R_4_categoricals_stats

Mike Montes

2025-01-06

1. Load and call dataset

```
library(readr)
loan <- read_csv("C:/Users/mmsax/School_Portfolio/Coding_Skills/loan.csv")
```

```
## Rows: 10000 Columns: 11
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): term, grade, emp_length, home_ownership, verification_status, loan_...
## dbl (5): id, loan_amnt, int_rate, installment, annual_inc
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
head(loan, 2)
```

```
## # A tibble: 2 x 11
##        id loan_amnt term    int_rate installment grade emp_length home_ownership
##     <dbl>     <dbl> <chr>      <dbl>       <dbl> <chr> <chr>      <chr>
## 1 1077501      5000 36 mon~     10.6       163.   B     10+ years  RENT
## 2 1077430      2500 60 mon~     15.3        59.8 C     < 1 year   RENT
## # i 3 more variables: annual_inc <dbl>, verification_status <chr>,
## #   loan_status <chr>
```

2. Show continuous & categorical variables in the dataset.

I made this a little fancier to practice.

I could have switched all the characters to factors here, but I reserved not doing it yet. I realize that integers and logicals could be included in !numeric.

```
categoricals <- sapply(loan, class) != 'numeric'
cat_names <-  names(loan[categoricals])
print('The categorical variables are:')
```

```
## [1] "The categorical variables are:"
```

```
print(cat_names)
```

```
## [1] "term"                "grade"               "emp_length"
## [4] "home_ownership"      "verification_status" "loan_status"
```

```
numericals <- sapply(loan, class) == 'numeric'
num_names <- names(loan[numericals])
print('The numerical variables are:')
```

```
## [1] "The numerical variables are:"
```

```
print(num_names)
```

```
## [1] "id"          "loan_amnt"   "int_rate"    "installment" "annual_inc"
```

3. Calculate the minimum, maximum, mean, median, standard deviation and three quartiles (25th, 50th and 75th percentiles) of loan_amnt.

I practiced old and new commands.

I could have done the quartiles separately, but I wanted to put them together without using a loop so I had to learn 'collapse' so the 2nd and 3rd values wouldn't get cut off

The boring method is first but is always useful

```
(summary(loan$loan_amnt))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1000    6000   11200   12862   17500   35000
```

```
print(paste('The minimum loan_amount value is: $', (min(loan$loan_amnt))))
```

```
## [1] "The minimum loan_amount value is: $ 1000"
```

```
print(paste('The maximum loan_amount value is: $', (max(loan$loan_amnt))))
```

```
## [1] "The maximum loan_amount value is: $ 35000"
```

```
print(paste('The mean loan_amount value is: $', round(mean(loan$loan_amnt), 2)))
```

```
## [1] "The mean loan_amount value is: $ 12861.64"
```

```
print(paste('The median loan_amount value is: $', (median(loan$loan_amnt))))
```

```
## [1] "The median loan_amount value is: $ 11200"
```

```
print(paste('The standard deviation of the loan_amount values is: $', round(sd(loan$loan_amnt), 2)))
```

```
## [1] "The standard deviation of the loan_amount values is: $ 8491.81"
```

```
# I could have done these separately, but I wanted to put them together without using a
# loop so I had to learn 'collapse' so the 2nd and 3rd values wouldn't get cut off
print(paste('The 25th, 50th and 75th percentiles of the loan_amount values are: $',
            paste(quantile(loan$loan_amnt, probs = c(0.25, 0.5, 0.75)), collapse = ", ")))
```

```
## [1] "The 25th, 50th and 75th percentiles of the loan_amount values are: $ 6000, 11200, 17500"
```

4. Calculate the minimum, maximum, mean, median, standard deviation and three quartiles (25th, 50th and 75th percentiles) of int_rate.

```
(summary(loan$int_rate))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    5.42    8.90   12.42   12.43   15.27   24.11
```

```
# Practicing old and new commands:
print(paste('The minimum int_rate value is: $', (min(loan$int_rate))))
```

```
## [1] "The minimum int_rate value is: $ 5.42"
```

```
print(paste('The maximum int_rate value is: $', (max(loan$int_rate))))
```

```
## [1] "The maximum int_rate value is: $ 24.11"
```

```
print(paste('The mean int_rate value is: $', round(mean(loan$int_rate), 2)))
```

```
## [1] "The mean int_rate value is: $ 12.43"
```

```
print(paste('The median int_rate value is: $', (median(loan$int_rate))))
```

```
## [1] "The median int_rate value is: $ 12.42"
```

```
print(paste('The standard deviation of the int_rate values is: $', round(sd(loan$int_rate), 2)))
```

```
## [1] "The standard deviation of the int_rate values is: $ 4.24"
```

```
print(paste('The 25th, 50th and 75th percentiles of the int_rate values are: $',
            paste(quantile(loan$int_rate, probs = c(0.25, 0.5, 0.75)), collapse = ", ")))
```

```
## [1] "The 25th, 50th and 75th percentiles of the int_rate values are: $ 8.9, 12.42, 15.27"
```

5. Calculate the correlation coefficient of int_rate and installment and detemine if they have a strong relationship.

```r
print(paste('The correlation between int_rate and installment is:', round(cor(loan$int_rate, loan$instal
```

```
## [1] "The correlation between int_rate and installment is: 0.28198"
```

```r
print("This is a very low correlation value, so they do not have a strong relationship")
```

```
## [1] "This is a very low correlation value, so they do not have a strong relationship"
```

6. Frequency table and mode of term.

```r
is.factor(loan$term)
```

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

```r
class(loan$term)
```

```
## [1] "character"
```

```r
loan$term <-  as.factor(loan$term)
is.factor(loan$term)
```

```
## [1] TRUE
```

```r
levels(loan$term)
```

```
## [1] "36 months" "60 months"
```

```r
print('The frequency table for loan$term:')
```

```
## [1] "The frequency table for loan$term:"
```

```r
print(table(loan$term))
```

```
##
## 36 months 60 months
##      6649      3351
```

```r
print(paste('The mode of term is:', names(sort(table(loan$term), decreasing = TRUE))[1]))
```

```
## [1] "The mode of term is: 36 months"
```

7. The proportion table and mode of loan_status.

```r
is.factor(loan$loan_status)
```

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

```r
class(loan$loan_status)
```

```
## [1] "character"
```

```r
loan$loan_status <-  as.factor(loan$loan_status)
is.factor(loan$loan_status)
```

```
## [1] TRUE
```

```r
levels(loan$loan_status)
```

```
## [1] "Charged Off"        "Current"            "Default"
## [4] "Fully Paid"         "In Grace Period"    "Late (16-30 days)"
## [7] "Late (31-120 days)"
```

```r
print('The proportion table for loan$status:')
```

```
## [1] "The proportion table for loan$status:"
```

```r
print(proportions((table((loan$loan_status)))))
```

```
##
##        Charged Off            Current            Default         Fully Paid
##             0.1517             0.0956             0.0002             0.7487
##    In Grace Period  Late (16-30 days) Late (31-120 days)
##             0.0008             0.0006             0.0024
```

```r
  print(paste('The mode of loan_status is:', names(sort(table(loan$loan_status), decreasing = TRUE))[1]
```

```
## [1] "The mode of loan_status is: Fully Paid"
```

8. The cross table of term and loan_status and proportions by row and column respectively.

```r
xtabs(~term + loan_status, data = loan)
```

```
##            loan_status
## term        Charged Off Current Default Fully Paid In Grace Period
##   36 months         754       0       0       5895               0
##   60 months         763     956       2       1592               8
##            loan_status
## term        Late (16-30 days) Late (31-120 days)
##   36 months                 0                  0
##   60 months                 6                 24
```

```r
print('Proportion table by row')
```

```
## [1] "Proportion table by row"
```

```r
prop.table((xtabs(~term + loan_status, data = loan)), margin = 1)
```

```
##             loan_status
## term         Charged Off      Current      Default   Fully Paid In Grace Period
##   36 months 0.1134005114 0.0000000000 0.0000000000 0.8865994886    0.0000000000
##   60 months 0.2276932259 0.2852879737 0.0005968368 0.4750820651    0.0023873471
##             loan_status
## term        Late (16-30 days) Late (31-120 days)
##   36 months      0.0000000000       0.0000000000
##   60 months      0.0017905103       0.0071620412
```

```r
print('Proportion table by column')
```

```
## [1] "Proportion table by column"
```

```r
prop.table((xtabs(~term + loan_status, data = loan)), margin = 2)
```

```
##             loan_status
## term        Charged Off   Current   Default Fully Paid In Grace Period
##   36 months   0.4970336 0.0000000 0.0000000  0.7873648       0.0000000
##   60 months   0.5029664 1.0000000 1.0000000  0.2126352       1.0000000
##             loan_status
## term        Late (16-30 days) Late (31-120 days)
##   36 months         0.0000000          0.0000000
##   60 months         1.0000000          1.0000000
```

9. The summary all the variables using one command.

```r
summary(loan)
```

```
##        id            loan_amnt            term         int_rate
##  Min.   : 458165   Min.   : 1000   36 months:6649   Min.   : 5.42
##  1st Qu.: 878178   1st Qu.: 6000   60 months:3351   1st Qu.: 8.90
##  Median : 987925   Median :11200                    Median :12.42
##  Mean   : 963545   Mean   :12862                    Mean   :12.43
##  3rd Qu.:1033696   3rd Qu.:17500                    3rd Qu.:15.27
##  Max.   :1077501   Max.   :35000                    Max.   :24.11
##
##   installment         grade            emp_length        home_ownership
##  Min.   :  22.24   Length:10000       Length:10000       Length:10000
##  1st Qu.: 193.58   Class :character   Class :character   Class :character
##  Median : 322.25   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 363.82
##  3rd Qu.: 480.33
##  Max.   :1288.10
##
##    annual_inc      verification_status             loan_status
##  Min.   :   6000   Length:10000        Charged Off       :1517
##  1st Qu.:  42000   Class :character    Current           : 956
##  Median :  60000   Mode  :character    Default           :   2
##  Mean   :  70267                       Fully Paid        :7487
##  3rd Qu.:  84500                       In Grace Period   :   8
##  Max.   :1782000                       Late (16-30 days) :   6
##                                        Late (31-120 days):  24
```