeric (Ungrouped) Data

Describing Numerical Data

- Numerical data may be described using:
  - Measures of Central Tendency (Sample Mean, Sample Median, Sample Mode)
  - Measures of Dispersion (Sample Variance, Sample Standard Deviation, Range)
  - Measures of Location (Percentiles, Quartiles, Inter-Quartile Range (IQR))
  - Measures of Shape (Skewness, Kurtosis)
- **Note**: The formulas for some of these measures may be different from the ones used for describing population random variables.
- The **R** code file is: DescriptiveStat.R

Practice with calculator

Numerical Data - Measures of Central Tendency (**DescriptiveStat.R**)

- Measures of center of the data include:
  - Sample Mean or Average
    - In a *random sample* of size n (meaning that every observation is equally likely), the sample mean of data is the numerical average

      $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$ where i =1,..,n – For Income
    - In **R**, *mean*(income) gives the sample mean of income
    - Note: $\overline{X}$ is the sample mean.

```
> # Mean of Income
> print(paste("The mean of Income is: ", round(mean(income), 2)))
[1] "The mean of Income is:  165790"
```

  - Sample Median or the "Middle" of the data – useful when data has extreme values
    - The Median is the *middle ranked* data (or 50th percentile – will see shortly)

      In **R**, *median*(income) gives the sample median of income

      With a calculator, we need to rank the data and get the 50th percentile.

```
> # Median of Income
> print(paste("The median of Income is: ", round(median(income), 2)))
[1] "The median of Income is:  142000"
```

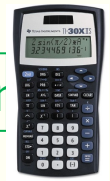
  - Sample Mode(s) or the most frequent value(s) in the data
    - The mode (after "fashion") is the *most frequent data point*

      There is **no function** to compute mode in **R**. Instead, we can use a frequency table, using the *table*() function.

      Mode is most often used with categorical data. For numerical data, often every value is a mode.

```
> # Frequency table to obtain mode
> table(income)
income
 38900  66000  82000 111000 112000 172000 182000 218000 242000 434000
     1      1      1      1      1      1      1      1      1      1
>
```

Practice with calculator



Measures of Central Tendency – Usefulness (**DescriptiveStat.R**)

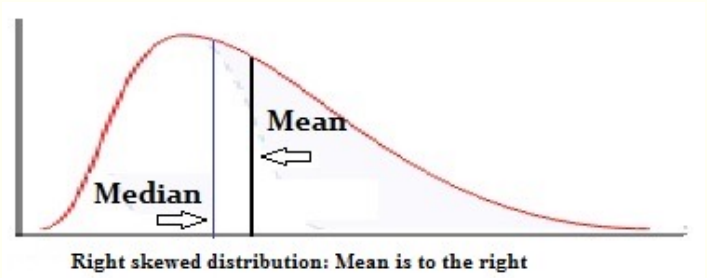
- In ***Descriptive Statistics*** central tendency measures are often used as a *representative single value* for the data belonging to the sample.
  - For income, the mean of 165790 is "representative" of income in the data set. It behaves like the moment of inertia in Physics.

```
> # Mean of Income for Males
> print(paste("The mean of Male Income is: ", round(mean(income[gender=="M"]), 2)))
[1] "The mean of Male Income is:  158316.67"
>
> # Mean of Income for Feales
> print(paste("The mean of Female Income is: ", round(mean(income[gender=="F"]), 2)))
[1] "The mean of Female Income is:  177000"
```

- They are also used to *compare variables or groups* in a sample.
- In addition, a comparison of mean vs median gives an indication of skewness of data in the sample – whether the data is:
  - Symmetric: mean = median
  - If a distribution is Left-skewed (longer-tail to the left) then generally (not always) mean < median (mean is to the "left")
  - If a distribution is Right-skewed (longer-tail to the right) then generally (not always) mean > median (mean is to the "right")
  - In the example, mean of income is greater than median so income so the data may be right-skewed

```
> # Mean of Income
> print(paste("The mean of Income is: ", round(mean(income), 2)))
[1] "The mean of Income is:  165790"
```

```
> # Median of Income
> print(paste("The median of Income is: ", round(median(income), 2)))
[1] "The median of Income is:  142000"
```



Right skewed distribution: Mean is to the right

Practice with calculator

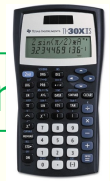
Measures of Dispersion (DescriptiveStat.R)

- Measures of the spread (or dispersion) of the data include:
  - (Sample) Variance (the square of the standard deviation)
  - (Sample) Standard Deviation
- Both these measure the spread or deviation of the data from the mean (center) of the data.
- To calculate sample standard deviation:
  - Subtract each observation value from the overall mean. This is called the *deviation* of the observation
  - Square all the deviations and add them up. This is the *sum of squared deviations*
  - Calculate Variance as (sum of squared deviations / (**n – 1**)), where n is the number of observations
  - Calculate standard deviation as the *square root of variance.*
- Standard Deviation is preferred over variance because standard deviation *has the same unit as the original observations* (e.g., standard deviation is in dollars, if the original observations are in dollars, but the unit of variance is dollars^2).
- Sometimes Range = [maximum – minimum] is used as a measure of dispersion.

Calculation	R Function
Sample Mean $\bar{\bar{X}} = \frac{1}{n}\sum_{i=1}^{n} x_i$	mean()
Sample Variance $s^2 = \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \bar{X})^2$	var()
Sample standard deviation $s = \sqrt{s^2}$	sd()

```
> deviation <- vector("numeric", 10)
> sq_deviation <- vector("numeric", 10)
> sum_sq_deviation = 0
> # Measures of dispersion
> for (i in 1:10) {
+   deviation[i] <- income[i] - mean(income)
+   sq_deviation[i] <- deviation[i]^2
+   sum_sq_deviation <- sq_deviation[i] + sum_sq_deviation
+ }
>
> print(paste("The sum of squared deviations from mean is ", sum_sq_deviation))
[1] "The sum of squared deviations from mean is  119746969000"
> print(paste("The sample variance is ", sum_sq_deviation/9))
[1] "The sample variance is  13305218777.7778"
> print(paste("The sample variance using var() function is ", var(income)))
[1] "The sample variance using var() function is  13305218777.7778"
> print(paste("The sample standard deviation is ", sqrt(sum_sq_deviation/9)))
[1] "The sample standard deviation is  115348.249998766"
> print(paste("The sample standard deviation using sd() is ", sd(income)))
[1] "The sample standard deviation using sd() is  115348.249998766"
```

Practice with calculator



Usefulness of Variance/Standard Deviation (**DescriptiveStat.R**)

- The larger the variance/standard deviation the greater the spread of data. It may be considered a measure of consistency of the data.
  - Consider two basketball players A & B. A shoots an average of 22 points per game with the standard deviation of 6 points. B shoots an average of 30 points with a standard deviation of 12 points. Clearly A is more consistent in scoring game to game than the latter, who is a higher scorer on average.
- Chebychev's theorem that we saw earlier also works for sample data. The theorem implies that less than $1/c^2$ percent of the data is found beyond c standard deviations from the mean (at least $1 - 1/c^2$)* 100 % of the data is found within c standard deviations from the mean.
  - Income data sorted:

```
> print(income[order(income)])
 [1]  38900  66000  82000 111000 112000 172000 182000 218000 242000 434000
```

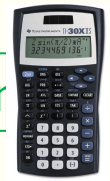
  - Mean ± 2 standard deviations = 177000 ± 115348.25 = [61,651.75. 292,348.25] should contain at least 7.5 observations; It contains 9 observations.

Descriptive Statistics - Measures of Location – Percentiles (Quantiles) (**DescriptiveStat.R**)

- Measures of location such as percentiles (quantiles) tell us what percent of values in the sample lie below the specified location.
- The **kth percentile** of a variable is the observation value that cuts off the first *k percent* of the data values, when it is sorted in ascending order.
- The kth percentile *may or may not* be part of the data.
  - To see this, let us first order (sort) income, and then compute the corresponding (approx.) percentile.
  - If we take 38900 (minimum) there are no observations below it, so it is the 0th percentile. If we take 82000, there are two observations(out of 10) below it so it is the 20th percentile. We notice that the 25th percentile (for example) is not part of the data.

```
> # print sorted income to see the quantiles
> print("Sorted Income")
[1] "Sorted Income"
> print(income[order(income)])
 [1]  38900  66000  82000 111000 112000 172000 182000 218000 242000 434000
```

Practice with calculator



Book Formula for finding the k^{th} Percentile

- Approach in your book:
  - Order the data from smallest to largest.
  - Calculate $i = (n+1)*k/100$ *where*
    - k = the k^{th} percentile. It may or may not be part of the data.
    - i = the index (ranking or position of a data value)
    - n = the total number of data
  - If i is an integer, then the k^{th} percentile is the data value in the i^{th} position in the ordered set of data.
  - If i is not an integer, then round i up and round i down to the nearest integers. Average the two data values in these two positions in the ordered data set.

```
> print(income[order(income)])
 [1]  38900  66000  82000 111000 112000 172000 182000 218000 242000 434000
```

    - Example: **25th percentile (1st quartile)** k = 25, n = 10 so i = 11*25/100 = 2.75. Rounding up, i = 3 and rounding down i = 2. So, we average 66000 and 82000 to get **74000** as the 25th percentile or 1st quartile.
    - 50th percentile or median: i = 11*50/100 = 5.5; So we average the 5th and 6th ranked observations 112000 and 172000 to get **142000 as the median**.

Practice with calculator

Percentiles in R (**DescriptiveStat.R**)

- There are many algorithms (with slightly different answers) to find percentiles– The R language has 9 different types you can choose from.
- The closest algorithm in **R** to the book's approach (it is not the same) is Type = 2.

```
> print(quantile(income, probs=seq(0, 1, 0.05), type = 2))
    0%     5%    10%    15%    20%    25%    30%    35%    40%    45%    50%    55%    60%    65%    70%
 38900  38900  52450  66000  74000  82000 111000 111000 111500 112000 142000 172000 182000 182000 218000
   75%    80%    85%    90%    95%   100%
218000 230000 242000 338000 434000 434000
```

- If asked to calculate by hand, use book's approach; In **R** use Type 2.

Usefulness of Percentiles – IQR and Outliers (**DescriptiveStat.R**)

- Percentiles can also be used to identify unusual observations (potential outliers)
- The **interquartile range (IQR)** is a number that indicates the spread of the middle half or the middle 50% of the data. It is the difference between the third quartile (*Q*3) and the first quartile (*Q*1). i.e., ***IQR*** = *Q*3 – *Q*1;
- The *IQR* can help to determine potential **outliers. A value is suspected to be a potential outlier if it is less than (1.5)(*IQR*) below the first quartile or more than (1.5)(*IQR*) above the third quartile.**
- Potential outliers always require further ir

- In **R**, *IQR() gives the Inter-Quartile Range.*

```
> #
> income_iqr <- IQR(income, type = 2)
> print(income_iqr)
[1] 136000
> p25 <- quantile(income, probs = 0.25, na.rm=FALSE, names = TRUE, type=2)
> p75 <- quantile(income, probs = 0.75, na.rm=FALSE, names = TRUE, type=2)
> print(paste(min(income)," p25 ",p25, median(income), " p75 ",p75, " ", max(income)))
[1] "38900  p25  82000 142000  p75  218000   434000"
> llimit <- p25 - 1.5*income_iqr
> ulimit <- p75 + 1.5*income_iqr
> print(paste("Lower limit ", llimit, " Upper limit ", ulimit))
[1] "Lower limit  -122000  Upper limit  422000"
```

From the data we see that Jake Ritter's Income of 434,000 is an *outlier*

ID	Name	Age	Gender	Education	Credit Score	Income	Net Worth	Sales
001	Adams, John	36	M	HS	350	38,900	65,924	1,535
002	Ramesh, Jyoti	23	F	Bachelors	600	172,000	178,154	2,196
003	Mendez, Nick	67	M	Bachelors	700	218,000	265,209	1,287
004	Mendez, Joan	38	F	PhD	550	182,000	85,277	2,143
005	Ritter, Jake	24	M	Masters	625	434,000	193,760	707
006	Rao, Eric	61	M	PhD	770	82,000	314,953	2,170
007	Blake, Ann	26	F	HS	490	112,000	192,946	1,229
008	Bishop, Marge	44	F	Masters	540	242,000	339,705	520
009	Ahmed, Mo	31	M	Masters	680	111,000	185,767	2,326
010	Shultz, Dante	44	M	Bachelors	280	66,000	97,778	588

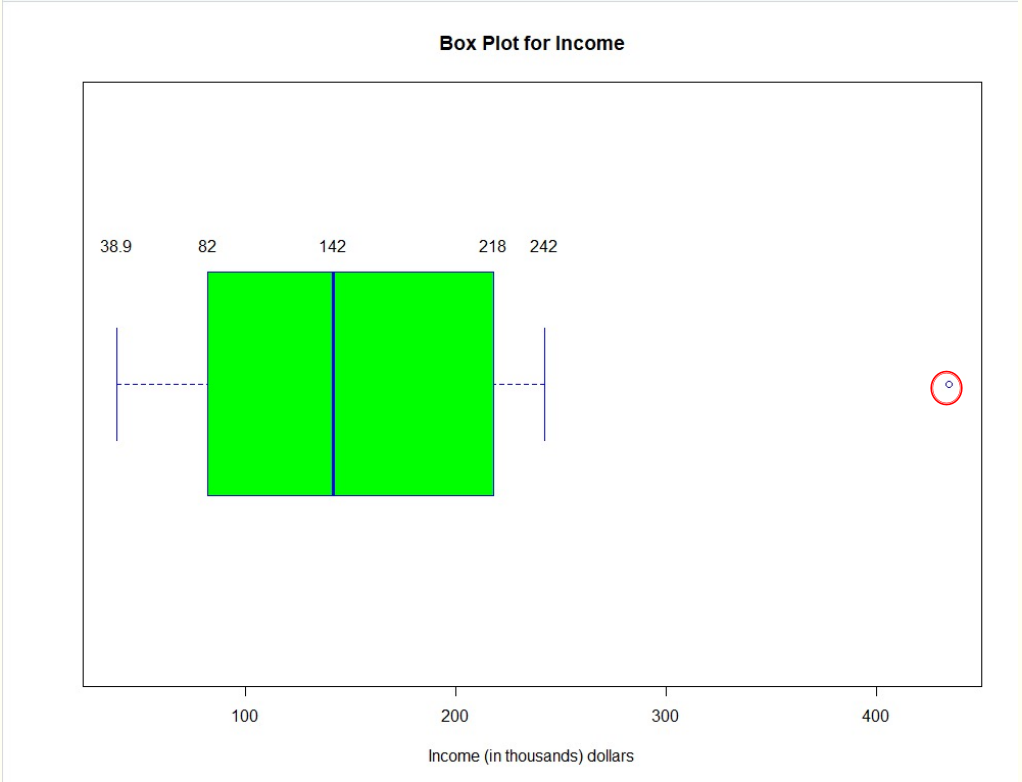
Box Plots and Percentiles (**DescriptiveStat.R**)

- A graphical view of percentiles, IQR and outliers can be obtained using Box Plots.
- We can obtain the statistics used in the BoxPlot by using *boxplot.stats(income)$stats*.

```
> print(boxplot.stats(income)$stats)
[1]  38900  82000 142000 218000 242000
```

- The calculation of the two whiskers in **R** is complicated. The two whiskers (38900 and 242000) are the lowest datum still within 1.5 IQR of the lower quartile, and the highest datum still within 1.5 IQR of the upper quartile (often called the Tukey boxplot)
- Outliers are identified if they lie outside the whiskers. Here, 434000 is an outlier.
- **Note**: I have done income/1000 in the box plot for ease of visualization.

```
#
boxplot(income/1000, main="Box Plot for Income",
        xlab="Income (in thousands) dollars",
        border="blue",
        col="green",
        horizontal = TRUE)
text(x=boxplot.stats(income/1000)$stats, labels = boxplot.stats(income/1000)$stats, y = 1.25)
#
```



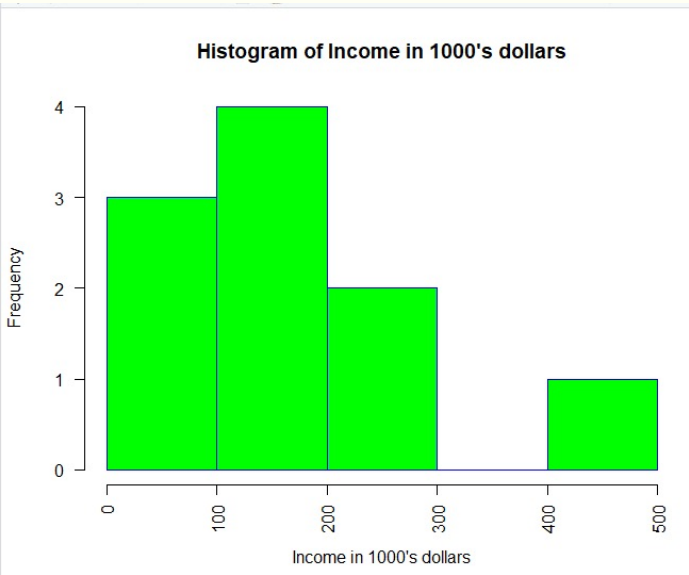

```
[1] "Lower limit  -122000  Upper limit  422000"
> print(income[order(income)])
 [1]  38900  66000  82000 111000 112000 172000 182000 218000 242000 434000
```

Shape of Data Distribution – Graphical (**DescriptiveStat.R**)

- **Histograms**: Histograms show the frequency of data points within user-defined "bins" (for ratio or *numerical* data). They can show asymmetry in the data (skewness) as well as whether it is peaked or flat (kurtosis).

```
hist(income/1000, main="Histogram of Income in 1000's dollars",
     xlab="Income in 1000's dollars",
     border="blue",
     col="green",
     xlim=c(0,500),
     las=2,
     breaks=seq(0,500,100))
```

- In **R**, you can use the default bins, or you can define it yourself.
  - For the Income variable, we have only 10 data points. Further, we will divide income by 1000 to keep the x-axis manageable.
  - We will define the breaks (that define the bins) so that each bin is wide enough to have multiple records where possible.
  - *breaks=seq(0,500,100))* will create bins (0 to 100), (100 to 200),…, (400 to 500)
  - *las=2* determines the orientation of x-axis labels (try = 1)
- The histogram shows what appears to be an outlier, causing the data to appear right-skewed.

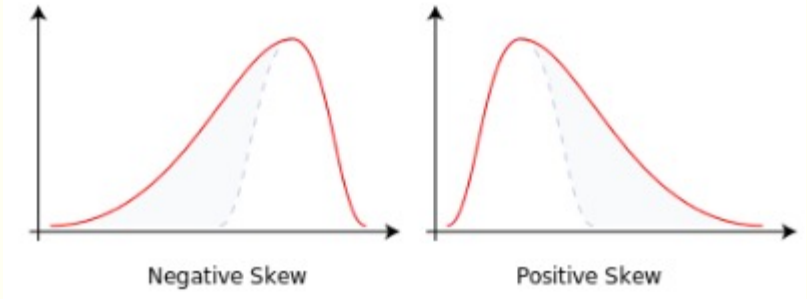


Shape of Data Distribution – Numerical descriptions

- As seen earlier we can describe the shape of a distribution using:
  - Symmetric: mean = median
  - Left-skewed – longer tail to the left
  - Right-skewed – longer tail to the right

- The bell-shaped symmetric normal distribution is often the standard by which the shape (lack of symmetry) of sample data is measured.
  - Skewness is a measure asymmetry of the data. The skewness of a normal distribution is 0. If the skewness measure of a distribution is < 0, has a negative skew and > 0 it has a positive skew.

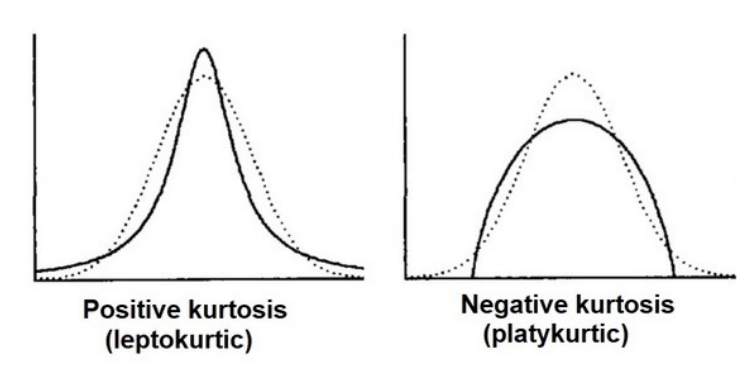
- Consider the two distributions in the figure just below. :
  - *negative skew*: The left tail is longer; the mass of the distribution is concentrated on the right of the figure. The distribution is said to be *left-skewed*, *left-tailed*, or *skewed to the left*, despite the fact that the curve itself appears to be skewed or leaning to the right; *left* instead refers to the left tail being drawn out and, often, the mean being skewed to the left of a typical center of the data. A left-skewed distribution usually appears as a *right-leaning* curve.[1]
  - *positive skew*: The right tail is longer; the mass of the distribution is concentrated on the left of the figure. The distribution is said to be *right-skewed*, *right-tailed*, or *skewed to the right*, *despite* the fact that the curve itself appears to be skewed or leaning to the left; *right* instead refers to the right tail being drawn out and, often, the mean being skewed to the right of a typical center of the data. A right-skewed distribution usually appears as a *left-leaning* curve.[1]



Source: Wikipedia

Shape of Data Distribution – Numerical descriptions

- Another numerical measure of the shape of a distribution is (Pearson's) *Kurtosis*
- Kurtosis is often referred to (somewhat incorrectly) as the "peakedness" of a distribution. It is better understood as a measure of the "lightness" or "heaviness" of the tails. That is, how much of the data is in the tails.
- The normal distribution has a kurtosis of 3, and other distributions are described in terms of kurtosis, relative to the kurtosis of the normal distribution (i.e., 3). Thus, excess kurtosis = kurtosis – 3.
- The normal distribution is *mesokurtic* with an excess kurtosis of 0
- A distribution with positive excess kurtosis is called *leptokurtic*, or leptokurtotic. "Lepto-" means "slender". In terms of shape, a leptokurtic distribution has fatter tails (more infrequent extreme values).
- A distribution with negative excess kurtosis is called platykurtic, or platykurtotic. "Platy-" means "broad".[11] In terms of shape, a platykurtic distribution has thinner tails (fewer infrequent extreme values).



Source: Wikipedia

Skewness and Kurtosis in R

- There are no built-in functions for skewness and kurtosis in R
- However, here are many external installable libraries that can do it.
- Here is an example that uses the “moments” library

```
> install.packages("moments")
Installing package into 'C:/Users/sarathy/Documents/R/win-library/3.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/moments_0.14.zip'
Content type 'application/zip' length 40819 bytes (39 KB)
downloaded 39 KB

package 'moments' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\sarathy\AppData\Local\Temp\RtmpGs2d8Q\downloaded_packages
> library("moments")
Warning message:
package 'moments' was built under R version 3.4.4
> skewness(income)
[1] 1.21632
> kurtosis(income)
[1] 3.934066
```

- The skewness measure tells us that Income is right-skewed, and the kurtosis measure tells us that the distribution has fatter/heavier tails (more frequent extreme values than the normal distribution)

DESCRIPTIVE STATISTICS

Grouped Data

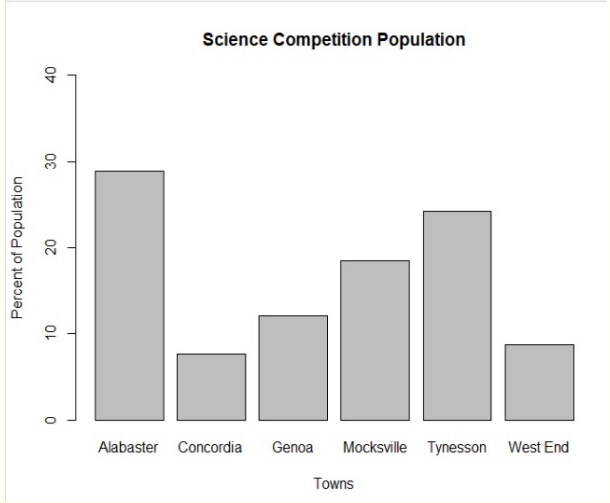
Grouped Data

- Sometimes the sample data is either collected or presented as count data.
- The groups can be nominal or ordinal.
- The data is often presented as a frequency (count) table.
- It is also possible that what appears to be count data can actually be analyzed as numerical data…Analyzing data as numerical data is always preferable to analyzing as count data.
- The formulas for calculating the sample statistics will differ from those for numerical data.
- The typical graphical representations of grouped data are **Pie Charts**, **Bar Charts** and **Histograms**.

Descriptive Statistics – Nominal Group Variables (GroupData.R)

- If the sample data available is Nominal data, i.e., we have counts of the each category, the descriptive statistics that can be provided is very limited, often restricted to visual displays of the counts in each group.
- High School is a **Nominal Group Variable**
- Typical visual descriptive statistics include tables, bar graphs and pie charts. We can deduce basic information such as which group has the greatest counts, which has the least, etc.

```
> science_pct <- c(28.9, 7.6, 12.1, 18.5, 24.2, 8.7)
> barplot(science_pct, main="Science Competition Population", horiz=FALSE, xlab ="Towns", ylab="Percent of Population",
ylim=c(0, 40), names.arg=c("Alabaster", "Concordia", "Genoa", "Mocksville", "Tynesson", "West End"))
```

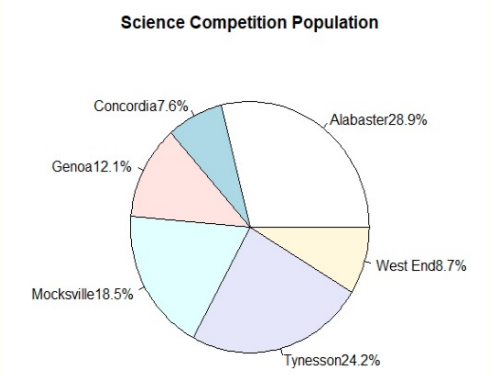


10. David County has six high schools. Each school sent students to participate in a county-wide science competition. **Table 2.41** shows the percentage breakdown of competitors from each school, and the percentage of the entire stude population of the county that goes to each school. Construct a bar graph that shows the population percentage of competit from each school.

High School	Science competition population	Overall student population
Alabaster	28.9%	8.6%
Concordia	7.6%	23.2%
Genoa	12.1%	15.0%
Mocksville	18.5%	14.3%
Tynneson	24.2%	10.1%
West End	8.7%	28.8%

Table 2.41

```
> science_pct <- c(28.9, 7.6, 12.1, 18.5, 24.2, 8.7)
> towns <- c("Alabaster", "Concordia", "Genoa", "Mocksville", "Tynesson", "West End")
> labeled_towns <- paste(towns, science_pct,"%",sep="")
> pie(science_pct, labels = labeled_towns, main="Science Competition Population")
```



Example: Book Page 129

Practice with calculator

Descriptive Statistics – **Ordinal Group** Variables (**GroupData.R**)

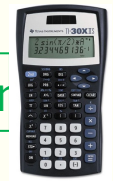
- Example: **Book Problem 2.108; Page 151**
  - Santa Clara County, CA, has approximately 27,873 Japanese-Americans. Their ages are as shown in Table.
  - Here Age can be treated as an **Ordinal Group Variable**. There is an order to the age groups from low to high.
  - **When we have ordinal categories, we can get additional information in the form of relative frequencies**.

- The relative frequency can be interpreted similar to probability
  - For example, the relative frequency for 18-24 (8.0%) is interpreted as the probability of Japanese Americans in Santa Clara who are "24 and under". Thus, it behaves like a discrete probability distribution.
  - Similarly, the cumulative relative frequency can be interpreted as a cumulative probability. That is, the "probability" of the event "Japanese-American in Santa Clara age 24 or under" is 0.269. We can get rough percentiles as well. For example, age 24 is roughly the 27th percentile and age group 45 – 54 is roughly the 78th percentile.

- Graphically, we use bar charts (or bar plots). The bar plot of frequencies is similar to an nominal variable. The bar chart of *relative frequencies provides additional information.*

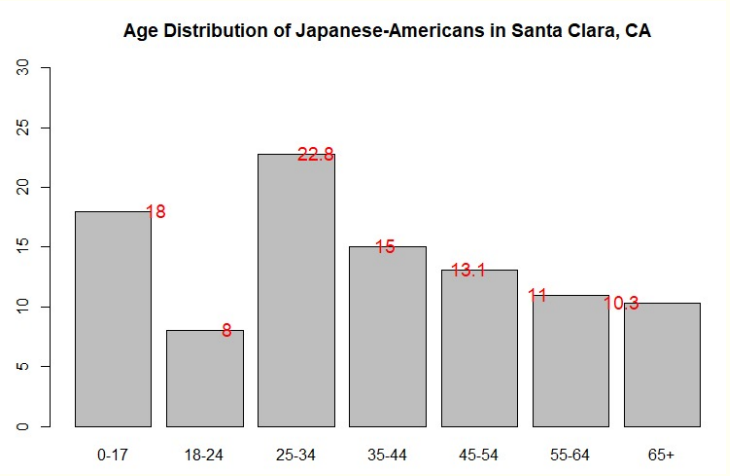
Age	Group Percent of Community (Relative Frequency)	Relative Group Percent (Cumulative Relative Frequency)
0–17	18.9	18.9
18–24	8.0	26.9
25–34	22.8	49.7
35–44	15.0	64.7
45–54	13.1	77.8
55–64	11.9	89.7
65+	10.3	100

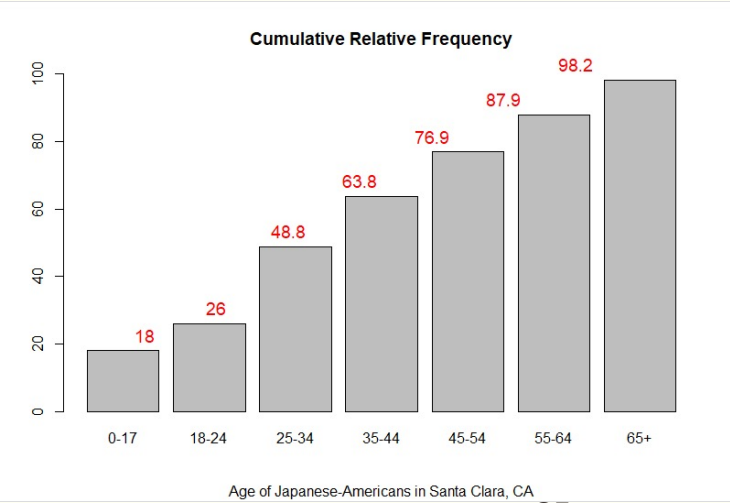
Practice with calculator



Descriptive Statistics – **Ordinal Group** Variables (**GroupData.R**)

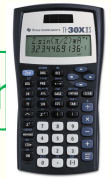
```
18 # Ordinal Group Example
19 # Book Page 151 - Problem 2.108
20 # Bar Plots
21 #
22 age_group <-c("0-17", "18-24", "25-34", "35-44", "45-54", "55-64", "65+")
23 rel_freq <- c(18.0, 8.0, 22.8, 15.0, 13.1, 11.0, 10.3)
24 cum_rel_freq <- cumsum(rel_freq)
25 barplot(rel_freq,
26         main = "Age Distribution of Japanese-Americans in Santa Clara, CA",
27         horiz = FALSE,
28         ylim =c(0, 30),
29         names.arg=age_group)
30 text(rel_freq, labels =rel_freq, cex=1.2, pos=4, col="red")
31 barplot(cum_rel_freq,
32         main = "Cumulative Relative Frequency",
33         sub = "Age of Japanese-Americans in Santa Clara, CA",
34         ylim=c(0, 100),
35         horiz = FALSE,
36         names.arg=age_group)
37 text(cum_rel_freq, labels =cum_rel_freq, cex=1.2, pos=3, col="red")
```





Descriptive Statistics – Numerical *Range* Group Variables (GroupData.R)

Practice with calculator



- When it is possible to treat the group data as a range (or interval) of *numbers*, we can calculate additional descriptive statistics from the frequency table.
- The mid-points of each range is taken as a representative value for that range
- **Example: Book Example 2.30, Page 107**
  - A frequency (number of students) table displaying professor Blount's last statistic test is shown. We have also calculated the relative frequency.
  - If we treat the midpoint as a discrete random variable $X=x_i$, and the *frequency as* f_i *(the relative frequency can be considered as probability* p_i), we can calculate the expected value as for any discrete distribution.
  - *Expected value (sample mean* $\bar{X}$*)* = 77.69
    - **Note** that this is an approximate mean of the Grade because we are using mid-points of grade intervals as representative of Grade.
  - Sample standard deviation (s) = 10.53

	x_i	(Frequency f_i)	p_i
Grade Interval	**Midpoint**	**Number of Students**	**Relative Frequency**
50–56.5	53.25	1	0.0526
56.5–62.5	59.5	0	0
62.5–68.5	65.5	4	0.2105
68.5–74.5	71.5	4	0.2105
74.5–80.5	77.5	2	0.1053
80.5–86.5	83.5	3	0.1579
86.5–92.5	89.5	4	0.2105
92.5–98.5	95.5	1	0.0526

- n is the number of categories/intervals in

Calculation
Sample Mean $\bar{\bar{X}} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$
Sample Variance $s^2 = \frac{1}{(\sum_{i=1}^{n} f_i - 1)} \sum_{i=1}^{n} f_i (x_i - \bar{X})^2$
Sample standard deviation $s = \sqrt{s^2}$

Note: Book approach is not used

Descriptive Statistics – Numerical *Range* Group Variables (**GroupData.R**)

Practice with calculator

```
# Numerical (range) Group variables
# Book Example 2.30, Page 107
# The mid-point of the numerical range is used as a number value for the variable
#
grade_interval <- c("50-56.5", "56.5-62.5", "62.5-68.5","68.5-74.5","74.5-80.5","80.5-86.5","86.5-92.5","92.5-98.5")
xi <- c(53.25, 59.5, 69.5, 71.5, 77.5, 83.5, 89.5, 95.5)
fi <- c(1,0,4,4,2,3,4,1)
df1 <- data.frame(grade_interval, xi, fi)
df1$pi <- df1$fi/sum(df1$fi)
df1$xifi <- df1$xi*df1$fi
expected_val <- sum(df1$xifi)/sum(df1$fi)
df1$deviation <- df1$xi - expected_val
df1$sq_dev = df1$deviation*df1$deviation
df1
varian <- sum(df1$fi*df1$sq_dev)/(sum(df1$fi)-1)
std_dev <- sqrt(varian)
print(expected_val)
print(std_dev)
```

```
  grade_interval    xi fi         pi   xifi   deviation       sq_dev
1       50-56.5 53.25  1 0.05263158  53.25 -24.4473684 597.67382271
2     56.5-62.5 59.50  0 0.00000000   0.00 -18.1973684 331.14421745
3     62.5-68.5 69.50  4 0.21052632 278.00  -8.1973684  67.19684903
4     68.5-74.5 71.50  4 0.21052632 286.00  -6.1973684  38.40737535
5     74.5-80.5 77.50  2 0.10526316 155.00  -0.1973684   0.03895429
6     80.5-86.5 83.50  3 0.15789474 250.50   5.8026316  33.67053324
7     86.5-92.5 89.50  4 0.21052632 358.00  11.8026316 139.30211219
8     92.5-98.5 95.50  1 0.05263158  95.50  17.8026316 316.93369114
> varian <- sum(df1$fi*df1$sq_dev)/(sum(df1$fi)-1)
> std_dev <- sqrt(varian)
> print(expected_val)
[1] 77.69737
> print(std_dev)
[1] 10.52859
```

Calculation

Sample Mean $\bar{\bar{X}} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$

Sample Variance $s^2 = \frac{1}{(\sum_{i=1}^{n} f_i - 1)} \sum_{i=1}^{n} f_i (x_i - \bar{X})^2$

Sample standard deviation $s = \sqrt{s^2}$

Note: Book approach is not used

Descriptive Statistics – Numerical Group Variables (GroupData.R)

Practice with calculator

- Sometimes, the group variable is actually a numerical variable whose counts are reported as a table. i.e., the numerical value repeats itself.
- **Example – Book Problem 2.15 – Page 91**
  - Clearly, we can analyze this as numerical data rather than group data.
  - We will do it both ways and compare results.
- **Method 1: (Analyzing as raw numeric data)**

Calculation	R Function
Sample Mean $\bar{\bar{X}} = \frac{1}{n}\sum_{i=1}^{n} x_i$	mean()
Sample Variance $s^2 = \frac{1}{(n-1)}\sum_{i=1}^{n}(x_i - \bar{X})^2$	var()
Sample standard deviation $s = \sqrt{s^2}$	sd()

Example 2.15

Fifty statistics students were asked how much sleep they get per school night (rounded to the nearest hour). The results were: x_i f_i

AMOUNT OF SLEEP PER SCHOOL NIGHT (HOURS)	FREQUENCY	RELATIVE FREQUENCY	CUMULATIVE RELATIVE FREQUENCY
4	2	0.04	0.04
5	5	0.10	0.14
6	7	0.14	0.28
7	12	0.24	0.52
8	14	0.28	0.80
9	7	0.14	0.94
10	3	0.06	1.00

Table 2.22

```
hours_sleep <- c(4,4,5,5,5,5,5,6,6,6,6,6,6,6,7,7,7,7,7,7,7,7,7,7,7,
                 8,8,8,8,8,8,8,8,8,8,8,8,8,8,9,9,9,9,9,9,9,10,10,10)
```

```
> #From raw data
> #
> print(paste("The expected value (sample mean) is ",round(mean(hours_sleep),4),sep=""))
[1] "The expected value (sample mean) is 7.28"
> print(paste("The sample standard deviation is ",round(sd(hours_sleep),4),sep=""))
[1] "The sample standard deviation is 1.4988"
```

Practice with calculator

(GroupData.R)

Method 2: (Analyzing the table data)

We can use the frequency table to calculate the sample mean and sample standard deviation.

n is the number of categories in group

x_i is the value of data point and f_i its frequency

```
# Load the data into a vector using the c() function
hours_sleep <- c(4,4,5,5,5,5,5,6,6,6,6,6,6,6,7,7,7,7,7,7,7,7,7,7,7,7,
                 8,8,8,8,8,8,8,8,8,8,8,8,8,8,9,9,9,9,9,9,9,10,10,10)
# Create a frequency table of the data using the table() function
#
tbl_stu_freq <- table(hours_sleep)
# Put the table in a data Frame
df_students <- data.frame(tbl_stu_freq)
df_students
#
# When we load table tbl_stu_freq into the data frame df_students
# the first column containing number of students in the data frame
# becomes a class variable (categorical)
# We want to convert them into numbers
# We do that using as.numeric(as.character(df_students$hours_sleep))
# num_hours is now a numeric vector representing hours_sleep
#We simply add num_hours as a new column in the data frame df_students
#
df_students$num_hours <- as.numeric(as.character(df_students$hours_sleep))
#
# Add cum_freq and cum_rel_freq as new columns in the data frame df_students
df_students$rel_freq <- df_students$Freq/sum(df_students$Freq)
df_students$cum_rel_freq <- cumsum(df_students$rel_freq)
df_students
```

Example 2.15

Fifty statistics students were asked how much sleep they get per school night (rounded to the nearest hour). The results were:

AMOUNT OF SLEEP PER SCHOOL NIGHT (HOURS)	FREQUENCY	RELATIVE FREQUENCY	CUMULATIVE RELATIVE FREQUENCY
4	2	0.04	0.04
5	5	0.10	0.14
6	7	0.14	0.28
7	12	0.24	0.52
8	14	0.28	0.80
9	7	0.14	0.94
10	3	0.06	1.00

Table 2.22

	hours_sleep	Freq	num_hours	rel_freq	cum_rel_freq
1	4	2	4	0.04	0.04
2	5	5	5	0.10	0.14
3	6	7	6	0.14	0.28
4	7	12	7	0.24	0.52
5	8	14	8	0.28	0.80
6	9	7	9	0.14	0.94
7	10	3	10	0.06	1.00

Practice with calculator

(GroupData.R)

```
#
#Multiply num_hours * Freq to get xifi and add as a column to df_students
#
df_students$xifi <- df_students$num_hours * df_students$Freq
df_students
#
# We now calculate the expected value (sample mean) from the frequency table
#
expected_val = sum(df_students$xifi)/sum(df_students$Freq)
print(paste("The expected value of student sleep hours (sample mean) is ",round(expected_
#
# We calculate the deviation of num_students from the mean as well as its square
#
df_students$mean_dev <- df_students$num_hours - expected_val
df_students$mean_dev_sq <- df_students$mean_dev * df_students$mean_dev
df_students
#
varian <- sum(df_students$Freq*df_students$mean_dev_sq)/(sum(df_students$Freq) - 1)
stand_dev <- sqrt(varian)
print(paste("The sample standard deviation is ",round(stand_dev,4),sep=""))
#
#From raw data
#
print(paste("The expected value (sample mean) is ",round(mean(hours_sleep),4),sep=""))
print(paste("The sample standard deviation is ",round(sd(hours_sleep),4),sep=""))
print(paste("The sample median is ",round(median(hours_sleep),4),sep=""))
```

```
> df_students
  hours_sleep Freq num_hours rel_freq cum_rel_freq xifi
1           4    2         4     0.04          0.04    8
2           5    5         5     0.10          0.14   25
3           6    7         6     0.14          0.28   42
4           7   12         7     0.24          0.52   84
5           8   14         8     0.28          0.80  112
6           9    7         9     0.14          0.94   63
7          10    3        10     0.06          1.00   30
```

```
  hours_sleep Freq num_hours rel_freq cum_rel_freq xifi mean_dev mean_dev_sq
1           4    2         4     0.04          0.04    8    -3.28     10.7584
2           5    5         5     0.10          0.14   25    -2.28      5.1984
3           6    7         6     0.14          0.28   42    -1.28      1.6384
4           7   12         7     0.24          0.52   84    -0.28      0.0784
5           8   14         8     0.28          0.80  112     0.72      0.5184
6           9    7         9     0.14          0.94   63     1.72      2.9584
7          10    3        10     0.06          1.00   30     2.72      7.3984
```

```
[1] "The expected value of student sleep hours (sample mean) is 7.28"
#
```

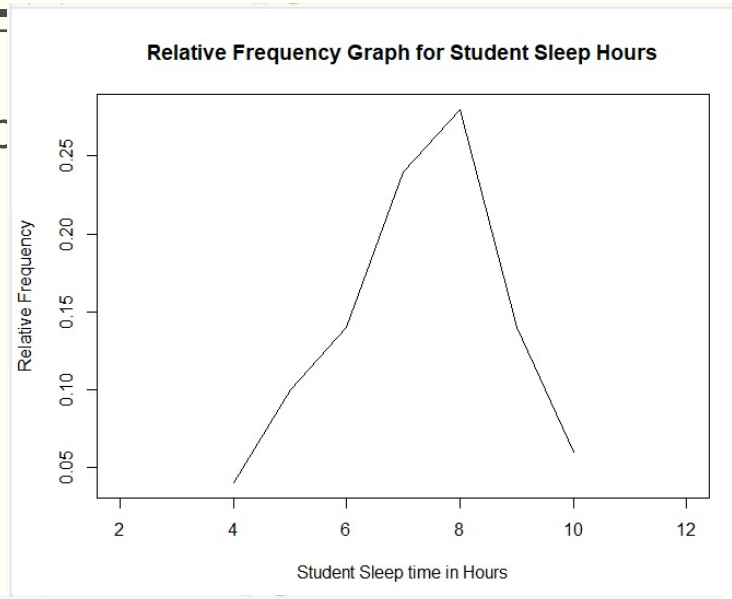
```
[1] "The sample standard deviation is 1.4988"
```

Calculation
Sample Mean $\bar{\bar{X}} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$
Sample Variance $s^2 = \frac{1}{(\sum_{i=1}^{n} f_i - 1)} \sum_{i=1}^{n} f_i (x_i - \bar{X})^2$
Sample standard deviation $s = \sqrt{s^2}$

Plotting the Frequency and Cumulative Frequency (**GroupData.R**)

- We can also plot frequency and relative frequency line grap

```
#
plot(df_students$num_hours, df_students$rel_freq,
     main="Relative Frequency Graph for Student Sleep Hours",
     xlab="Student Sleep time in Hours",
     xlim=c(2,12),
     ylab = "Relative Frequency",
     type ="l")
plot(df_students$num_hours, df_students$cum_rel_freq,
     main="Cumulative Relative Frequency Graph for Student Sleep Hours",
     xlab="Student Sleep time in Hours",
     xlim=c(2,12),
     ylab = "Cumulative Relative Frequency",
     type ="l")
```

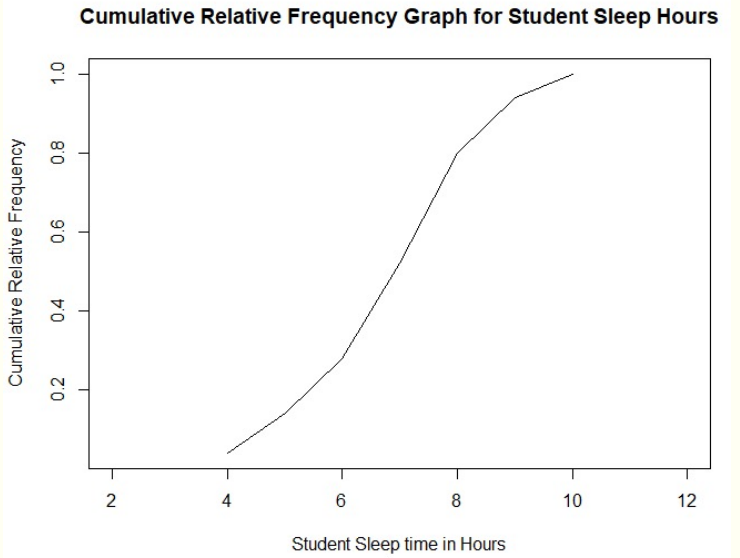


Example 2.15

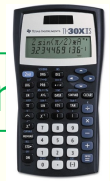
Fifty statistics students were asked how much sleep they get per school night (rounded to the nearest hour). The results were:

AMOUNT OF SLEEP PER SCHOOL NIGHT (HOURS)	FREQUENCY	RELATIVE FREQUENCY	CUMULATIVE RELATIVE FREQUENCY
4	2	0.04	0.04
5	5	0.10	0.14
6	7	0.14	0.28
7	12	0.24	0.52
8	14	0.28	0.80
9	7	0.14	0.94
10	3	0.06	1.00

Table 2.22



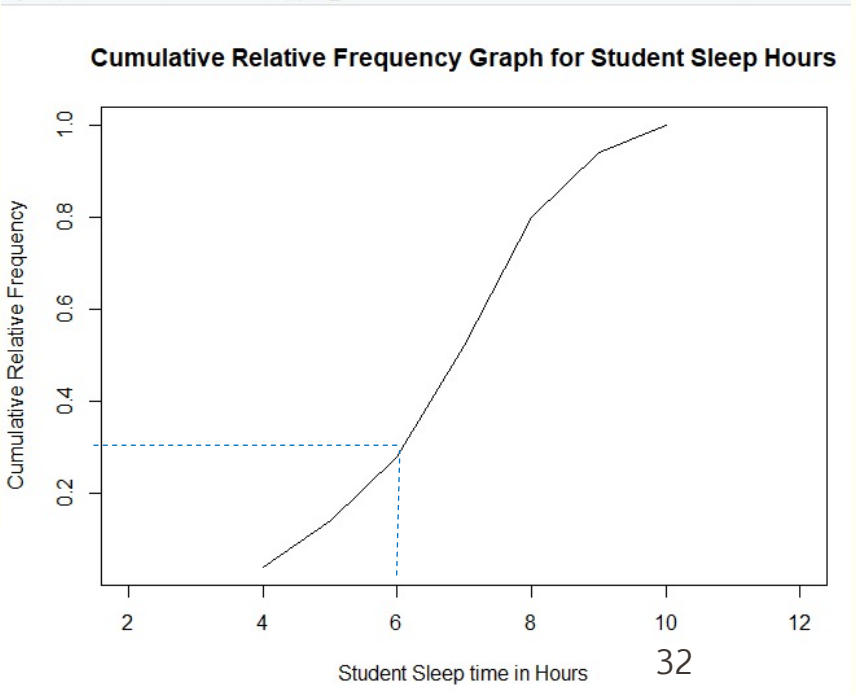
Practice with calculator



Using Frequency Tables and Plots (**GroupData.R**)

- We can use the cumulative frequency table (or the plot of cumulative frequency) to answer some of the questions.
- In this sample, what percent of students slept less than 6 hours?
  - Based on the cumulative frequency table, which behaves like a cdf, that corresponds to a relative cumulative frequency of 0.28.
  - This also corresponds to the 28th percentile for the data.
- What is the median of the data?
  - We notice that 7 hours of sleep has a relative frequency of 0.52. So we know the median is close to 7 hours.
- What is the 75th percentile for the data?
  - We notice that 8 hours corresponds to the 80th percentile, so it is close to 8.
- *The mean calculated earlier was 7.28 and the median is 7, so the data is somewhat right-skewed*

	stat_students	Freq	cum_freq	cum_rel_freq
4	4	2	2	0.04
5	5	5	7	0.14
6	6	7	14	0.28
7	7	12	26	0.52
8	8	14	40	0.80
9	9	7	47	0.94
10	10	3	50	1.00



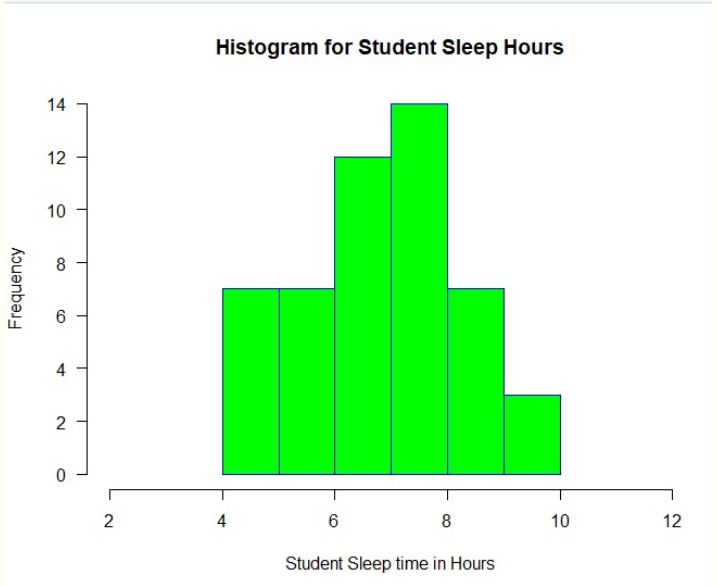

```
> # Obtain the quantiles and the IQR
> quantile(hours_sleep, probs = seq(0,1, 0.05), na.rm = FALSE, names = TRUE, type = 2)
  0%   5%  10%  15%  20%  25%  30%  35%  40%  45%  50%  55%  60%  65%  70%  75%  80%  85%  90%  95% 100%
 4.0  5.0  5.0  6.0  6.0  6.0  7.0  7.0  7.0  7.0  7.0  8.0  8.0  8.0  8.0  8.0  8.5  9.0  9.0 10.0 10.0
> IQR(hours_sleep, type = 2)
[1] 2
```

Histogram for Grouped Data (**GroupData.R**)

- **Histograms**: Histograms show the frequency of data points within existing classes of nominal data. They can show asymmetry in the data (skewness) as well as whether it is peaked or flat (kurtosis).

```
hist(hours_sleep, main="Histogram for Student Sleep Hours",
     xlab="Student Sleep time in Hours",
     border="blue",
     col="green",
     xlim=c(2,12),
     las=1,
     breaks=7)
```

```
> library("moments")
Warning message:
package 'moments' was built under R version 3.4.4
> skewness(hours_sleep)
[1] -0.2667215
> kurtosis(hours_sleep)
[1] 2.536407
```



The mean calculated earlier was 7.28 and the median is close to 7 (though not exactly known), so the data is somewhat right-skewed by this measure. But, skewness measure is negative indicating left-skewness.

The positive kurtosis value means that the distribution is platykurtic (fatter/heavier tails - more frequent extreme values than the normal distribution)

Summary on Descriptive Statistics

- Descriptive Statistics describe a sample through calculations of summary *statistics* such as mean, median, variance, standard deviation, percentiles, skewness, kurtosis etc.
- They provide information on the nature of the variables that represent the population.
- The sample can also be described graphically for visual ease
- Descriptive statistics are an attempt to understand the underlying population from which the sample is drawn. Assuming that an appropriate sampling technique (such as random sampling) is used, and assuming a reasonable sample size, the sample behaves as a representative of the population.
- For example:
  - If we see that the sample has a skewed distribution, it tells us that (potentially) the population distribution *may* also be skewed.
  - If the sample variance or standard deviation is large, the population *may* also have a large variance
- However, descriptive statistics are limited in their ability to confirm (or deny) specific population characteristics. They serve a purely *exploratory* function.
- To make specific inferences about population (parameters) we need the notion of the *sampling distribution* that is at the *core of inferential statistics*.

INFERENTIAL STATISTICS

See Book Chapter 7 – Additional Material included here

Population

- A population is a well-defined collection of *measurements* on "*subjects*" (people, things, processes, etc.) about which you need to *make a conclusion*.
  - The measurements are represented by a quantity X = Heights of Males in Oklahoma
- Generally the population of measurements is considered so large that it is not fully knowable. Further, we are usually only interested in some summary aspect of the population, such as the mean or standard deviation.
- In other words we want to estimate or conclude about some aspect of the population represented by the measurements.
- Often, the measurements represent a *probability distribution*. We learned from the previous lectures that probability distributions can be summarized by the pdf or pmf involving a few *parameters.*
- If we know the **parameters and the type of distribution**, we can describe the population (of measurements) better.
- So, in inferential statistics, we are really interested in *concluding about unknown population parameters* such as the *mean* or the *standard deviation* of the population.

Inferential Statistics

- Inferential Statistics is about *estimating parameters of population distributions*
- We have seen the following distributions:
  - Binomial with parameters (**n**, **p**)
  - Multinomial with parameters (**k**, **n**, $\mathbf{p_k}$)
  - Negative Binomial with parameters (**n**, **r**, **p**) – Special case is Geometric with r = 1
  - Poisson with parameter (λ)
  - Uniform with parameters (**a**, **b**)
  - Exponential with parameter (λ)
  - Normal with parameter (μ, σ)
- One of these (and of many others) could be the population distribution, depending on the context.
- Generally, the population is considered too large and/or too expensive to enumerate and therefore parameters of the population distribution are generally unknown.
- We are interested in obtaining "good" *estimates* of these parameters, by *taking a sample*.
- Thus, inferential statistics operates based on an underlying "population" model of the data. This may be different in data mining.

Samples and Inferential Statistics

- Estimation of (the usually unknown) population parameters (such as mean and standard deviation) is made by taking a *sample* from the population and computing statistics (also called *estimators*) that estimate the parameters from the sample.
- The most common sampling technique is *random sampling*, where every population observation has equal chance of being selected in the sample. Other types of sampling may also be appropriate depending on context.
- **Two important things to keep in mind:**

    1. *Any quantity that is computed from the sample* is a statistic

        In a sample of Oklahomans, the average (or the mean) height of the sample, the standard deviation, the largest height, the smallest height, the middle-ranked height (median), the most common height (mode), etc., are all statistics

    2. *Sample Statistics estimate Population Parameters*

- A statistic used to estimate a population parameter is also called an *estimator*. The value of the statistic is called the *estimate*.
- A single population parameter may be estimated using many different statistics or estimators.
    - Thus, we may choose to estimate the unknown (population) mean height of Oklahomans, using the sample median, instead of the sample mean

Sample Statistics to Estimate Population Parameters

- Thus far, we have seen that we can calculate many statistics from a single sample
  - Statistics for center of sample – Mean, Median, Mode
  - Statistics for spread of sample – Standard Deviation, Variance, IQR
  - Statistics for location of data in a sample – Percentiles and Quartiles
- Conclusions about these statistics are not useful by themselves, since *they vary from sample to sample*
- Their real purpose in inferential statistics is in **estimating unknown population parameters and model parameters**
- We begin by understanding how they estimate population parameters, and later extend these concepts to models such as regression models.
- Central to our understanding of inferential statistics is the concept of *sampling distribution of a statistic*.

The Sample as a Collection of iid Random Variables

- Suppose we are interested in establishing the effectiveness of a diet pill, in terms of weight loss over a 28-day period. We would measure the weight loss in a *random sample* of people of 50 who took the pill (as prescribed) and use their weight loss to *infer* the *effectiveness* of the diet pill *in the population*.

- Let us say that the weight loss over a 28-day period (X) in the population follows a normal distribution with an average of $\boldsymbol{\mu} = 10$ pounds and a standard deviation of $\boldsymbol{\sigma} = 5$ pounds.
  - (**Note**: In practice, these are quantities we would **not know**, but are precisely the ones we want to estimate)

- Let us consider the random sample of size 50. Each measurement comes from the same distribution, and being a random sample, each measurement is independent of the other. Thus, our random sample is a collection of *independent, identically distributed* (i.i.d) normal random variables $\{X_1, X_2,\ldots,X_{50}\}$ with each $X_i \sim N(\boldsymbol{\mu} = 10, \boldsymbol{\sigma} = 5)$.

- You can construct such a sample in R using: `sampl1 <- c(rnorm(50, 10, 5)).`
  - rnorm(50, 10, 5) generates 50 observations from $N(\boldsymbol{\mu} = 10, \boldsymbol{\sigma} = 5)$.

- We will construct 2 such sample `sampl1 <- c(rnorm(50, 10, 5)).` and `sampl2 <- c(rnorm(50, 10, 5)).` and look at their descriptive statistics.

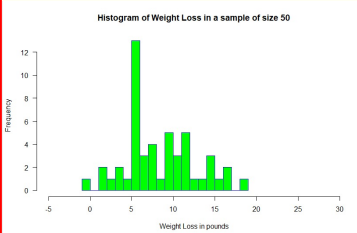
The Sample as a collection of Random Variables – **SamplingDist.R**

- Let us look at the descriptive statistics of this sample and another sample:

```
# First sample - X1 is sampl1
pop_mean = 10
pop_sd = 5
samp_size = 50
X1 <- c(rnorm(samp_size, pop_mean, pop_sd))
X1
#
print(paste("The sample mean is: ", round(mean(X1),4)))
print(paste("The sample standard deviation is: ", round(sd(X1),4)))
print(paste("The sample median is: ", round(median(X1),4)))
print(paste("The sample skewness is: ", round(skewness(X1),4)))
print(paste("The sample kurtosis is: ", round(kurtosis(X1),4)))
h <- hist(X1, main="Histogram of Weight Loss in a sample of size 50",
          xlab="Weight Loss in pounds",
          border="blue",
          col="green",
          xlim=c(-5,30),
          las=1,
          breaks=14)
#
# Second Sample - X2 is sampl2
X2 <- c(rnorm(samp_size, pop_mean, pop_sd))
#
print(paste("The sample mean is: ", round(mean(X2),4)))
print(paste("The sample standard deviation is: ", round(sd(X2),4)))
print(paste("The sample median is: ", round(median(X2),4)))
print(paste("The sample skewness is: ", round(skewness(X2),4)))
print(paste("The sample kurtosis is: ", round(kurtosis(X2),4)))
hist(X2, main="Histogram of Weight Loss in a sample of size 50",
     xlab="Weight Loss in pounds",
     border="blue",
     col="green",
     xlim=c(-5,30),
     las=1,
     breaks=14)
#
```

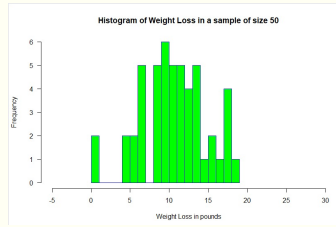
$sampl_1$

```
> #
> print(paste("The sample mean is: ", round(mean(X1),4)))
[1] "The sample mean is:  9.8993"
> print(paste("The sample standard deviation is: ", round(sd(X1),4)))
[1] "The sample standard deviation is:  4.7614"
> print(paste("The sample median is: ", round(median(X1),4)))
[1] "The sample median is:  10.7339"
> print(paste("The sample skewness is: ", round(skewness(X1),4)))
[1] "The sample skewness is:  -0.9059"
> print(paste("The sample kurtosis is: ", round(kurtosis(X1),4)))
[1] "The sample kurtosis is:  4.6267"
```



$sampl_2$

```
> print(paste("The sample mean is: ", round(mean(X2),4)))
[1] "The sample mean is:  9.6557"
> print(paste("The sample standard deviation is: ", round(sd(X2),4)))
[1] "The sample standard deviation is:  5.2037"
> print(paste("The sample median is: ", round(median(X2),4)))
[1] "The sample median is:  10.0022"
> print(paste("The sample skewness is: ", round(skewness(X2),4)))
[1] "The sample skewness is:  -0.1897"
> print(paste("The sample kurtosis is: ", round(kurtosis(X2),4)))
[1] "The sample kurtosis is:  2.2635"
```



What did we learn from Multiple Samples?

- Our goal is to estimate the unknown population mean **μ**.
- Suppose we want to use the *sample mean* statistic $\bar{X}$ to estimate the unknown population mean **μ**.
- BUT,..., we see that each time we take a sample (of size, say, 50), the sample mean is different!
  - For example, in the two samples of size 50 that we took the sample means were 10.57 and 11.81
  - Similarly, other statistics such as sample median, sample standard deviation will be different and the samples will "look" different (Histograms)
- If we had taken 100 samples of size 50, we will get 100 different sample means!
- In practice, you take only one sample and use $\bar{X}$ from that sample to estimate the unknown population mean **μ**.
- How do we know how close that $\bar{X}$ from a single sample is to **μ**?
- To answer that, we need to know *how much $\bar{X}$ varies from sample to sample.*

The Distribution of $\bar{X}$ Across Samples

- The main concept we need to know is that the sample mean $\bar{X}$ (or any sample statistic) has its own distribution, because $\bar{X}$ is a *random variable*!
- This distribution is called the ***sampling distribution*** of $\bar{X}$.
- Remember that $\bar{X} = \frac{\sum_i^n X_i}{n} = \frac{X_1+X_2+,...+X_n}{n}$.
- Since each $X_i \sim N(\mu = 10, \sigma = 5)$ random variable, then the *sum of the* X_i is also a (normally distributed, in this case) random variable.
- The distribution of the sum $\sum_i^n X_i$ is given by $N(n\mu, \sqrt{n}*\sigma)$
- The distribution of the sample mean $\bar{X} = \frac{\sum_i^n X_i}{n}$ is given by $N(\mu, \sigma/\sqrt{n})$ i.e,. $\bar{X} \sim N(\mu, \sigma/\sqrt{n})$.
- This important relation says that, if we took *all possible samples* of size n, and calculated the sample mean for $\bar{X}$ each sample, then:
  - The average of all these sample means, i.e., the Expected Value $(\bar{X}) = \mu$, the unknown population mean we are trying to estimate.
  - The standard deviation of all these sample means, Standard Deviation $(\bar{X}) = \sigma/\sqrt{n}$. This is also called ***Standard Error***, of $\bar{X}$ and is an important quantity in inferential statistics.
- The standard error gives us the answer to the question: *how much does* $\bar{X}$ *varies from sample to sample?*
- If the standard error is small, then $\bar{X}$ does not vary much from sample to sample, AND ANY SAMPLE MEAN should be close the unknown population mean (μ)and therefore should be a good estimator of μ.

A Digression…Sum and Average of *(iid)* Random Variables

- If we take n *independent and identically distributed (iid)* random variables, then:
  - $E(\sum_i^n X_i = X_1 + X_2 +, \ldots + X_n)$ = nE(X), where E(X) is the expected value of any one of those random variables
  - $E(\frac{\sum_i^n X_i}{n} = \frac{X_1 + X_2 +, \ldots + X_n}{n})$ = E($\overline{X}$) = nE(X)/n = E(X), where E(X) is the expected value of any one of those random variables
  - The variance $V(\sum_i^n X_i = X_1 + X_2 +, \ldots + X_n)$ = nV(X), where V(X) is the variance of any one of those random variables
  - The variance $V(\frac{\sum_i^n X_i}{n} = \frac{X_1 + X_2 +, \ldots + X_n}{n})$ = V($\overline{X}$) = V(X), where V(X) is the variance of any one of those random variables
  - The standard deviation s($\overline{X}$) = s(X)/$\sqrt{n}$, where s(X) is the standard deviation of any one of those random variables
- Therefore, if a sample consists on n iid random variables, drawn from a population N($\boldsymbol{\mu}$, $\boldsymbol{\sigma}$), then $\overline{X}$ ~ N($\boldsymbol{\mu}$, $\boldsymbol{\sigma}/\sqrt{n}$). This is the sampling distribution of $\overline{X}$.
  - This tells us that: E($\overline{X}$) = $\boldsymbol{\mu}$, the unknown population we are trying to estimate.

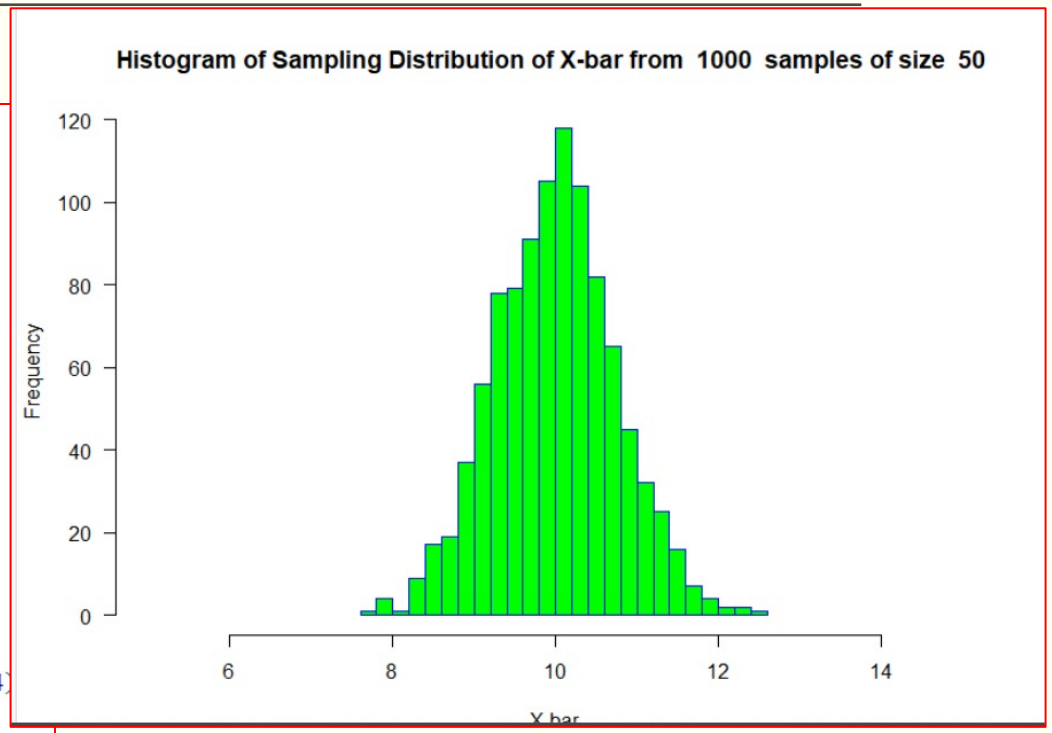
Simulating the Sampling Distribution of $\bar{X}$– SamplingDist.R

- We saw that if X has a normal population N(μ, σ), then $\bar{X}$~ N(μ, $\sigma/\sqrt{n}$). This is the sampling distribution of $\bar{X}$.
- How do we understand the sampling distribution of $\bar{X}$ (or any statistic)?
  - The easiest way to understand is to fix a sample size n, and think of repeatedly taking many, many (almost infinite number of samples).
  - For each sample, calculate the $\bar{X}$. The collection of $\bar{X}$'s from all these samples forms the sampling distribution.
- Let us simulate the sampling distribution for our example:
  - Take 1000 samples of size 50 from N(10, σ =5), and computing the $\bar{X}$ for each sample.
  - Look at the distribution of $\bar{X}$'s from these 1000 samples.
- Given $\bar{X}$~ N(μ, $\sigma/\sqrt{n}$), we should expect the mean of all the $\bar{X}$ to be 10 and standard deviation = $5/\sqrt{50}$ = 0.7071.

Simulating the Sampling Distribution of $\bar{X}$- SamplingDist.R

Given population is N(**10**, **5**)

```
# Generating an empirical sampling distribution of sample mean - X-bar
# Define the x_bar vector
num_samp = 1000
samp_size = 50
x_bar <- vector("numeric", num_samp)
#
# Each x-bar is the mean of a random sample of size 50 drawn from a Normal(10, 5)
#
# We are generating 1000 X-bars (from 1000 samples) and storing them in the x-bar vector
#
for (i in 1:num_samp) {
  x_bar[i] = mean(rnorm(samp_size, pop_mean, pop_sd))
}
#
# Calculate the empirical mean and emoirical standard deviation (called standard error) of x-bar
# from the empirical sampling distribution formed by 1000 samples
#
Exp_x_bar <- mean(x_bar)
stderr <- sd(x_bar)
print(paste("The Empirical Expected value of X-bar is: ", round(Exp_x_bar,4)))
print(paste("The Theoretical Expected value of X-bar is: ", pop_mean))
print(paste("The Empirical Standard Error or standard deviation of X-bar is: ", round(stderr,4)))
print(paste("The Theoretical Standard Error or standard deviation of X-bar is: ", round(pop_sd/sqrt(samp_size),4)
hist(x_bar,
     main=paste("Histogram of Sampling Distribution of X-bar from ",num_samp,
                " samples of size ", samp_size, ""),
     xlab="X-bar",
     border="blue",
     col="green",
     xlim=c(5, 15),
     las=1,
     breaks=20)
#
```

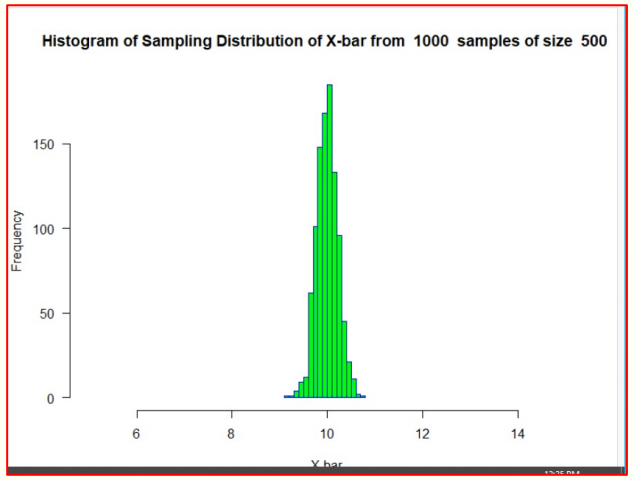



```
> # Calculate the empirical mean and emoirical standard deviation (called standard error) of x-bar
> # from the empirical sampling distribution formed by 1000 samples
> #
> Exp_x_bar <- mean(x_bar)
> stderr <- sd(x_bar)
> print(paste("The Empirical Expected value of X-bar is: ", round(Exp_x_bar,4)))
[1] "The Empirical Expected value of X-bar is:  9.9934"
> print(paste("The Theoretical Expected value of X-bar is: ", pop_mean))
[1] "The Theoretical Expected value of X-bar is:  10"
> print(paste("The Empirical Standard Error or standard deviation of X-bar is: ", round(stderr,4)))
[1] "The Empirical Standard Error or standard deviation of X-bar is:  0.7444"
> print(paste("The Theoretical Standard Error or standard deviation of X-bar is: ", round(pop_sd/sqrt(samp_size),4)))
[1] "The Theoretical Standard Error or standard deviation of X-bar is:  0.7071"
```

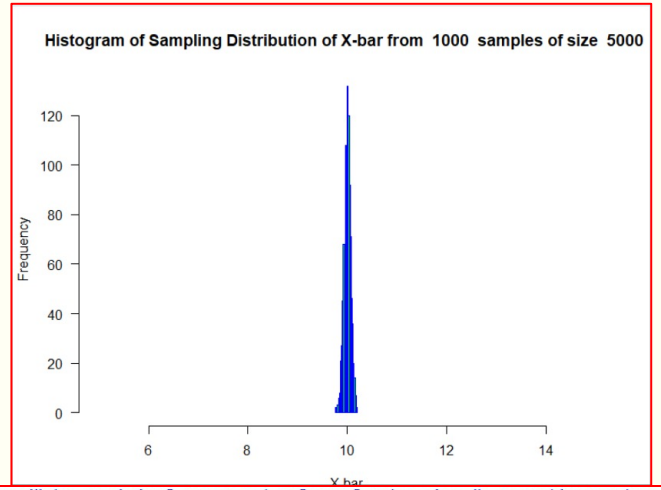
Given $\bar{X} \sim N(\mu, \sigma/\sqrt{n})$, we should expect the mean of all the $\bar{X}$ to be **10** and standard deviation = $5/\sqrt{50}$ = **0.7071**

The Effect of Sample Size on Standard Error

- We saw that if X has a normal population N(μ, σ), then $\bar{X}$~ N(μ, $\sigma/\sqrt{n}$). This is the sampling distribution of $\bar{X}$.
- This tells us that as sample size increases, the standard error of $\bar{X}$ (the standard deviation of its sampling distribution) will decrease and variability of the samples around the true population mean (μ) will also decrease.
- This means the sample mean of every sample gets closer to the true population mean.
- Let us simulate this for sample sizes 500 and 5000 and we will see that it is true.




```
> print(paste("The Empirical Expected value of X-bar is: ", round(Exp_x_bar,4)))
[1] "The Empirical Expected value of X-bar is:  9.9942"
> print(paste("The Theoretical Expected value of X-bar is: ", pop_mean))
[1] "The Theoretical Expected value of X-bar is:  10"
> print(paste("The Empirical Standard Error or standard deviation of X-bar is: ", round(stderr,4)))
[1] "The Empirical Standard Error or standard deviation of X-bar is:  0.2219"
> print(paste("The Theoretical Standard Error or standard deviation of X-bar is: ", round(pop_sd/sqrt(samp_size),4)))
[1] "The Theoretical Standard Error or standard deviation of X-bar is:  0.2236"
```




```
> print(paste("The Empirical Expected value of X-bar is: ", round(Exp_x_bar,4)))
[1] "The Empirical Expected value of X-bar is:  10.0033"
> print(paste("The Theoretical Expected value of X-bar is: ", pop_mean))
[1] "The Theoretical Expected value of X-bar is:  10"
> print(paste("The Empirical Standard Error or standard deviation of X-bar is: ", round(stderr,4)))
[1] "The Empirical Standard Error or standard deviation of X-bar is:  0.0685"
> print(paste("The Theoretical Standard Error or standard deviation of X-bar is: ", round(pop_sd/sqrt(samp_size),4)))
[1] "The Theoretical Standard Error or standard deviation of X-bar is:  0.0707"
```

Standard Error and the Inferential Validity of the Sample Mean

- We can calculate the probability of a *sample mean* being a certain distance (number of standard errors) away from the **population mean**.
- In other words, we can calculate the percentage of all possible samples in which the sample mean is within a certain distance of the population mean
- To do this, we use the sampling distribution of the sample mean; in this case, $\bar{X} \sim N(\mu, = 10\ \sigma = 5/\sqrt{n})$.
- Theoretically for sample size 50:

```
> #
> # Theoretically, 80% of X-bars are below:
> print(paste("Theoretically, 80% of X-bars are between: ", qnorm(0.1, pop_mean, pop_sd/sqrt(samp_size)), " and ", qnorm(0.9, pop_mean, pop_sd/sqrt(samp_size))))
[1] "Theoretically, 80% of X-bars are between:  9.09380619756318  and  10.9061938024368"
>
```

- Empirically for sample size 50:

```
> # Empirically, 80% of X-bars are below:
> print(paste("Empirically, 80% of X-bars are between: ", quantile(x_bar, probs = 0.1, na.rm=FALSE, names = TRUE, type=2)," and ",quantile(x_bar, probs = 0.9, na.rm=FALSE, names = TRUE, type=2) ))
[1] "Empirically, 80% of X-bars are between:  9.09735900636646  and  10.9428948351063"
```

Standard Error and the Inferential Validity of the Sample Mean

- For sample size 500:

```
> # Theoretically, 80% of X-bars are below:
> print(paste("Theoretically, 80% of X-bars are between: ", qnorm(0.1, pop_mean, pop_sd/sqrt(samp_size)), " and ", qnorm(0.9, pop_mean, pop_sd/sqrt(samp_size))))
[1] "Theoretically, 80% of X-bars are between:  9.7134363582771  and  10.2865636417229"
>
> # Empirically, 80% of X-bars are below:
> print(paste("Empirically, 80% of X-bars are between: ", quantile(x_bar, probs = 0.1, na.rm=FALSE, names = TRUE, type=2)," and ",quantile(x_bar, probs = 0.9, na.rm=FALSE, names = TRUE, type=2) ))
[1] "Empirically, 80% of X-bars are between:  9.7155658823236  and  10.2664991994121"
```

- For sample size 5000:

```
> # Theoretically, 80% of X-bars are below:
> print(paste("Theoretically, 80% of X-bars are between: ", qnorm(0.1, pop_mean, pop_sd/sqrt(samp_size)), " and ", qnorm(0.9, pop_mean, pop_sd/sqrt(samp_size))))
[1] "Theoretically, 80% of X-bars are between:  9.90938061975632  and  10.0906193802437"
>
> # Empirically, 80% of X-bars are below:
> print(paste("Empirically, 80% of X-bars are between: ", quantile(x_bar, probs = 0.1, na.rm=FALSE, names = TRUE, type=2)," and ",quantile(x_bar, probs = 0.9, na.rm=FALSE, names = TRUE, type=2) ))
[1] "Empirically, 80% of X-bars are between:  9.91507012634236  and  10.0968040157318"
```

- The moral of the story is that:
  - When standard error is small, the sample mean of ***any sample*** gets closer and to the true population mean (as well as to the other sample means). So the any such sample has good validity in using the *sample mean to estimate an unknown population mean* i.e., good inferential power and validity.
  - *Increase in sample size, decreases the standard error, and increases the validity of such inferences* .