Describing Numerical and Categorical Data

STA 032: Gateway to data science Lecture 10

Jingwei Xiong

April 24, 2023

Reminders

- HW 2 due April 26 12pm.
- HW 3 due May 3 12pm.
  - Please start the homework as soon as possible.
- **Midterm 1** (Open book, take home, approximate 2 hours, time limit 5 hours)
  - **Due April 29 midnight, cover lecture 1-12**
  - Preparing guides: Finish Homework 1-3, be familiar to the lecture slides.
  - You can copy any your own codes in homework 1-3 to finish the open book exam.
- The PDF version of slides can be found on Canvas.
  - You can use search on it to find function examples.

Recap

- Describing numerical distributions
  - Histograms
  - Measures of central tendency: mean, median, mode
  - Shape: skewness and modality
  - Spread: variance and standard deviation, range and interquartile range
  - Boxplots
  - Unusual observations
  - Density plot

Today

- Relationships between numerical variables
  - Scatterplot
  - Hex plot
  - Correlation coefficient
  - Line graph
- Describing categorical distributions
  - Bar plot
- Relationships between categorical data
  - Contingency tables

Data: Lending Club

- Lending Club is a platform that allows individuals to lend to other individuals

```
loans <- loans_full_schema %>%
  select(loan_amount, interest_rate, term, grade,
         state, annual_income, homeownership, debt_to_income,
         issue_month)
glimpse(loans)
```

```
Rows: 10,000
Columns: 9
$ loan_amount    <int> 28000, 5000, 2000, 21600, 23000, 5000, 24000, 20000, 2
$ interest_rate  <dbl> 14.07, 12.61, 17.09, 6.72, 14.07, 6.72, 13.59, 11.99,
$ term           <dbl> 60, 36, 36, 36, 36, 36, 60, 60, 36, 36, 60, 60, 36, 60
$ grade          <fct> C, C, D, A, C, A, C, B, C, A, C, B, C, B, D, D, D, F,
$ state          <fct> NJ, HI, WI, PA, CA, KY, MI, AZ, NV, IL, IL, FL, SC, CO
$ annual_income  <dbl> 90000, 40000, 40000, 30000, 35000, 34000, 35000, 11000
$ homeownership  <fct> MORTGAGE, RENT, RENT, RENT, RENT, OWN, MORTGAGE, MORTG
$ debt_to_income <dbl> 18.01, 5.04, 21.15, 10.16, 57.96, 6.46, 23.66, 16.19,
$ issue_month    <fct> Mar-2018, Feb-2018, Feb-2018, Jan-2018, Mar-2018, Jan-
```

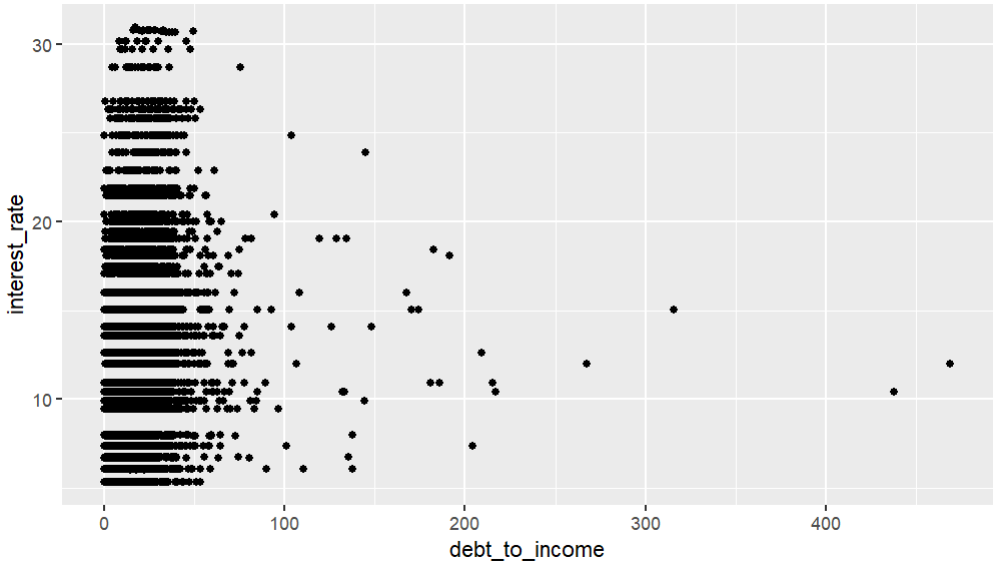
Relationships between numerical variables

- Paired or bivariate data
  - Scatterplot
  - Hexplot
  - Correlation
  - Line graph

Scatterplot

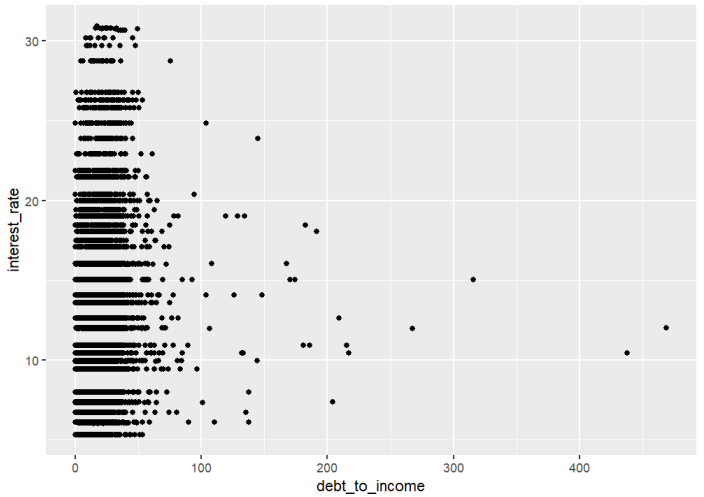
- We have seen examples of scatterplots
    - Our ggplot 2 example, our facet example.
    - Remember how to change the theme?
- Each point is a single observation with two characteristics, or variables, plotted on the x- and y-axis respectively

```
ggplot(loans, aes(x = debt_to_income, y = interest_rate)) +
geom_point()
```

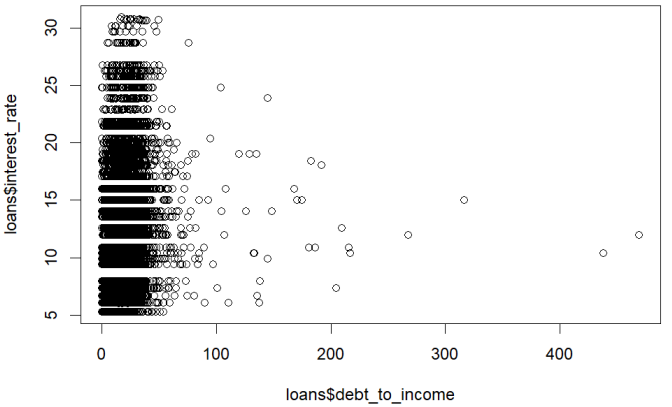


Scatterplot in base R

```
ggplot(loans, aes(x = debt_to_income, y = intere
  geom_point()
```



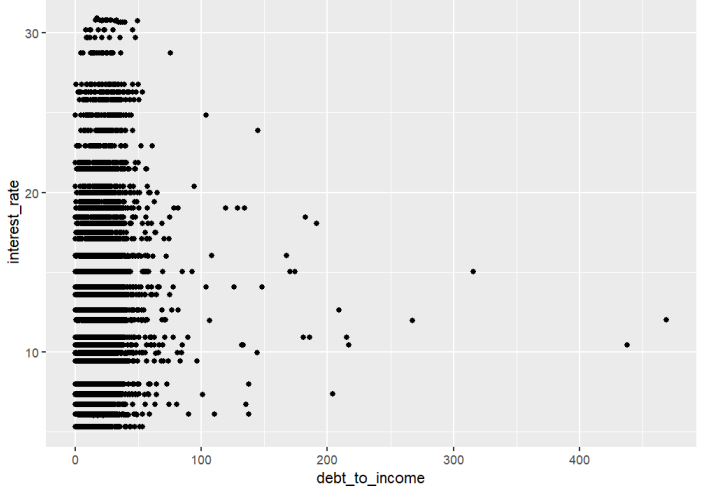

```
plot(loans$debt_to_income, loans$interest_rate)
```



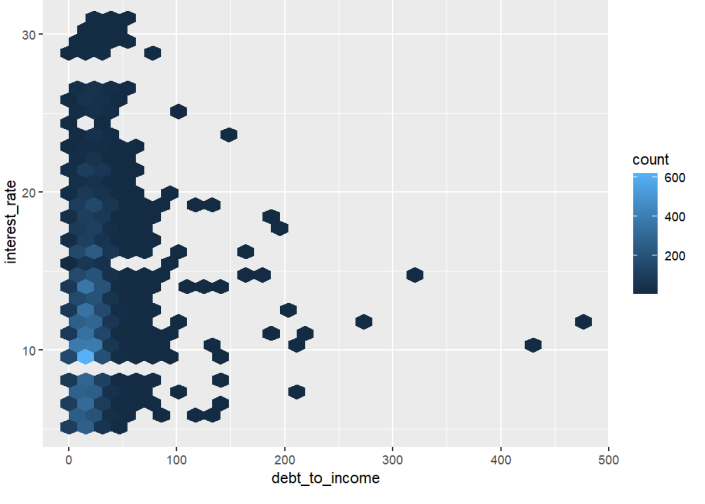
Overplotting and hex plots

- Overplotting is when points are plotted on top of each other
- Common in large data sets
- A few ways to deal with this include using `alpha`, or `jitter()`
- Alternatively, hex plots or hexbin plots

```
ggplot(loans, aes(x = debt_to_income,
                  y = interest_rate)) +
  geom_point()
```

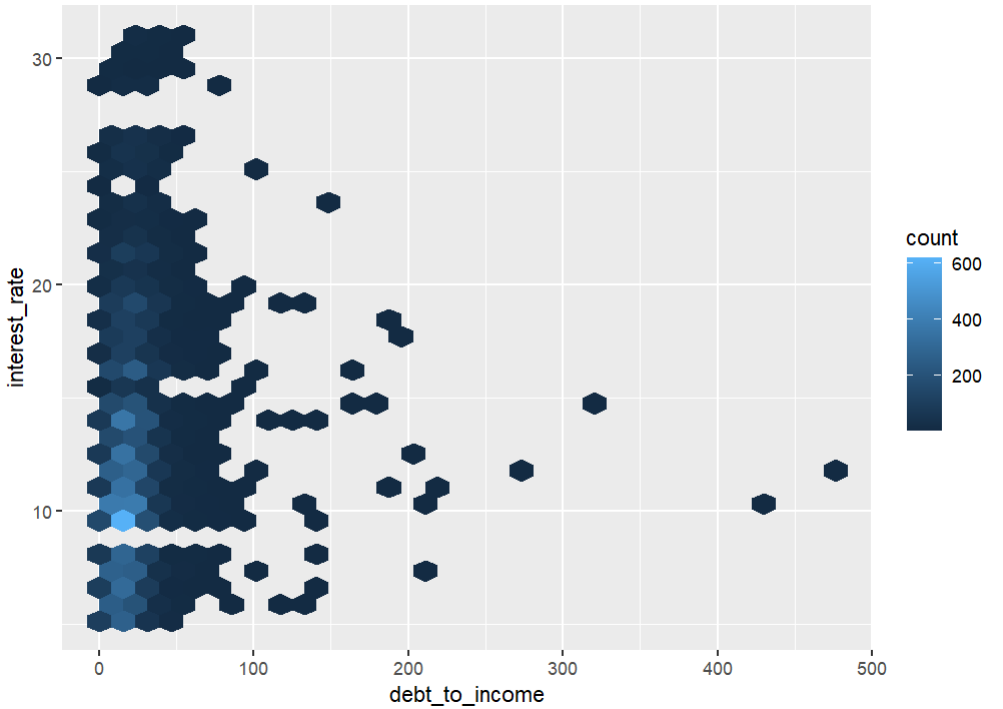



```
ggplot(loans, aes(x = debt_to_income,
                  y = interest_rate)) +
  geom_hex()
```



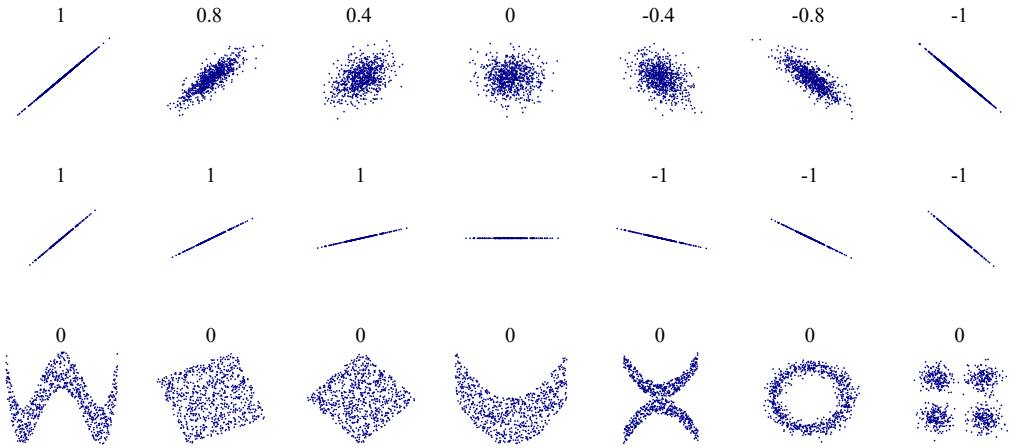
Hex plot

- Hex plots divide the graphing surface into hexagons
  - All points are grouped into their respective hexagonal regions
  - Color gradient indicates the number of observations (count) in each hexagonal area.



Correlation

- Correlation is the association between two variables
- (Pearson) Correlation coefficient is a measure of **linear** correlation between two sets of data
- Ranges from -1 to 1



Correlation

Recall:

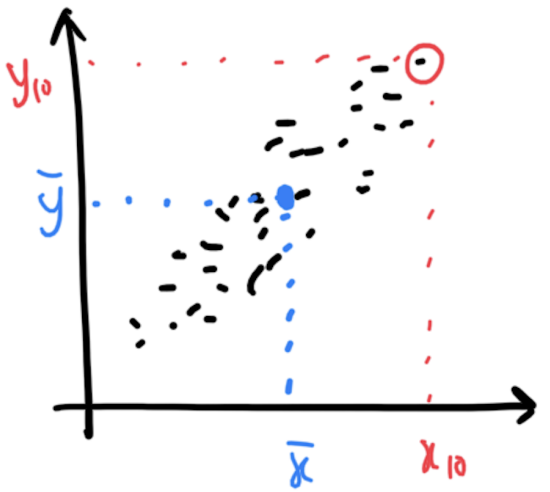
- Sample mean: $\bar{x} = \frac{x_1+x_2+...+x_n}{n} = \frac{\sum_{i=1}^{n} x_i}{n}$
- Sample variance: $s^2 = \frac{(x_1-\bar{x})^2+(x_2-\bar{x})^2+...+(x_n-\bar{x})^2}{n-1} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n-1}$
- Population mean: μ
- Population variance: σ^2
- When talking about a population parameter for a variable x, might use subscript x, e.g., μ_x, σ_x^2; similarly for a sample statistic, e.g., s_x^2

Correlation

- Sample correlation: $r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$
- Population correlation: ρ
- To calculate: `cor(vector1, vector2)` in R

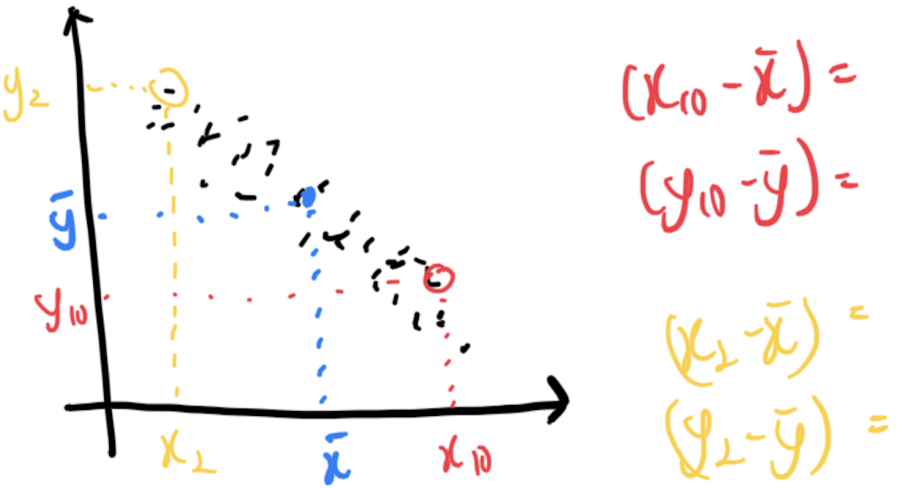
Correlation

- Sample correlation: $r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}}$



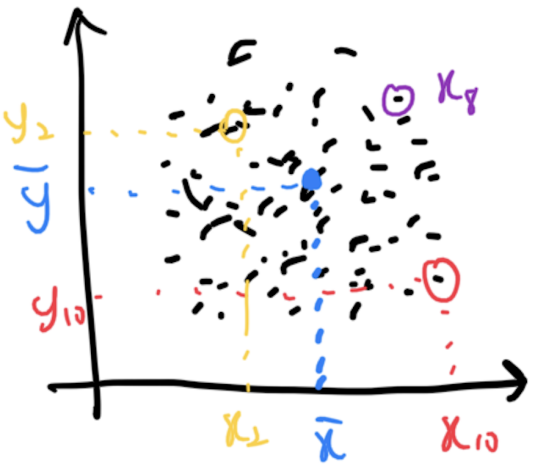
Correlation

- Sample correlation: $r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}}$



Correlation

- Sample correlation: $r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}}$
- Nominator "Variance" of X and Y.



$(x_{10} - \bar{x}) =$
$(y_{10} - \bar{y}) =$
$(x_1 - \bar{x}) =$
$(y_1 - \bar{y}) =$
$(x_8 - \bar{x}) =$
$(y_8 - \bar{y}) =$

Correlation

- Sample correlation: $r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}}$
- Nominator "Variance" of X and Y.
- What does the denominator look like?
  - Recall: Sample variance $s^2 = \frac{(x_1-\bar{x})^2+(x_2-\bar{x})^2+...+(x_n-\bar{x})^2}{n-1} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n-1}$
  - Denominator: $\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2} = \sqrt{(n-1)s_x^2(n-1)s_y^2} = (n-1)s_x s_y$
- Pearson correlation coefficient is scale and location-invariant
  - You can think of the denominator as a scaling factor
  - Subtract sample means, $\bar{x}$ and $\bar{y}$ does not change it

- Pearson correlation coefficient is scale and location-invariant
  - You can think of the denominator as a scaling factor
  - Subtract sample means, $\bar{x}$ and $\bar{y}$ does not change it

```
x = c(1,2,3,4,5)
y = c(2,2,1,4,5)
cor(x,y)
```

```
[1] 0.7698004
```

```
cor(x-mean(x), y-mean(y))
```

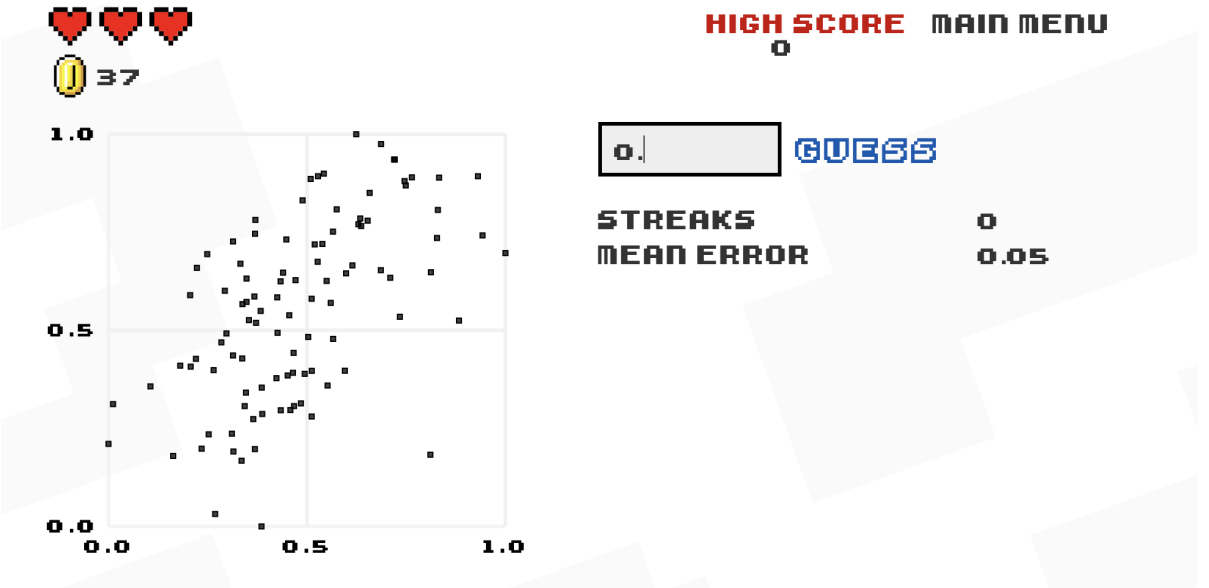
```
[1] 0.7698004
```

```
cor(x/2, y/2)
```

```
[1] 0.7698004
```

Guess the correlation

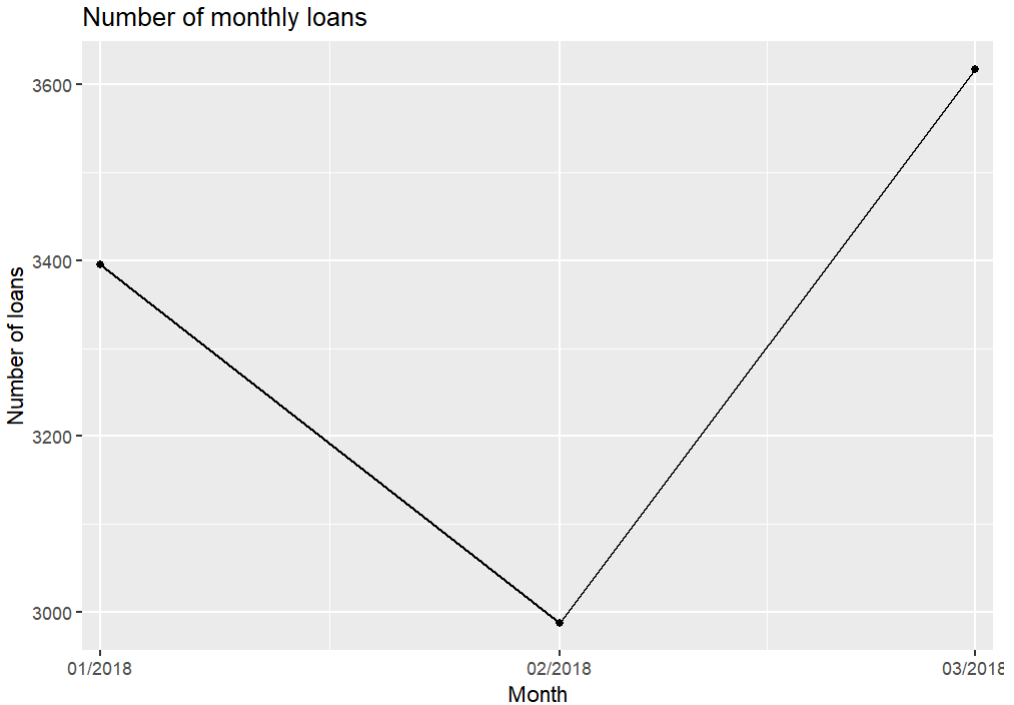
http://guessthecorrelation.com/



Line graphs: Observation against time

Line graphs are most commonly used for data over time, *time series data*

Plot | Code



Line graphs: Observation against time

Line graphs are most commonly used for data over time, *time series data*

Plot | Code

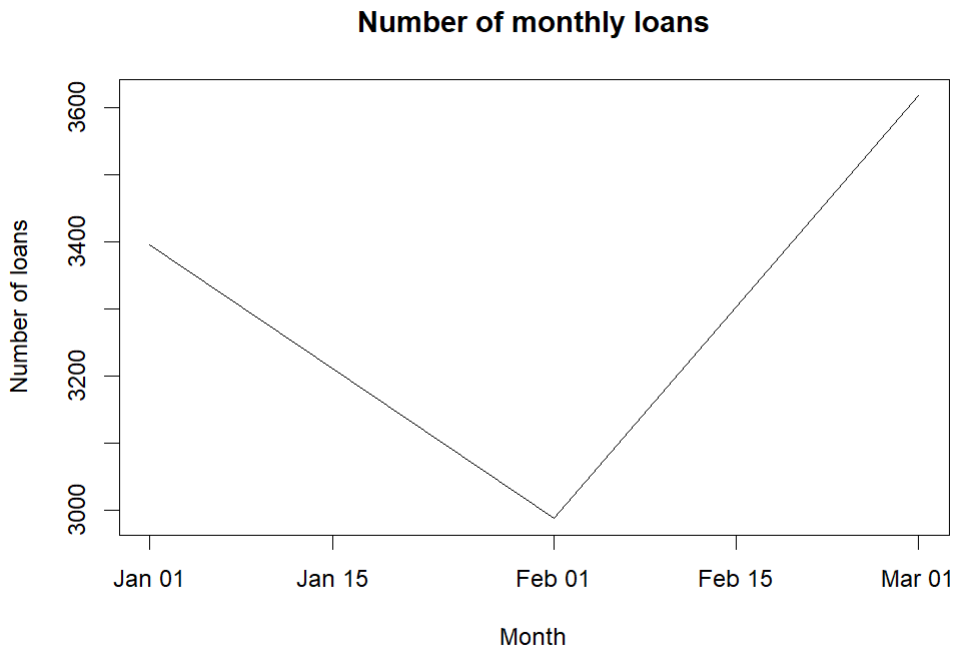
```
loans %>%
  group_by(issue_month) %>%
  summarize(count = n()) %>%
  ggplot(aes(x = as.Date(paste0("01-", issue_month), format = "%d-%b-%Y"), y = count)) +
  geom_point() +
  geom_line() +
  scale_x_date(labels = scales::date_format(format = "%m/%Y"),
               breaks = scales::date_breaks(width = "1 month"),
               expand = c(.02, .02)) +
  labs(title = "Number of monthly loans",
    y = "Number of loans",
    x = "Month")
```

Line graphs

- Be careful of `geom_path()` vs. `geom_line()`:
  - `geom_path()` connects the observations in the order in which they appear in the data
  - `geom_line()` connects them in order of the variable on the x axis.
- In base R: `plot(x, y, type = "l")`. Also see `lines()`

Line graphs in base R

Plot | Code



Line graphs in base R

Plot | Code

```
tmpDF <- loans %>%
  group_by(issue_month) %>%
  summarize(count = n()) %>%
  mutate(issue_month = as.Date(paste0("01-", issue_month), format = "%d-%b-%Y")) %>%
  arrange(issue_month)
plot(tmpDF$issue_month, tmpDF$count, type = "l",
     main = "Number of monthly loans",
     xlab = "Month",
     ylab = "Number of loans")
```

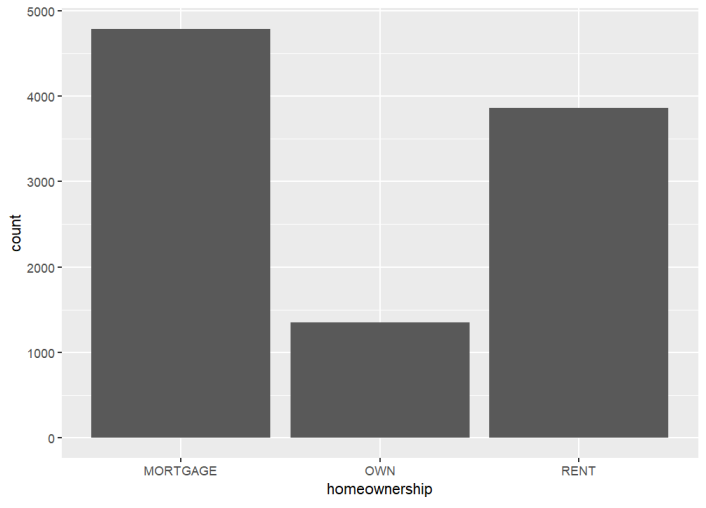
Describing categorical data

- Describing categorical distributions
  - Bar plot
- Relationships between categorical data
  - Contingency tables
  - Stacked bar plot (Next lecture)

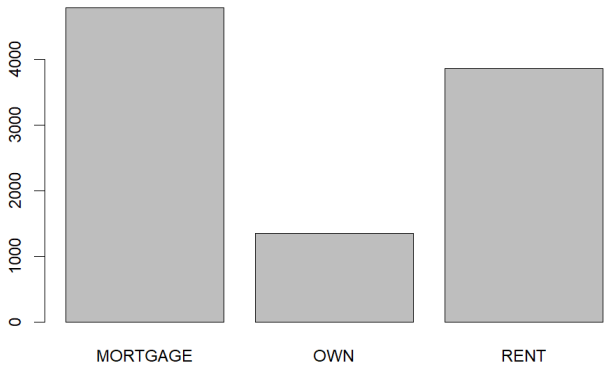
Bar plot: category counts

A bar plot is common way to display a single categorical variable.

```
ggplot(loans, aes(x = homeownership)) +
  geom_bar()
```

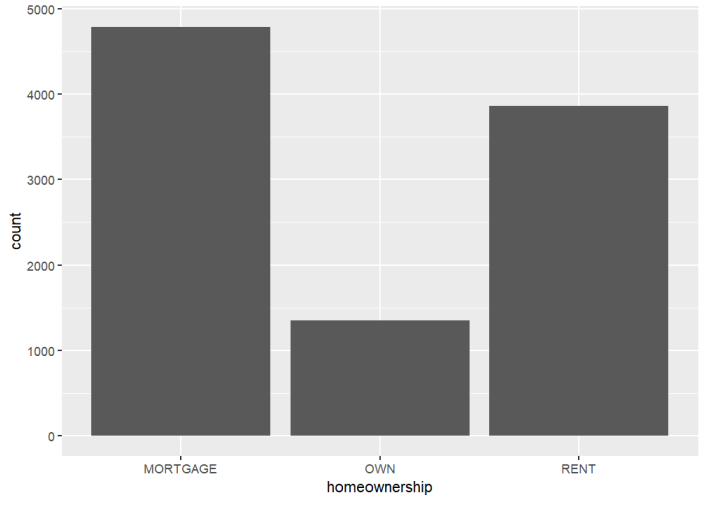



```
barplot(table(loans$homeownership)
        [table(loans$homeownership) > 0])
```

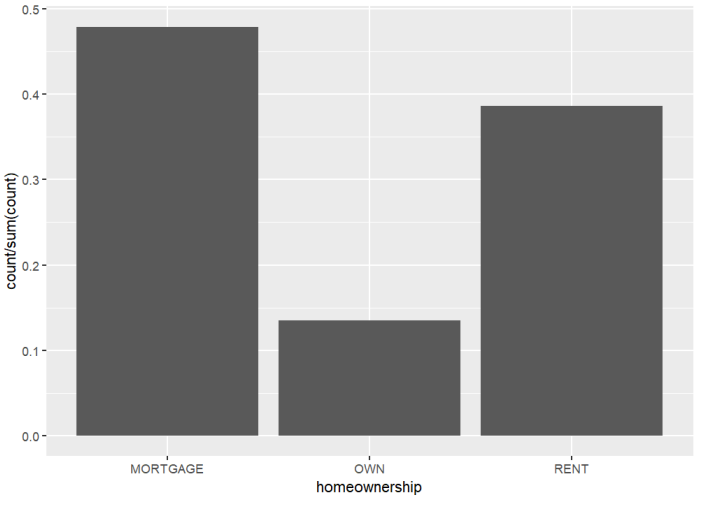


Bar plot with proportions

```
ggplot(loans, aes(x = homeownership)) +
  geom_bar()
```



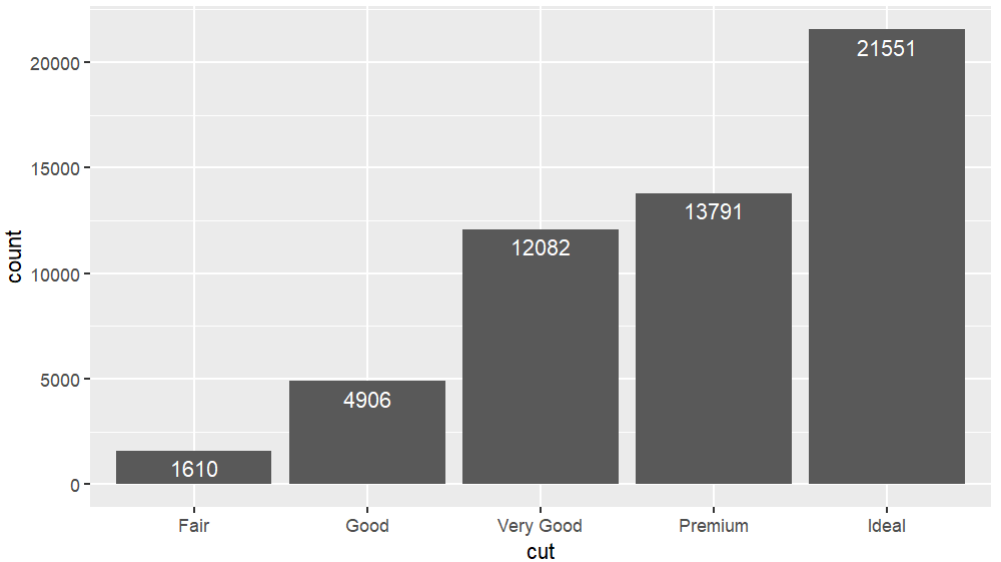

```
ggplot(loans, aes(x = homeownership)) +
  geom_bar(aes(y = ..count../sum(..count..)))
```



`..count..` is an internal variable that ggplot creates to store the count values (https://stackoverflow.com/questions/36604127/creating-a-bar-plot-with-proportions-on-ggplot)

Adding the count numbers to the barplot

```
# Below the top
ggplot(data = diamonds, mapping = aes(x = cut)) +
  geom_bar() +
  geom_text(aes(label = ..count..), stat = "count", vjust = 1.5, colo
```



Changing the vjust to a negative value will make the number on the top of the bars.

GGplot bar plot from 2 types of datasets

Overview | Column is Category | Column is count

There are 2 types of data to construct bar plot.

- Each row is 1 observation, with a column indicating category.
- Here we use the case 1 code.
- Each row is 1 category, with a row indicating the counts/totals
- Here we have to use the case 2 code, `ggplot(T1, aes(x = homeownership, y = n)) + geom_bar(stat = "identity")`

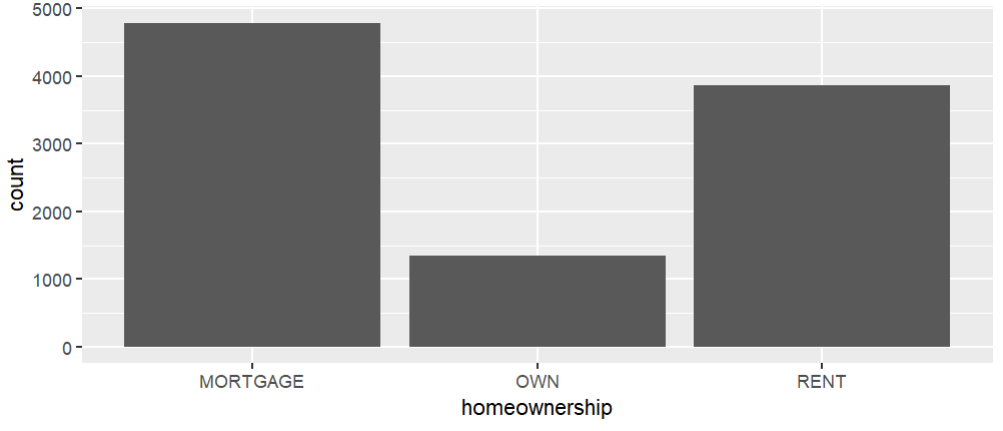
GGplot bar plot from 2 types of datasets

Overview | Column is Category | Column is count

```
head(loans)
```

```
# A tibble: 6 × 9
  loan_amount interest_rate  term grade state annual_i…¹ homeo…² debt_…³ issue…⁴
        <int>         <dbl> <dbl> <fct> <fct>      <dbl> <fct>     <dbl> <fct>
1       28000          14.1    60 C     NJ         90000 MORTGA…   18.0  Mar-20…
2        5000          12.6    36 C     HI         40000 RENT       5.04 Feb-20…
3        2000          17.1    36 D     WI         40000 RENT      21.2  Feb-20…
4       21600           6.72   36 A     PA         30000 RENT      10.2  Jan-20…
5       23000          14.1    36 C     CA         35000 RENT      58.0  Mar-20…
6        5000           6.72   36 A     KY         34000 OWN        6.46 Jan-20…
# … with abbreviated variable names ¹annual_income, ²homeownership,
#   ³debt_to_income, ⁴issue_month
```

```
ggplot(loans, aes(x = homeownership)) +
  geom_bar()
```



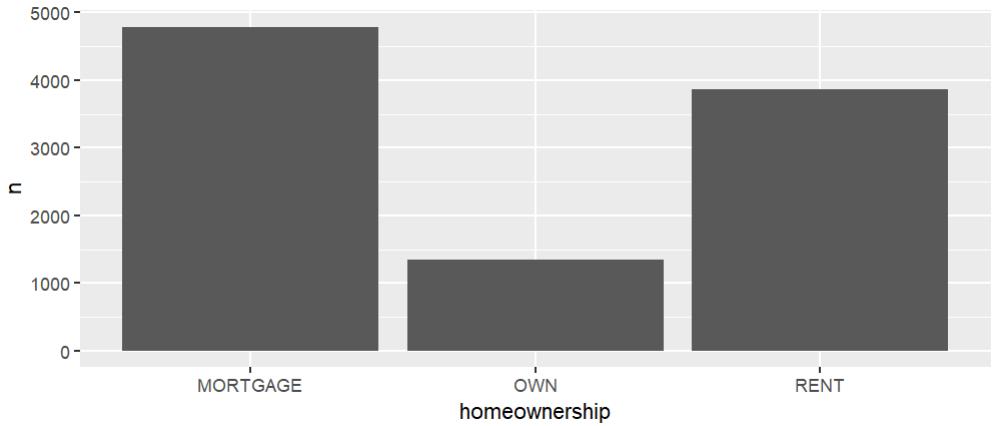
GGplot bar plot from 2 types of datasets

Overview | Column is Category | Column is count

```
T1 = loans %>% count(homeownership)
T1
```

```
# A tibble: 3 × 2
  homeownership     n
  <fct>         <int>
1 MORTGAGE       4789
2 OWN            1353
3 RENT           3858
```

```
ggplot(T1, aes(x = homeownership, y = n)) +
  geom_bar(stat = "identity")
```



Contingency tables

A contingency table summarizes data for two categorical variables

```
xtabs(~ homeownership + grade, data = loans_full_schema)
```

```
              grade
homeownership          A    B    C    D    E    F    G
                   0    0    0    0    0    0    0    0
     ANY           0    0    0    0    0    0    0    0
     MORTGAGE      0 1285 1499 1234  587  148   32    4
     OWN           0  347  414  335  211   38    5    3
     RENT          0  827 1124 1084  648  149   21    5
```

Each value in the table represents the number of times a particular combination of variable outcomes occurred, in other words the **frequency distribution** of the variables

Contingency tables

```
xtabs(~ homeownership + grade, data = loans_full_schema)
```

```
              grade
homeownership     A    B    C    D    E    F    G
            0     0    0    0    0    0    0    0
     ANY    0     0    0    0    0    0    0    0
     MORTGAGE    0 1285 1499 1234  587  148   32    4
     OWN    0  347  414  335  211   38    5    3
     RENT   0  827 1124 1084  648  149   21    5
```

- An additional row for column totals is often included
- Similarly, an additional column for row totals
- How do we code this in R?

Contingency tables with row and column totals

```
outTable <- xtabs(~ homeownership + grade, data = loans_full_schema)
outTableTotals <- outTable %>%
  cbind(rowTotal = rowSums(outTable))
outTableTotals <- outTableTotals %>%
  rbind(columnTotal = colSums(outTableTotals))
outTableTotals
```

```
                A    B    C    D   E  F  G rowTotal
            0   0    0    0    0   0  0  0        0
ANY         0   0    0    0    0   0  0  0        0
MORTGAGE    0 1285 1499 1234  587 148 32  4     4789
OWN         0  347  414  335  211  38  5  3     1353
RENT        0  827 1124 1084  648 149 21  5     3858
columnTotal 0 2459 3037 2653 1446 335 58 12    10000
```

Another way

```
outTable <- xtabs(~ homeownership + grade, data = loans_full_schema)
addmargins(outTable)
```

```
              grade
homeownership     A     B     C     D     E     F     G   Sum
            0     0     0     0     0     0     0     0     0
  ANY       0     0     0     0     0     0     0     0     0
  MORTGAGE  0  1285  1499  1234   587   148    32     4  4789
  OWN       0   347   414   335   211    38     5     3  1353
  RENT      0   827  1124  1084   648   149    21     5  3858
  Sum       0  2459  3037  2653  1446   335    58    12 10000
```

Summary

- Relationships between numerical variables
  - Scatterplot
  - Hex plot
  - Correlation coefficient
  - Line graph
- Describing categorical distributions
  - Bar plot
- Relationships between categorical data
  - Contingency tables

Readings

- Chapter 9 Visualizing data distributions
- R for Data Science Chapter 7