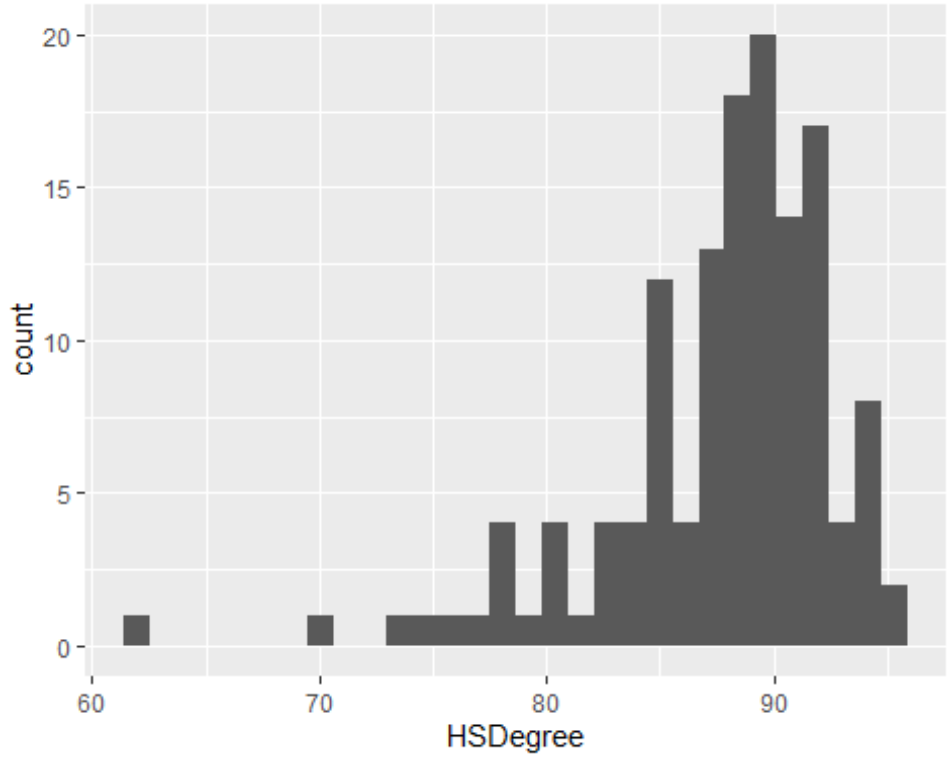
AmericanCommunitySurveyExercise_Torres_Gloria.R

2023-04-02

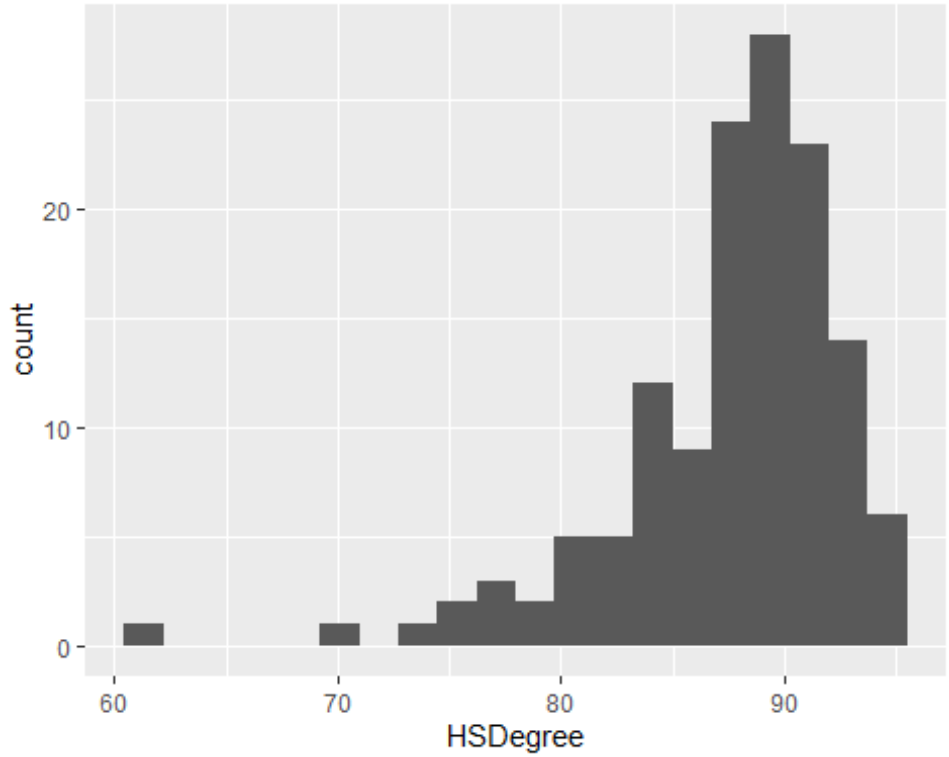
```
# Assignment: American Community Survey Exercise
# Name: Torres, Estrada
# Date: 2023-03-31

# Activate the ggplot2 package
library(ggplot2)

# List the name of each field and what you believe the data type and intent
is of the data included in each field (Example: Id - Data Type: varchar
(contains text and numbers) Intent: unique identifier for each row)

  # Id = National County ID.  Intent: Identify data by the ID.
  # Id = County ID number.  Intent: Filter data by county  number
  # Geography = County and State. Intent: Pull information and understand
data  by county and state
  # PopGroupID = Group of population. Intent: Understand and pull data by
group
  # POPGROUP.display-label = Description of the group Id. Intent: Identify
the groups(Hispanic, White, etc.)
  # RacesReported = Number of people who reported his race. Intent:
understand the number of records
  # HSDegree = Highchool Deploma. Intent: Displays the percentage of people
that finished high school per county
  # BachDegree = bachelor's degree. Intend: Displays the percentage of people
with  a bachelor's degree per county



# Create a Histogram of the HSDegree variable using the ggplot2 package.

setwd('C:/Users/glori/OneDrive/Documents/Gloria GIT/Gloria_Torres_DSC_520')

ac_survey <- read.csv("data/acs-14-1yr-s0201.csv")

# Run the following functions and provide the results: str(); nrow(); ncol()

str(ac_survey)
```

```
## 'data.frame':    136 obs. of  8 variables:
##  $ Id                    : chr  "0500000US01073" "0500000US04013"
"0500000US04019" "0500000US06001" ...
```

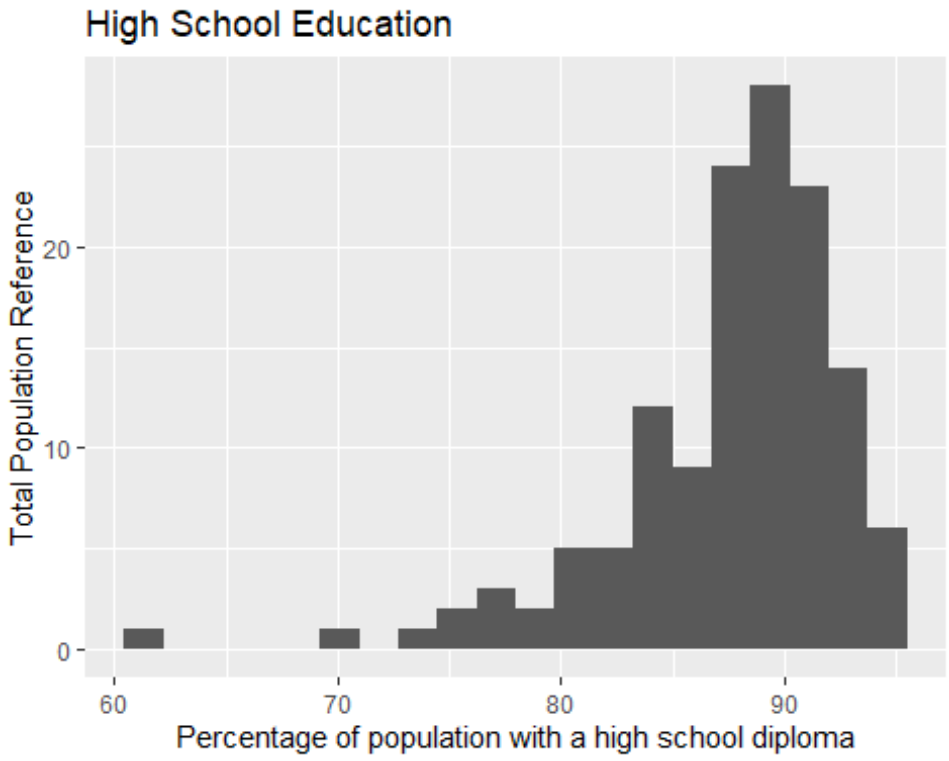
```
##  $ Id2                   : int  1073 4013 4019 6001 6013 6019 6029 6037
6059 6065 ...
##  $ Geography             : chr  "Jefferson County, Alabama" "Maricopa
County, Arizona" "Pima County, Arizona" "Alameda County, California" ...
##  $ PopGroupID            : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ POPGROUP.display.label: chr  "Total population" "Total population"
"Total population" "Total population" ...
##  $ RacesReported         : int  660793 4087191 1004516 1610921 1111339
965974 874589 10116705 3145515 2329271 ...
##  $ HSDegree              : num  89.1 86.8 88 86.9 88.8 73.6 74.5 77.5 84.6
80.6 ...
##  $ BachDegree            : num  30.5 30.2 30.8 42.8 39.7 19.7 15.4 30.3 38
20.7 ...

nrow(ac_survey)

## [1] 136

ncol(ac_survey)

## [1] 8

ggplot(ac_survey, aes(HSDegree)) + geom_histogram()

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```r
# Set a bin size for the Histogram that you think best visuals the data (the bin size will determine how many bars display and how wide they are)
ggplot(ac_survey, aes(HSDegree)) + geom_histogram(bins = 20)
```




```r
# Include a Title and appropriate X/Y axis labels on your Histogram Plot.
ggplot(ac_survey, aes(HSDegree)) + geom_histogram(bins = 20)+ ggtitle('High School Education') + xlab('Percentage of population with a high school diploma') + ylab('Total Population Reference')
```




```
# Answer the following questions based on the Histogram produced:
# Based on what you see in this histogram, is the data distribution unimodal?
# The data distribution is not unimodal. Unimodal distribution has only one
clear peak. Also, I ran the following function to confirm.
library(LaplacesDemon)
is.unimodal(ac_survey)
```

```
## Warning in as.vector(as.numeric(as.character(x))): NAs introduced by
coercion

## Warning in as.vector(as.numeric(as.character(x))): NAs introduced by
coercion

## [1] FALSE
```

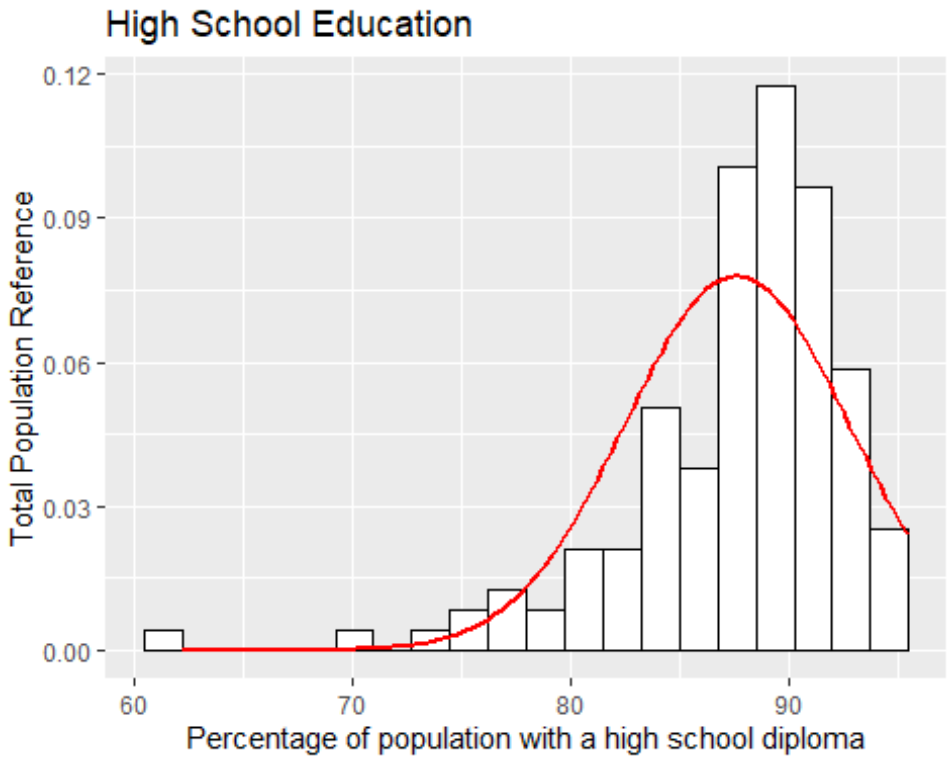
```
# Is it approximately symmetrical?
# No, it is not approximately symmetrical (the left and right sides do not
look the same).

# Is it approximately bell-shaped?
# No, it is not symmetric around its center

# Is it approximately normal?
# No, but sometimes is hard to identify normal distribution from the
histograms.

# If not normal, is the distribution skewed? If so, in which direction?
```

```
# yes, it is left-skewed (Negative skewed).

# Include a normal curve to the Histogram that you plotted.
normal_curve <- ggplot(ac_survey, aes(HSDegree)) + geom_histogram(bins = 20,
aes(y = ..density..), fill ='white', colour='black') + stat_function(fun =
dnorm, args = list(mean = mean(ac_survey$HSDegree, na.rm = TRUE),
sd=sd(ac_survey$HSDegree, na.rm = TRUE)), colour ="red",  size=1) +
ggtitle('High School Education') + xlab('Percentage of population with a high
school diploma') + ylab('Total Population Reference')
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
normal_curve
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2
3.4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

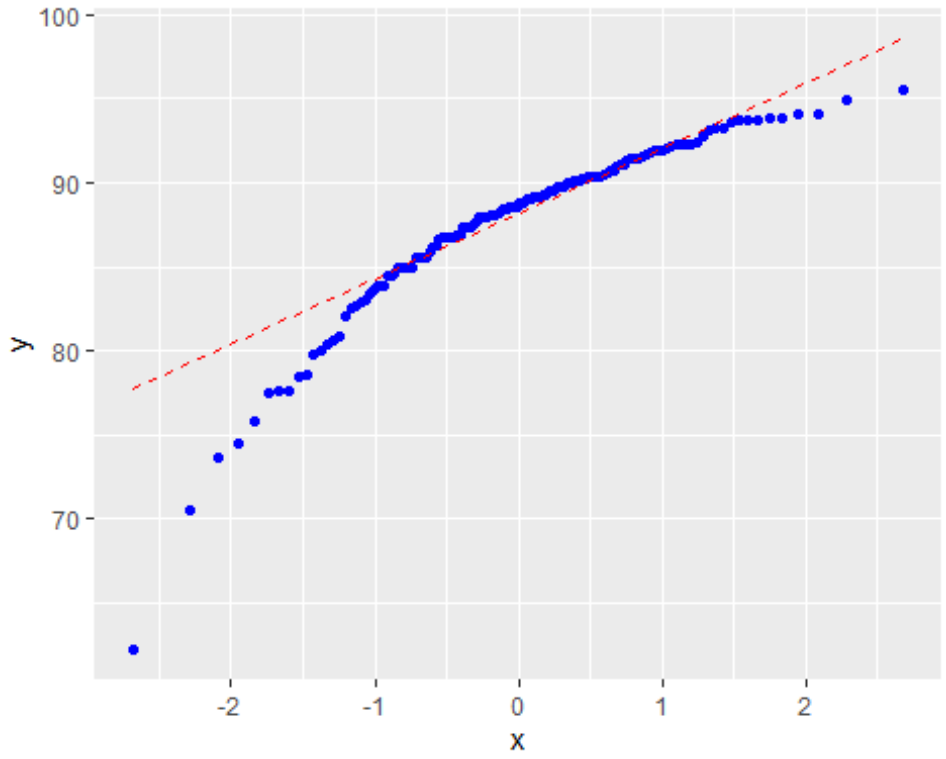



```
# Explain whether a normal distribution can accurately be used as a model for
this data.
```

```r
# The normal distribution model is not a  good choice in this case since the distribution is skewed to the left; it is not bell-shaped.

# Create a Probability Plot of the HSDegree variable.

ggplot(ac_survey, aes(sample=HSDegree))
```



```r
p <- ggplot(ac_survey, aes(sample=HSDegree))
p + stat_qq(col="blue") + stat_qq_line(col="red", lty=2)
```




```
# Answer the following questions based on the Probability Plot:
# Based on what you see in this probability plot, is the distribution
approximately normal? Explain how you know.
# The distribution approximately is not normal; the plot shows some deviation
from normal (it looks like a snake).

# If not normal, is the distribution skewed? If so, in which direction?
Explain how you know.
 # yes, it is skewed to the right.

# Now that you have looked at this data visually for normality, you will now
quantify normality with numbers using the stat.desc() function. Include a
screen capture of the results produced.
library(pastecs)
stat.desc(ac_survey$HSDegree, basic = FALSE, norm = TRUE)

##        median          mean       SE.mean  CI.mean.0.95           var
##  8.870000e+01  8.763235e+01  4.388598e-01  8.679296e-01  2.619332e+01
##       std.dev      coef.var      skewness      skew.2SE      kurtosis
##  5.117941e+00  5.840241e-02 -1.674767e+00 -4.030254e+00  4.352856e+00
##      kurt.2SE    normtest.W    normtest.p
##  5.273885e+00  8.773635e-01  3.193634e-09

# In several sentences provide an explanation of the result produced for
skew, kurtosis, and z-scores. In addition, explain how a change in the sample
size may change your explanation?
# Skew:  When the skew = 0, the distribution is symmetrical. In the above
```

```
situation, the mean should equal the median. In this case, the distribution
doesn't equal 0, and the mean and median do not have the same result.
# Media: 8.870000e+01
# Mean: 8.763235e+01
# Skewness: -1.674767e+00
# Skew.2SE: -4.030254e+00
# # Kurtosis: 4.352856e+00 indicates that distribution is peaked. Higher
kurtosis means a high tail. Kurtosis > 3 is known as Leptokurtic.
# # z-scores:
# Z-score helps to understand the difference between the raw score value and
the mean in terms of standard deviation.
# In this case, the standard deviation is 5.117941e+00, and the mean =
8.763235e+01; the above results will help us to obtain the z – Scores.
# If the standard deviation is > 3, the normal distribution curve will fall
to the right.
# The data size matters: For example, If the data sample size increases, the
new data can be different from the actual one, which can change the median,
the mean, the skewness, etc. For this sample of data, Johnson County, Kansas,
has the highest rate of population with a high school degree (95.5%), vs.
Hidalgo County, Texas, where only 62.2% of the population has a high school
diploma. Still, if counties with high poverty get added to the sample data,
the level of education could decrease


# Calculate z - scores
data <- c(ac_survey$HSDegree)
mean(data)

## [1] 87.63235

sd(data)

## [1] 5.117941

zscore <- (data - mean(data)) / sd(data)
zscore

##   [1]  0.286765161 -0.162634350  0.071834960 -0.143095241  0.228147834
##   [6] -2.741796762 -2.565944779 -1.979771504 -0.592494752 -1.374059119
##  [11] -0.162634350 -1.764841303 -0.201712568  0.091374069 -1.960232394
##  [16]  0.091374069 -0.045399695 -0.006321476 -1.803919521 -0.787885844
##  [21]  0.833860218 -0.416642769  1.009712201  1.263720620  0.423538925
##  [26]  0.325843380  0.364921598  0.482156253  0.501695362  0.775242891
##  [31]  0.149991397  0.267226052 -0.064938804 -0.260329896 -1.315441791
##  [36]  0.052295851  0.013217633  0.482156253 -0.533877424  0.247686943
##  [41]  0.521234471  0.149991397  0.716625563  0.071834960  0.814321109
##  [46] -0.416642769  0.912016655 -0.924659608  0.521234471  0.599390908
##  [51] -0.514338315  1.537268149  0.228147834  0.169530506  0.833860218
##  [56]  0.540773581  0.638469126 -0.416642769 -0.631572970 -1.002816045
##  [61]  0.286765161  0.912016655  1.263720620  0.892477546 -0.729268516
##  [66]  0.482156253  0.286765161  0.325843380  1.166025074 -0.533877424
```

```
##  [71]  1.087868638  0.443078035  0.462617144  1.087868638  0.110913179
##  [76] -0.612033861  0.755703781  0.130452288 -0.416642769 -0.826964062
##  [81]  0.286765161  1.068329528  0.794782000 -0.748807625 -0.279869005
##  [86]  0.071834960 -3.347509146  0.579851799 -1.491293774  0.521234471
##  [91]  0.599390908 -0.162634350 -1.413137337  0.423538925 -0.045399695
##  [96]  0.267226052  0.364921598  0.931555764  0.091374069  0.462617144
## [101]  0.560312690  0.403999816  0.677547345 -0.162634350  0.189069615
## [106]  0.677547345  0.501695362  1.224642402  1.224642402  0.912016655
## [111]  0.755703781 -0.533877424  1.185564183 -0.983276935 -1.100511591
## [116] -0.182173459 -0.045399695 -0.905120499  1.185564183 -1.960232394
## [121]  0.833860218 -2.311936360  0.189069615 -1.530371992 -4.969255208
## [126] -0.338486333 -0.533877424  0.189069615  0.364921598  1.185564183
## [131]  0.755703781  0.912016655  0.521234471  0.853399327  1.420033494
## [136] -0.143095241
```