assignment4.2_Chattapadhyay_Kausik.R

kausik

2022-09-22

```
# Assignment: ASSIGNMENT 4.2
# Name: Chattapadhyay, Kausik
# Date: 2022-09-21

## Load the ggplot2 package
library(ggplot2)
library(plyr)
library(dplyr)
library(readxl)
theme_set(theme_minimal())

## Set the working directory to the root of your DSC 520 directory
setwd("/Users/kausik/desktop/MS Data Science/DSC 520/dsc520-stats-r-assignments")

## Load the `2014 American Community Survey` to
survey_df <- read.csv("data/acs-14-1yr-s0201.csv")
head(survey_df)
```

```
##               Id  Id2                      Geography PopGroupID
## 1 0500000US01073 1073       Jefferson County, Alabama          1
## 2 0500000US04013 4013        Maricopa County, Arizona          1
## 3 0500000US04019 4019            Pima County, Arizona          1
## 4 0500000US06001 6001      Alameda County, California          1
## 5 0500000US06013 6013 Contra Costa County, California          1
## 6 0500000US06019 6019       Fresno County, California          1
##   POPGROUP.display.label RacesReported HSDegree BachDegree
## 1       Total population        660793     89.1       30.5
## 2       Total population       4087191     86.8       30.2
## 3       Total population       1004516     88.0       30.8
## 4       Total population       1610921     86.9       42.8
## 5       Total population       1111339     88.8       39.7
## 6       Total population        965974     73.6       19.7
```

```
## Load the `Housing dataset' to
housing_df <- read_excel("data/week-7-housing.xlsx", sheet="Sheet2")

# Renamimg the field names
colnames(housing_df)[2] <- "Sale_Price"
colnames(housing_df)[1] <- "Sale_Date"
str(housing_df)
```

```
## tibble [12,865 x 24] (S3: tbl_df/tbl/data.frame)
##  $ Sale_Date               : POSIXct[1:12865], format: "2006-01-03" "2006-01-03" ...
##  $ Sale_Price              : num [1:12865] 698000 649990 572500 420000 369900 ...
##  $ sale_reason             : num [1:12865] 1 1 1 1 1 1 1 1 1 1 ...
##  $ sale_instrument         : num [1:12865] 3 3 3 3 3 15 3 3 3 3 ...
##  $ sale_warning            : chr [1:12865] NA NA NA NA ...
##  $ sitetype                : chr [1:12865] "R1" "R1" "R1" "R1" ...
##  $ addr_full               : chr [1:12865] "17021 NE 113TH CT" "11927 178TH PL NE" "13315 174TH AVE
##  $ zip5                    : num [1:12865] 98052 98052 98052 98052 98052 ...
##  $ ctyname                 : chr [1:12865] "REDMOND" "REDMOND" NA "REDMOND" ...
##  $ postalctyn              : chr [1:12865] "REDMOND" "REDMOND" "REDMOND" "REDMOND" ...
##  $ lon                     : num [1:12865] -122 -122 -122 -122 -122 ...
##  $ lat                     : num [1:12865] 47.7 47.7 47.7 47.6 47.7 ...
##  $ building_grade          : num [1:12865] 9 9 8 8 7 7 10 10 9 8 ...
##  $ square_feet_total_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...
##  $ bedrooms                : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...
##  $ bath_full_count         : num [1:12865] 2 2 1 1 1 2 3 2 2 1 ...
##  $ bath_half_count         : num [1:12865] 1 0 1 0 0 1 0 1 1 0 ...
##  $ bath_3qtr_count         : num [1:12865] 0 1 1 1 1 1 1 0 1 1 ...
##  $ year_built              : num [1:12865] 2003 2006 1987 1968 1980 ...
##  $ year_renovated          : num [1:12865] 0 0 0 0 0 0 0 0 0 0 ...
##  $ current_zoning          : chr [1:12865] "R4" "R4" "R6" "R4" ...
##  $ sq_ft_lot               : num [1:12865] 6635 5570 8444 9600 7526 ...
##  $ prop_type               : chr [1:12865] "R" "R" "R" "R" ...
##  $ present_use             : num [1:12865] 2 2 2 2 2 2 2 2 2 2 ...
```

```
## A. Use the apply function on a variable in your dataset
# checking if any NAs on sale price.
any(is.na(housing_df$Sale_Price))
```

```
## [1] FALSE
```

```
# apply() on sale price to get mean sale price.
apply(data.frame(housing_df$Sale_Price), 2, mean)
```

```
## housing_df.Sale_Price
##              660737.7
```

```
# Zip code wise average sale using ddply()
avg_sale <- function(data) {
    c(avg_sale = with(data, mean(Sale_Price)))
}
ddply(housing_df, .variables = "zip5", .fun = avg_sale)
```

```
##    zip5 avg_sale
## 1 98052 649375.4
## 2 98053 672623.7
## 3 98059 645000.0
## 4 98074 951543.8
```

```r
# Date wise average sale using ddply()
abp <- ddply(housing_df, .variables = "Sale_Date", .fun = avg_sale)
head(abp)
```

```
##    Sale_Date avg_sale
## 1 2006-01-03 482509.5
## 2 2006-01-04 624592.1
## 3 2006-01-05 655475.0
## 4 2006-01-06 677475.0
## 5 2006-01-09 436750.0
## 6 2006-01-10 497631.0
```

```r
abp <- abp[order(abp$avg_sale, decreasing = TRUE),]
head(abp, 10)
```

```
##       Sale_Date avg_sale
## 1544 2011-11-17  3844292
## 1827 2012-11-30  3000000
## 1779 2012-10-02  2916391
## 738  2008-10-01  2880684
## 672  2008-07-01  2806134
## 1085 2010-03-02  2787500
## 2693 2016-02-05  2333000
## 584  2008-02-26  2245000
## 580  2008-02-20  1990000
## 1493 2011-09-13  1987176
```

```r
## B. Use the aggregate function on a variable in your dataset
# Aggregate sale price by zip codes.
aggregate(Sale_Price ~ zip5, housing_df, each(mean, median))
```

```
##    zip5 Sale_Price.mean Sale_Price.median
## 1 98052        649375.4          599950.0
## 2 98053        672623.7          584000.0
## 3 98059        645000.0          645000.0
## 4 98074        951543.8          820000.0
```

```r
# Aggregate sale price by year built.
aggregate(cbind(Sale_Price, sq_ft_lot) ~ year_built, housing_df, each(mean, median))
```

```
##     year_built Sale_Price.mean Sale_Price.median sq_ft_lot.mean sq_ft_lot.median
## 1         1900        394499.7          427500.0     305114.667       221720.000
## 2         1903        430000.0          430000.0      85377.000        85377.000
## 3         1905        620000.0          620000.0      22237.000        22237.000
## 4         1906        550000.0          550000.0      37026.000        37026.000
## 5         1909          1070.0            1070.0     221284.000       221284.000
## 6         1910        150000.0          150000.0      13064.000        13064.000
## 7         1912        619666.7          580000.0      83646.000        67953.000
## 8         1913        457500.0          457500.0      64810.500        64810.500
## 9         1914        835000.0          835000.0     138085.000       138085.000
## 10        1915        228150.0          228150.0       5917.000         5917.000
```

```
## 11        1916        350000.0        350000.0     389426.000      389426.000
## 12        1918       1033833.3       1200000.0      22027.000       14043.000
## 13        1919        476800.0        476800.0      43700.000       43700.000
## 14        1920        509083.3        522500.0      59333.333       31342.000
## 15        1922        424587.5        386675.0      29890.000       34099.500
## 16        1923        300000.0        300000.0     297950.000      297950.000
## 17        1924        649500.0        636500.0      63799.500       53780.500
## 18        1925        387250.0        402000.0      10988.750       11243.000
## 19        1926        318333.3        255000.0     151686.333       13650.000
## 20        1927       1173750.0       1282500.0      69141.000       10150.000
## 21        1928        520000.0        520000.0      21740.000       21740.000
## 22        1929       1242500.0       1242500.0      10046.000       10046.000
## 23        1930        402191.7        360000.0      53147.333       15930.000
## 24        1931        168828.5        168828.5     192535.000      192535.000
## 25        1932        588146.2        487031.0      48445.500       31820.500
## 26        1933        440500.0        465000.0      97481.167       56627.500
## 27        1934        750000.0        782500.0     427892.750      427998.000
## 28        1935       1616333.3        339000.0     516430.000      122403.000
## 29        1936        485182.3        430000.0      59216.000       83199.000
## 30        1937        846594.3        338750.0      81015.000       28717.000
## 31        1938       1675500.0       1675500.0      20676.500       20676.500
## 32        1939        520000.0        520000.0     123046.000      123046.000
## 33        1940        681411.1        520000.0      58126.889       21780.000
## 34        1941        348517.2        460000.0     109571.800       75358.000
## 35        1942        343561.0        392000.0     132876.250       42870.500
## 36        1943        501200.0        425000.0     196464.400       25703.000
## 37        1944        335626.5        335626.5      56283.000       56283.000
## 38        1945        354330.9        323250.0      26765.875       13760.000
## 39        1946        626875.0        637500.0     112280.000      112280.000
## 40        1947        390378.7        401000.0      37218.444       32300.000
## 41        1948        713522.6        605500.0     201448.700      113624.500
## 42        1949        485525.4        427350.0      55900.167       15987.000
## 43        1950        360315.0        360000.0      44615.556       33103.000
## 44        1951        583972.0        515000.0     134613.333       22215.000
## 45        1952        786191.7        500000.0      64979.067       36396.000
## 46        1953        463553.7        434000.0      43245.667       12369.000
## 47        1954        657591.3        530000.0     145532.222      130680.000
## 48        1955        563706.3        482500.0      38642.321       16000.000
## 49        1956        625561.5        550000.0     152507.462       24000.000
## 50        1957        511411.5        475000.0      64916.308       37026.000
## 51        1958        428233.8        440000.0      34512.105        8925.000
## 52        1959        468616.6        427500.0      21027.217       11146.000
## 53        1960        451005.4        448000.0      53451.222       14937.500
## 54        1961        581580.0        516252.0      43817.464       12635.000
## 55        1962        515826.5        435000.0      23405.327       10275.000
## 56        1963        508518.7        460000.0      20307.517        8976.000
## 57        1964        566355.5        461200.0      18501.827        8435.000
## 58        1965        484418.3        470000.0      12956.383        9100.000
## 59        1966        478482.7        465000.0      11897.468        8560.000
## 60        1967        497566.3        479950.0      15235.978        9600.000
## 61        1968        446930.1        439975.0      12030.409        8730.000
## 62        1969        444439.2        429725.0      18478.059        9600.000
## 63        1970        419788.3        391000.0      16112.033       10858.000
## 64        1971        442688.5        442000.0      14473.361        9361.000
```

```
## 65        1972        552177.1          543500.0      23434.274        10231.500
## 66        1973        556947.5          551017.0      35407.920         9923.500
## 67        1974        591669.8          539500.0      18401.594        11427.000
## 68        1975        535944.1          520000.0      14987.628         9310.000
## 69        1976        502248.9          495000.0      16440.098         9594.000
## 70        1977        494102.5          475000.0      19096.851         9600.000
## 71        1978        512763.1          485000.0      18466.409         9000.000
## 72        1979        545454.4          520000.0      28360.098         9800.000
## 73        1980        546471.3          520000.0      29383.869        11316.000
## 74        1981        539075.9          520000.0      41166.338        15711.000
## 75        1982        586006.0          527000.0      23019.095        10455.000
## 76        1983        527091.5          520000.0      24646.185        11892.000
## 77        1984        561059.2          540000.0      23390.579         9883.000
## 78        1985        599990.3          560000.0      27638.743        10760.000
## 79        1986        583642.8          555000.0      27701.204         8997.500
## 80        1987        662669.3          608000.0      35762.199        20000.000
## 81        1988        774747.3          744350.0      41371.646        32578.000
## 82        1989        762350.0          750000.0      47709.249        35557.000
## 83        1990        837696.4          767500.0      39257.395        35220.000
## 84        1991        807708.3          765000.0      45915.959        36046.000
## 85        1992        630408.5          609250.0      30946.386        25594.000
## 86        1993        700939.1          685000.0      32338.845        21781.000
## 87        1994        752529.6          736250.0      43243.139        32049.500
## 88        1995        694532.9          650000.0      38558.825        27227.000
## 89        1996        689408.3          675000.0      38062.080        26071.000
## 90        1997        738764.9          720500.0      52594.764        27012.000
## 91        1998        791991.1          752500.0      43295.603        23362.000
## 92        1999       1016032.6          860000.0     109805.127        22061.000
## 93        2000        829172.7          715000.0      28534.084         5846.000
## 94        2001        695094.1          595000.0      16339.512         5460.000
## 95        2002        599826.2          567000.0       6989.432         5077.000
## 96        2003        645323.4          595000.0      17847.412         5602.000
## 97        2004        632882.3          620000.0       8189.966         5562.000
## 98        2005        647728.2          622495.0       9225.573         5340.500
## 99        2006        692548.0          672000.0      12368.120         5885.000
## 100       2007        664465.2          656000.0      14462.150         6126.000
## 101       2008        866785.5          645470.0      14834.730         5141.000
## 102       2009        756906.6          616580.5       9041.452         4991.500
## 103       2010        649072.9          617750.0       8721.975         4885.500
## 104       2011        677745.2          626675.0      13513.599         5683.500
## 105       2012        922800.5          663900.0       7519.669         5580.000
## 106       2013        912130.4          705907.0      12459.934         5292.000
## 107       2014        825761.6          853990.0       9935.768         6238.000
## 108       2015        888559.7          940445.0      42651.718         7053.000
## 109       2016        893875.0          904480.5      21408.975         5856.500
```

```
# Aggregate sale price by lat, lon.
head(aggregate(Sale_Price ~ lat + lon, housing_df, each(mean, median)))
```

```
##        lat       lon Sale_Price.mean Sale_Price.median
## 1 47.69797 -122.1643          336300            336300
## 2 47.69778 -122.1643          315000            315000
## 3 47.69914 -122.1642         2150000           2150000
## 4 47.70280 -122.1641          424875            424875
```

```
## 5 47.70652 -122.1641          571000            571000
## 6 47.70622 -122.1640          390000            390000
```

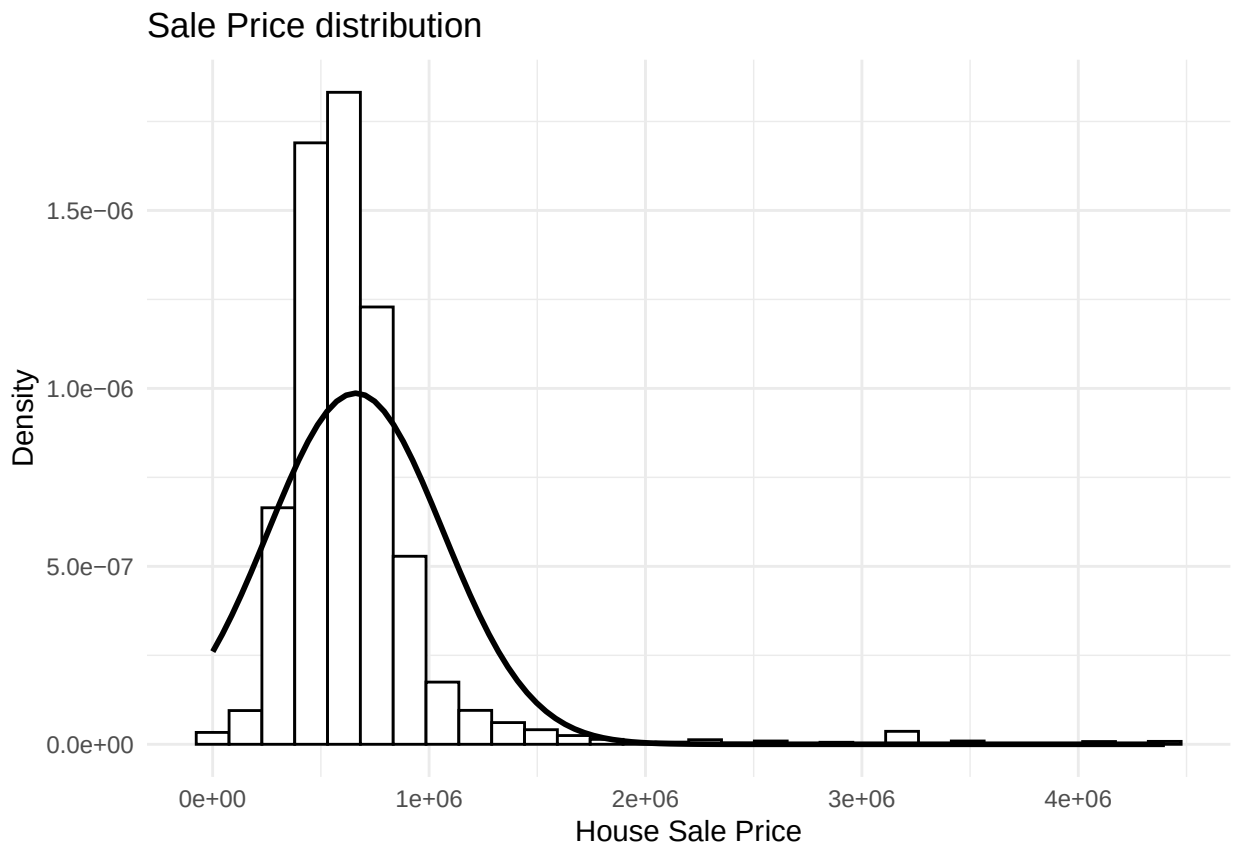
```
## C. Use the plyr function on a variable in your dataset - more specifically,
## I want to see you split some data, perform a modification to the data, and
## then bring it back together.
# Extracting only 2016 data
housing_df_2016 <- housing_df[housing_df$Sale_Date >= '2016-01-01',]
# Function to create per square ft sale price
avg_sale <- function(data) {
    c(avg_sale = with(data, mean(Sale_Price)))
}

# apply ddply() to split the data , perform action and return data frame
ddply(housing_df, .variables = "zip5", .fun = avg_sale)
```

```
##    zip5 avg_sale
## 1 98052 649375.4
## 2 98053 672623.7
## 3 98059 645000.0
## 4 98074 951543.8
```

```
## D. Check distributions of the data
#
ggplot(housing_df, aes(x=Sale_Price)) +
    labs(x = "House Sale Price", y = "Density", title = "Sale Price distribution") +
    geom_histogram(aes(y=..density..), color="black", fill="white", show.legend = F) +
    stat_function(fun=dnorm, args = list(mean = mean(housing_df$Sale_Price, na.rm = T),
    sd = sd(housing_df$Sale_Price, na.rm = T)), color="black", size=1)
```

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




```
## E. Identify if there are any outliers
summary(housing_df$Sale_Price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     698  460000  593000  660738  750000 4400000
```

```
# Missing value analysis
any(is.na(housing_df$Sale_Price))
```

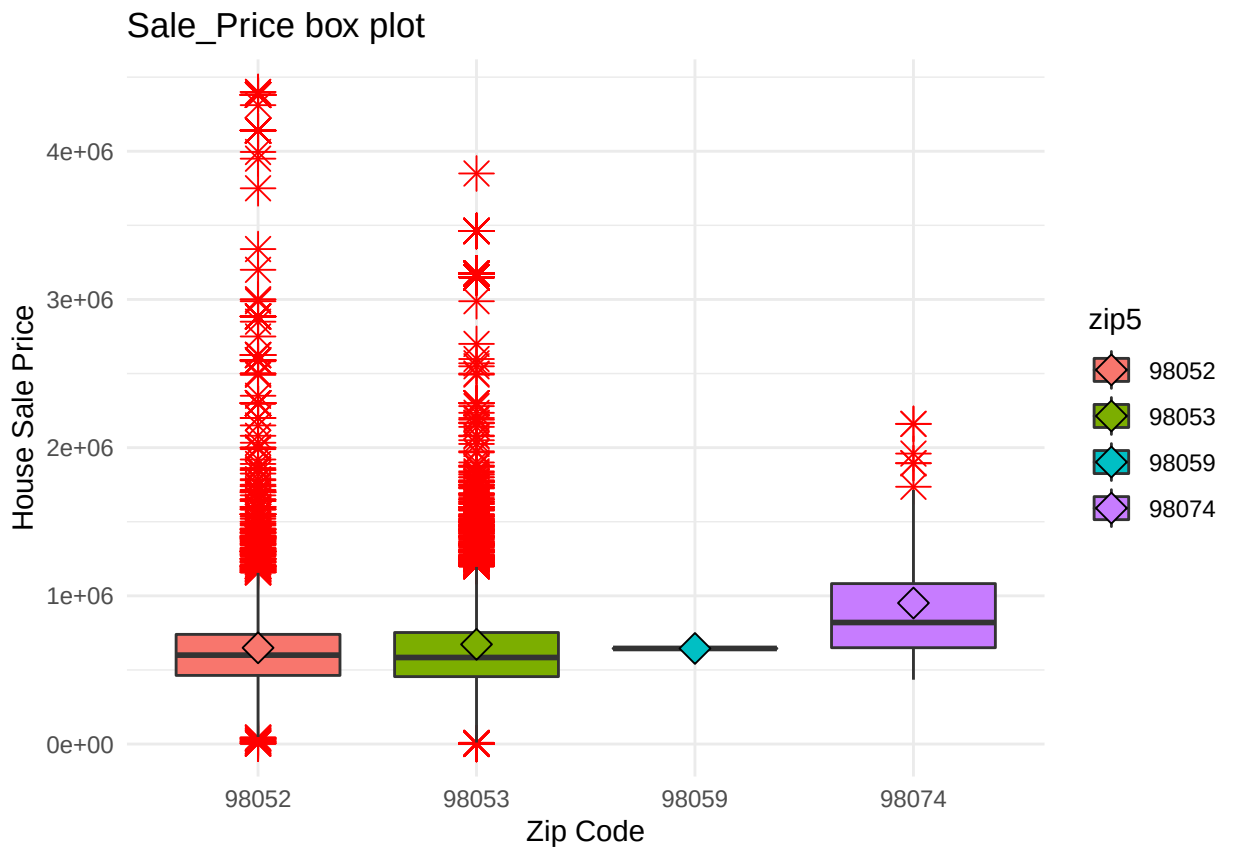
```
## [1] FALSE
```

```
summary(is.na(housing_df$Sale_Price))
```

```
##    Mode   FALSE
## logical   12865
```

```
# From the above result, it is clear that the dataset sale price contains NO Missing Values.
housing_df$zip5 = as.factor(housing_df$zip5)
ggplot(housing_df, aes(y=Sale_Price, x=zip5, fill=zip5))  +
    labs(x="Zip Code", y="House Sale Price", title="Sale_Price box plot") +
    geom_boxplot(outlier.colour="red", outlier.shape=8,
                 outlier.size=4) +
    stat_summary(fun.y=mean, geom="point", shape=23, size=4) +  theme_minimal()
```

```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```




```
# Box plot clearly shows some outliers specially in 98052 and 98053 zip codes
# with very low sale price and some out of range.

## F. Create at least 2 new variables
# Creating per square ft sale price
housing_df$sq_ft_lot_price <- housing_df$Sale_Price / housing_df$sq_ft_lot
housing_df$sq_ft_living_price <- housing_df$Sale_Price / housing_df$square_feet_total_living

# Creating total bath count variable
housing_df$total_bath_count <- housing_df$bath_full_count +
    housing_df$bath_half_count + housing_df$bath_3qtr_count
```