California Housing Data Analysis

Author: Nandini Ethirajulu

```
#Considering California Housing Data from KAGGLE
(https://www.kaggle.com/camnugent/california housing-prices) as source



#Reading the data and tranfering it to a binary incidence matrix

#install.packages("arules")
library("arules")

## Warning: package 'arules' was built under R version 4.3.2

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.3.3

##
## Attaching package: 'arules'

## The following objects are masked from 'package:base':
##
##     abbreviate, write

##Setting the directory for the source data
setwd("C:/Users/nandi/Documents/Personal/Academics/Projects/California
Housing Data Analysis/California Housing Data/archive")
housing_dats <- read.csv("housing.csv",header = TRUE)

colnames(housing_dats)

##  [1] "longitude"          "latitude"           "housing_median_age"
##  [4] "total_rooms"        "total_bedrooms"     "population"
##  [7] "households"         "median_income"      "median_house_value"
## [10] "ocean_proximity"

dim(housing_dats)

## [1] 20640    10

##20640 rows and   10 attributes

##preprocessing steps

##identifying na values
sum(is.na(housing_dats))
```

```
## [1] 207

##NA values derived from total_bedrooms variable in this dataset
total_bedrooms_check = housing_dats$total_bedrooms
sum(is.na(total_bedrooms_check))

## [1] 207

##identifying the mean value to use and fill in the missing datapoints
bedroom_median = median(housing_dats$total_bedrooms, na.rm=TRUE)
housing_dats$total_bedrooms[is.na(housing_dats$total_bedrooms)] =
bedroom_median

#Transforming data into binary dataset

#creating a new dataframe (binary_housing_data) for storing additional
binary data columns
binary_housing_data <- housing_dats
head(housing_dats)

##   longitude latitude housing_median_age total_rooms total_bedrooms
population
## 1   -122.23    37.88                 41         880            129
322
## 2   -122.22    37.86                 21        7099           1106
2401
## 3   -122.24    37.85                 52        1467            190
496
## 4   -122.25    37.85                 52        1274            235
558
## 5   -122.25    37.85                 52        1627            280
565
## 6   -122.25    37.85                 52         919            213
413
##   households median_income median_house_value ocean_proximity
## 1        126        8.3252             452600        NEAR BAY
## 2       1138        8.3014             358500        NEAR BAY
## 3        177        7.2574             352100        NEAR BAY
## 4        219        5.6431             341300        NEAR BAY
## 5        259        3.8462             342200        NEAR BAY
## 6        193        4.0368             269700        NEAR BAY

###head(housing_dats)

# Obtaining threshold values for numerical variables in this dataset
median_threshold <- median(binary_housing_data$housing_median_age)
rooms_threshold <- median(binary_housing_data$total_rooms)
bedrooms_threshold <- median(binary_housing_data$total_bedrooms)
population_threshold <- median(binary_housing_data$population)
income_threshold <- median(binary_housing_data$median_income)
value_threshold <- median(binary_housing_data$median_house_value)
```

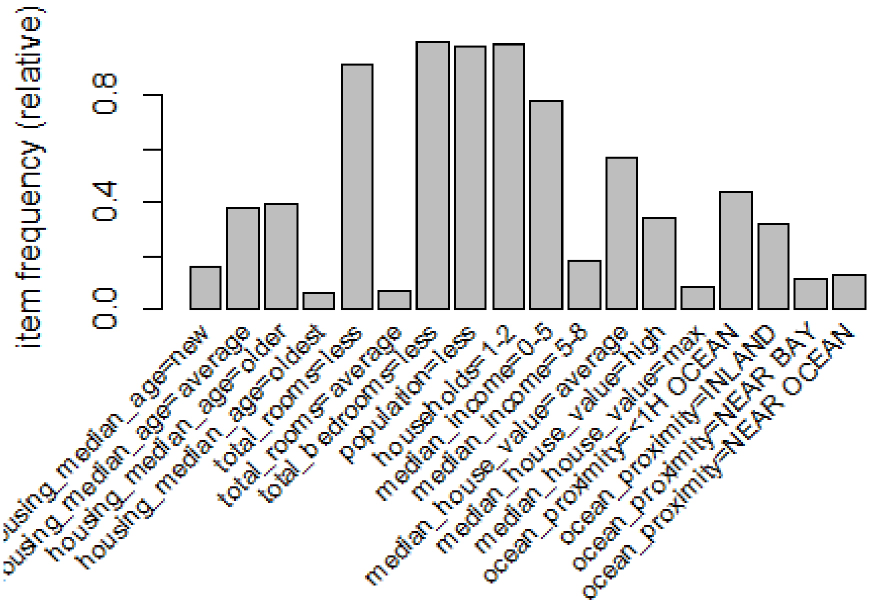
```
households_threshold <- median(housing_dats$households)

##identifying max values and using it for binning
max((binary_housing_data$total_bedrooms))

## [1] 6445

max((binary_housing_data$total_rooms))

## [1] 39320

max((binary_housing_data$population))

## [1] 35682

max((binary_housing_data$median_house_value))

## [1] 500001

max((binary_housing_data$total_bedrooms))

## [1] 6445

##Binning the variables into categories
binary_housing_data[["housing_median_age"]] <-
ordered(cut(binary_housing_data[["housing_median_age"]], c(0, 15, 30, 50,
70)), labels = c("new", "average", "older", "oldest"))


binary_housing_data[["total_rooms"]] <-
ordered(cut(binary_housing_data[["total_rooms"]], c(0, 5000, 10000, 27000,
50000)), labels = c("less", "average", "high", "max"))




binary_housing_data[["total_bedrooms"]] <- ordered(
  cut(binary_housing_data[["total_bedrooms"]], c(0, 3000, 5000, 10000)),
  labels = c("less", "average", "high")
)


binary_housing_data[["population"]] <-
ordered(cut(binary_housing_data[["population"]], c(0, 5000, 10000, 27000,
50000)), labels = c("less", "average", "high", "max"))



binary_housing_data[["median_income"]] <-
ordered(cut(binary_housing_data[["median_income"]], c(0, 5, 8, 12, 20)),
labels = c("0-5", "5-8", "8-12", "12-20"))
```

```
binary_housing_data[["median_house_value"]] <-
ordered(cut(binary_housing_data[["median_house_value"]], c(0, 50000, 200000,
400000, Inf)), labels = c("less", "average", "high", "max"))


binary_housing_data[["households"]] <-
ordered(cut(binary_housing_data[["households"]], c(0, 2000, 3000, 6000,
10000)), labels = c("1-2", "2-3", "3-4", "4-5"))

##Converting all variables into factors
binary_housing_data$housing_median_age <-
as.factor(binary_housing_data$housing_median_age)

binary_housing_data$total_rooms   <-
as.factor(binary_housing_data$total_rooms)
binary_housing_data$total_bedrooms   <-
as.factor(binary_housing_data$total_bedrooms)

binary_housing_data$population  <- as.factor(binary_housing_data$population)
binary_housing_data$median_income  <-
as.factor(binary_housing_data$median_income)
binary_housing_data$median_house_value  <-
as.factor(binary_housing_data$median_house_value)
binary_housing_data$households  <- as.factor(binary_housing_data$households)
binary_housing_data$ocean_proximity  <-
as.factor(binary_housing_data$ocean_proximity)

binary_housing_data <- binary_housing_data[, -
which(names(binary_housing_data) %in% c("longitude","latitude"))]
head(binary_housing_data)
```

```
##   housing_median_age total_rooms total_bedrooms population households
## 1              older        less           less       less        1-2
## 2            average     average           less       less        1-2
## 3             oldest        less           less       less        1-2
## 4             oldest        less           less       less        1-2
## 5             oldest        less           less       less        1-2
## 6             oldest        less           less       less        1-2
##   median_income median_house_value ocean_proximity
## 1          8-12                max        NEAR BAY
## 2          8-12               high        NEAR BAY
## 3           5-8               high        NEAR BAY
## 4           5-8               high        NEAR BAY
## 5           0-5               high        NEAR BAY
## 6           0-5               high        NEAR BAY
```

```
##Developing Binary incidence matrix
binary_incidence_matrix <- as(binary_housing_data, "transactions")
```

```
#Visualizing binary incidence matrix
itemFrequencyPlot(binary_incidence_matrix, support = 0.05, cex.names = 0.8)
```



```
#b. top three high lift rules
rule_params <- list(support = .005, confidence = .01, minlen = 2, maxlen = 6)
housing_arules <- apriori(binary_incidence_matrix, parameter = rule_params)
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##        0.01    0.1    1 none FALSE            TRUE       5   0.005      2
##  maxlen target  ext
##       6  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 103
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[32 item(s), 20640 transaction(s)] done [0.00s].
## sorting and recoding items ... [24 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 4 5 6
```

```
## Warning in apriori(binary_incidence_matrix, parameter = rule_params):
Mining
## stopped (maxlen reached). Only patterns up to a length of 6 returned!
```

```
##  done [0.00s].
## writing ... [16611 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
print("top three high lift rules of housing dataset are: ")
```

```
## [1] "top three high lift rules of housing dataset are: "
```

```
inspect(sort(housing_arules, by = "lift")[1:3,])
```

```
##     lhs                         rhs                    support confidence
coverage     lift count
## [1] {total_rooms=high,
##      population=average}  => {households=2-3}     0.005474806  0.6312849
0.008672481 84.06271   113
## [2] {total_rooms=high,
##      households=2-3}      => {population=average} 0.005474806  0.9262295
0.005910853 67.79212   113
## [3] {population=average,
##      households=2-3}      => {total_rooms=high}   0.005474806  0.8897638
0.006153101 66.29864   113
```

```
#c. top 4 rules according to confidence
print("top 4 rules according to confidence of housing dataset are: ")

## [1] "top 4 rules according to confidence of housing dataset are: "

inspect(sort(housing_arules, by = "confidence")[1:4,])

##     lhs                             rhs                    support    
confidence
## [1] {median_income=12-20}        => {population=less}     0.005474806 1
## [2] {median_income=12-20}        => {households=1-2}      0.005474806 1
## [3] {median_income=12-20}        => {total_bedrooms=less} 0.005474806 1
## [4] {housing_median_age=oldest} => {total_bedrooms=less} 0.064001938 1
##     coverage    lift     count
## [1] 0.005474806 1.014999  113
## [2] 0.005474806 1.009933  113
## [3] 0.005474806 1.003257  113
## [4] 0.064001938 1.003257 1321
```

```
#d. Recommendations for Purchasing an Average Priced Home Near the Ocean

##Analysis based on the association rules:

housing_near_ocean = subset(housing_arules, rhs %in% "ocean_proximity=NEAR 
OCEAN")

inspect(sort(housing_near_ocean, by = "lift")[1:5,])

##     lhs                               rhs                            
support confidence   coverage     lift count
## [1] {housing_median_age=average,                                            
##      median_house_value=max}      => {ocean_proximity=NEAR OCEAN} 
0.007218992  0.2738971 0.02635659 2.126876   149
## [2] {housing_median_age=average,                                            
##      total_bedrooms=less,                                                   
##      median_house_value=max}      => {ocean_proximity=NEAR OCEAN} 
0.007170543  0.2730627 0.02625969 2.120397   148
## [3] {housing_median_age=average,                                            
##      population=less,                                                       
##      median_house_value=max}      => {ocean_proximity=NEAR OCEAN}
```

```
0.007073643  0.2713755 0.02606589 2.107295   146
## [4] {housing_median_age=average,
##      total_bedrooms=less,
##      population=less,
##      median_house_value=max}      => {ocean_proximity=NEAR OCEAN}
0.007073643  0.2713755 0.02606589 2.107295   146
## [5] {housing_median_age=average,
##      population=less,
##      households=1-2,
##      median_house_value=max}      => {ocean_proximity=NEAR OCEAN}
0.007025194  0.2705224 0.02596899 2.100670   145

## the housing the person is looking for should be having average median age
(15-30), fewer bedrooms and the surrounding neighborhood are expected to be
less in population, with an average households of 1 to 2. And the median
house values are expected to be around more than 400000
```

```
#e.  Characteristics Associated with Low Population Areas

housing_less_population = subset(housing_arules, rhs %in% "population=less")
inspect(sort(housing_less_population, by = "lift")[1:5,])

##     lhs                              rhs                  support confidence
coverage     lift count
## [1] {median_income=12-20}    => {population=less} 0.005474806          1
0.005474806 1.014999   113
## [2] {median_income=12-20,
##      median_house_value=max} => {population=less} 0.005232558          1
0.005232558 1.014999   108
## [3] {total_rooms=less,
##      median_income=12-20}    => {population=less} 0.005038760          1
0.005038760 1.014999   104
## [4] {households=1-2,
##      median_income=12-20}    => {population=less} 0.005474806          1
0.005474806 1.014999   113
## [5] {total_bedrooms=less,
##      median_income=12-20}    => {population=less} 0.005474806          1
0.005474806 1.014999   113

###Low population areas associate with median house values more than 400000,
few bedrooms, very low households of 1-2, ver minimum total rooms and median
income of between 12-20
```