

Lecture 1: What is data science?

Beidi Qiang

SIUE

What is data science?

Data science as a blend of intellectual traditions from statistics, computer science and expert knowledge in domains (such as business, health care and etc.) to uncover actionable insights hidden in data.

- ▶ **Data collection:** collect raw data from all relevant sources
- ▶ **Data management:** set standards around data storage and structure, data cleaning, transforming and combining
- ▶ **Data analysis:** descriptive analysis, predictive analytics, machine learning
- ▶ **Communicate:** insights are presented with data visualizations for business analysts and other decision-makers to understand.

- ▶ Traditional statistical technique was originally developed in an environment where data were difficult or expensive to collect. So statisticians focused on creating methods that would maximize the strength of inference given the least amount of data.
- ▶ Today, datasets that are so large they can be processed only by machine. The increasing complexity of modern data means that each data analysis project needs to be custom-built with computer instructions and “code”.

A data scientist must be able to:

- ▶ Use techniques (such as SQL) for preparing and extracting data from databases. (CMIS 563)
- ▶ Extract insights from big data using predictive analytics, machine learning models, natural language processing, and deep learning.(STAT561 and 562)
- ▶ Write programs that automate data processing and calculations.
- ▶ Communication and teamwork. Collaborate with other team members, such as data and business analysts and convey the meaning of insights from data to decision makers.

Most popular programming languages to conduct data analysis:

- ▶ **R**: An open source programming language and environment for developing statistical computing and graphics.
- ▶ **Python**: A dynamic and flexible programming language that includes numerous libraries, such as NumPy, Pandas, for analyzing data.

Other tools:

- ▶ **Apache Spark, Hadoop, SQL and NoSQL**: manages the storage and processing of big data
- ▶ **PyTorch, TensorFlow**: building machine learning models
- ▶ **GitHub**: code sharing

CREATING CREDIT REPORTS

TransUnion is a credit reporting agency known for providing credit reports, fraud monitoring services and financial loans. The company's data science team is responsible for creating predictive models based on data reporting from auto dealers to retailers to mortgage companies. The company uses data science to extract insights from both an individual's credit data and public record data. These insights are used by financial institutions and lenders to make informed decisions about extending credit offers and loan opportunities.

THE EVOLUTION OF BASEBALL ANALYTICS (MDSR, section 1.2)

The use of statistics in baseball has a long and storied history. In 1947, Allan Roth, the first baseball's statistical analyst, made insights that the Dodgers used with his understanding of the game and his statistical analysis of baseball data. At the time, the tool of choice was often a spreadsheet. Over the next decade, the size of the data expanded so rapidly that a spreadsheet was no longer a viable mechanism for analyzing the baseball data. Today, many professional sports teams have research and development groups headed by people with training in machine learning.

Examples of Data Science Uses

PREDICTING RECIDIVISM WITHIN INCARCERATED POPULATIONS

Widely used by the American judicial system and law enforcement, Equivant's Northpointe software suite attempts to gauge an incarcerated person's risk of reoffending. Its algorithms predict that risk based on a questionnaire that covers the person's employment status, education level and more. The predictions were 71 percent accurate.

PREDICTING CONSUMERS' INTERESTS

Instagram uses data science to target its sponsored posts. The company's data scientists pull data from Instagram as well as its owner, Meta, which has exhaustive web-tracking infrastructure and detailed information on many users, including age and education. From there, the team crafts algorithms that convert users' likes and comments, their usage of other apps and their web history into predictions about the products they might buy.

SUGGESTING FRIENDS ON FACEBOOK

Meta's Facebook platform, of course, uses data science in various ways, but one of its buzzier data-driven features is the "People You May Know" sidebar, which appears on the social network's home screen. Often creepily prescient, it's based on a user's friend list, the people they've been tagged with in photos and where they've worked and gone to school. It's also based on "really good math," according to the Washington Post — specifically, a type of data science known as network science, which essentially forecasts the growth of a user's social network based on the growth of similar users' networks.

Lecture 2: Data Basics

Beidi Qiang

SIUE

Effective organization and description of data is a first step in most analyses. This lecture introduces the data matrix (dataset, data frame) for organizing data as well as some terminology.

Example of a Data Matrix

	loan_amount	interest_rate	term	grade	state	total_income	homeownership
1	7500	7.34	36	A	MD	70000	rent
2	25000	9.43	60	B	OH	254000	mortgage
3	14500	6.08	36	A	MO	80000	mortgage
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
50	3000	7.96	36	A	CA	34000	rent

Figure 1.3: Four rows from the `loan50` data matrix.

variable	description
<code>loan_amount</code>	Amount of the loan received, in US dollars.
<code>interest_rate</code>	Interest rate on the loan, in an annual percentage.
<code>term</code>	The length of the loan, which is always set as a whole number of months.
<code>grade</code>	Loan grade, which takes values A through G and represents the quality of the loan and its likelihood of being repaid.
<code>state</code>	US state where the borrower resides.
<code>total_income</code>	Borrower's total income, including any second income, in US dollars.
<code>homeownership</code>	Indicates whether the person owns, owns but has a mortgage, or rents.

- ▶ Observational units (observations or cases) are the rows of a data matrix.
- ▶ Variables are characteristics of the observational units, which are presented in the columns.

For example: For example, the 1st row represents a loan of \$7,500 with an interest rate of 7.34%, where the borrower is based in Maryland (MD) and has an income of \$70,000.

Types of variables

- ▶ Numerical variable: takes a wide range of numerical values, and it is sensible to add, subtract, or take averages with those values.
examples: loan amount, interest rate, term, total income
- ▶ Categorical variable: takes values of categories. The possible values are also called the variable's levels.
examples: grade, state, homeownership

Numerical variable

- ▶ Discrete: it can only take numerical values with jumps, usually integers.

examples: number of siblings, number of students, population



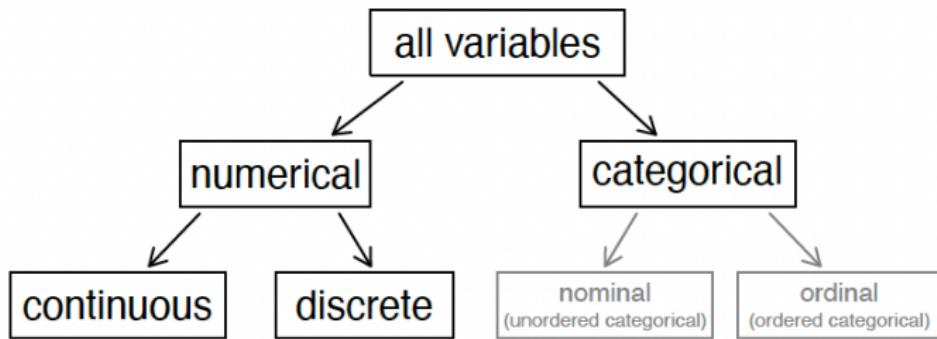
- ▶ continuous: it can take any value in an interval.

examples: interest rate, income

Categorical variable

- ▶ Ordinal: a categorical variable but the levels have a natural ordering.
examples: loan grade 
- ▶ Nominal: a categorical variable whose levels are unordered.
examples: homeownership, state 

Type of Variables



Relationships between Variables

In many analyses, researchers look for a relationship between two or more variables.

For example: Does an increase in county population tend to correspond to counties with higher or lower median household incomes?

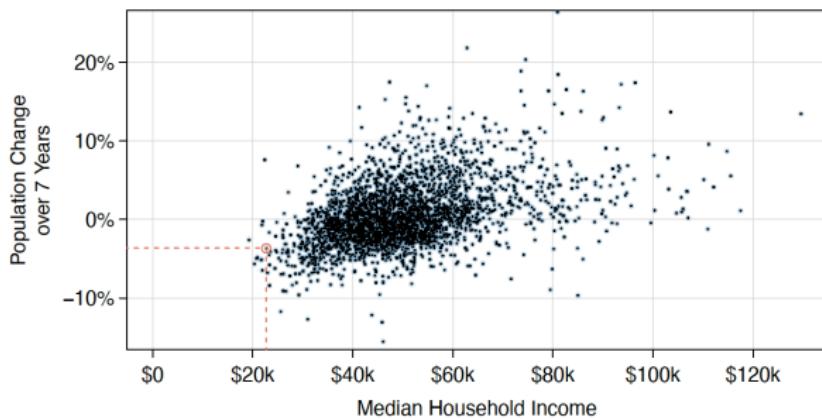


Figure 1.9: A scatterplot showing `pop_change` against `median_hh_income`. Owsley County of Kentucky, is highlighted, which lost 3.63% of its population from 2010 to 2017 and had median household income of \$22,736.

Explanatory v.s Response variables

When we ask questions about the relationship between two variables, we want to determine if one variable might causally affect another. We label the 1st variable the explanatory variable and the 2nd the response variable.

explanatory  affect response



If there is an increase in the median household income in a county, does this drive an increase in its population?



Here median household income is the explanatory variable and the population change is the response variable in the hypothesized relationship.

Type of Study (data collection)

Observational Study: Data are collected in a way that does not directly interfere with how the data arise. Observational studies can provide evidence of a naturally occurring association between variables, but they usually cannot show a causal connection.

For examples, collect information via surveys, review medical or company records.

Experiment: Researcher intentionally applies some treatments to individuals (subjects) and then measures a response variable. Experiments give information about cause and effect.

Observational or Experiment?

- ▶ Researchers measure birth weights of infants and record whether the mothers consumed alcohol during the pregnancy, to study whether drinking alcohol affects infant birth weight.
- ▶ Researchers gave some lab rats a trial vaccine and gave others a placebo. After a period of time, he then recorded whether the rats in the study contracted a certain disease.

STAT560-L3

Install R and Rstudio

R is an open source free software environment for statistical computing and graphics. You may download and install R here: <https://www.r-project.org/>

Rstudio is an open source free IDE (intergrated development environment) for R. You may download and install R studio here: <https://posit.co/> Rstudio is optional. You don't need R studio to use R, but Rstudio keeps R more organized.

Writing Scripts in R

Open file->New document in R or file->New file -> R script in Rstudio

Assigning a Value to an Object in R:

```
x=5  
print(x)  
  
## [1] 5  
y<-11  
print(y)  
  
## [1] 11  
y<-"Mike"  
y  
  
## [1] "Mike"  
x=8  
x  
  
## [1] 8  
ls()  
  
## [1] "x" "y"  
rm(y)  
ls()  
  
## [1] "x"
```

Arithmetic Operations in R:

```
5+10  
  
## [1] 15  
x=5;y=10  
x*y
```

```
## [1] 50
z=x*y
z

## [1] 50
x^2

## [1] 25
sqrt(x)

## [1] 2.236068
x/y

## [1] 0.5
log(x)

## [1] 1.609438
log2(x)

## [1] 2.321928
abs(-16)

## [1] 16
round(log(x),digits=2)

## [1] 1.61
```

Writing Comments in R:

```
#the following code is for...
#print(x)
```

Creating a vector in R:

```
x.vec=c(1,3,5,7,9)
x

## [1] 5
gender=c("male","female")
gender

## [1] "male"    "female"
y=2:7
y

## [1] 2 3 4 5 6 7
seq(from=2, to=7, by=0.25)

## [1] 2.00 2.25 2.50 2.75 3.00 3.25 3.50 3.75 4.00 4.25 4.50 4.75 5.00 5.25 5.50
## [16] 5.75 6.00 6.25 6.50 6.75 7.00
```

```

rep(1, times=10)

## [1] 1 1 1 1 1 1 1 1 1 1

rep(1:3,times=5)

## [1] 1 2 3 1 2 3 1 2 3 1 2 3 1 2 3

rep(1:3,each=2,times=3)

## [1] 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3

```

Vector operation in R:

```

x=c(1,3,5,7,9)
y=1:5
x*10

## [1] 10 30 50 70 90

x+y

## [1] 2 5 8 11 14

x*y

## [1] 1 6 15 28 45

x[2]

## [1] 3

y[-3]

## [1] 1 2 4 5

x[2:4]

## [1] 3 5 7

x[c(1,3)]

## [1] 1 5

x[-c(1,3)]

## [1] 3 7 9

x[x<6]

## [1] 1 3 5

```

Matrix in R:

```

m=matrix(1:9,nrow=3,byrow=T)
m

##      [,1] [,2] [,3]
## [1,]     1     2     3
## [2,]     4     5     6
## [3,]     7     8     9

```

```

matrix(1:9,nrow=3,byrow=FALSE)      
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
m[1,2]

## [1] 2
m[2,]

## [1] 4 5 6
m[,1]

## [1] 1 4 7
m[c(1,3),2]

## [1] 2 8
m+2

##      [,1] [,2] [,3]
## [1,]    3    4    5
## [2,]    6    7    8
## [3,]    9   10   11
m*10

##      [,1] [,2] [,3]
## [1,]   10   20   30
## [2,]   40   50   60
## [3,]   70   80   90

```

Logical statements (True/False) in R:

```

a=1>3
a

## [1] FALSE
x=c(1,3,5,7,9)      
x<7

## [1] TRUE  TRUE  TRUE FALSE FALSE
x==9

## [1] FALSE FALSE FALSE FALSE  TRUE
x<7&x>3

## [1] FALSE FALSE  TRUE FALSE FALSE
x<=7&x>=3

## [1] FALSE  TRUE  TRUE  TRUE FALSE
x<3|x>7

## [1]  TRUE FALSE FALSE FALSE  TRUE

```

```
as.numeric(x<3|x>7)
```

```
## [1] 1 0 0 0 1
```

Install packages in R

```
install.packages("MASS")
```

```
rownames(installed.packages())
```

```
## [1] "abind"           "aod"             "askpass"
## [4] "automap"         "backports"       "base"
## [7] "base64enc"       "BH"              "bit"
## [10] "bit64"          "blob"            "boot"
## [13] "brio"            "broom"           "bslib"
## [16] "cachem"          "callr"           "car"
## [19] "carData"         "caret"           "cellranger"
## [22] "class"           "cli"              "clipr"
## [25] "cluster"         "codetools"       "colorspace"
## [28] "colourpicker"    "commonmark"     "compiler"
## [31] "conflicted"      "corrplot"        "cowplot"
## [34] "cpp11"            "crayon"          "crosstalk"
## [37] "curl"             "data.table"     "datasets"
## [40] "DBI"              "dbplyr"          "dendextend"
## [43] "desc"             "diffobj"         "digest"
## [46] "doParallel"       "dplyr"           "DT"
## [49] "dtplyr"           "e1071"           "ellipse"
## [52] "ellipsis"         "evaluate"        "factoextra"
## [55] "FactoInvestigate" "FactoMineR"     "Factoshiny"
## [58] "fansi"            "farver"          "fastmap"
## [61] "flashClust"      "FNN"              "fontawesome"
## [64] "forcats"          "foreach"         "foreign"
## [67] "fs"               "future"          "future.apply"
## [70] "gargle"           "generics"        "ggplot2"
## [73] "ggpubr"           "ggrepel"         "ggsci"
## [76] "ggsignif"         "globals"         "glue"
## [79] "googledrive"      "googlesheets4"  "gower"
## [82] "graphics"         "grDevices"       "grid"
## [85] "gridExtra"         "gstat"            "gttable"
## [88] "hardhat"          "haven"           "highr"
## [91] "hms"              "htmltools"       "htmlwidgets"
## [94] "httpuv"           "httr"             "ids"
## [97] "intervals"        "ipred"            "isoband"
## [100] "iterators"        "jquerylib"       "jsonlite"
## [103] "KernSmooth"       "knitr"            "labeling"
## [106] "later"            "lattice"          "lava"
## [109] "lazyeval"         "leaps"            "lifecycle"
## [112] "lightgbm"          "listenv"          "lme4"
## [115] "lubridate"        "magrittr"         "maptools"
## [118] "markdown"          "MASS"             "Matrix"
## [121] "MatrixModels"     "MBA"              "memoise"
## [124] "methods"          "mgcv"             "mice"
## [127] "mime"              "miniUI"           "minqa"
## [130] "missMDA"          "ModelMetrics"    "modelr"
## [133] "munsell"          "mvtnorm"         "nlme"
```

```

## [136] "nloptr"
## [139] "openssl"
## [142] "pbkrtest"
## [145] "pkgload"
## [148] "praise"
## [151] "processx"
## [154] "progressr"
## [157] "ps"
## [160] "R6"
## [163] "rappdirs"
## [166] "RcppEigen"
## [169] "recipes"
## [172] "reprex"
## [175] "rlang"
## [178] "rprojroot"
## [181] "rvest"
## [184] "scatterplot3d"
## [187] "shinydashboard"
## [190] "sourcetools"
## [193] "SparseM"
## [196] "SQUAREM"
## [199] "stringi"
## [202] "sys"
## [205] "testthat"
## [208] "tidyverse"
## [211] "timechange"
## [214] "tools"
## [217] "utf8"
## [220] "vctrs"
## [223] "vroom"
## [226] "xfun"
## [229] "xtable"
## [232] "zoo"

library(MASS)

```

STAT560 - Foundation of Data Science - Lecture 4

Import Data from Excel to R

Save the data file as .csv (comma separated value) or .txt (tab delimited text).

You may use the following command to read in .csv file. .csv file is most common and usually a better way to go.

```
# myData=read.csv(file.choose())
# myData=read.table(file.choose(),header=T, sep=",")
```

You may use the following command to read in .txt file.

```
# myData=read.delim(file.choose())
# myData=read.table(file.choose(),header=T, sep="\t")
```

You may also use the package “readr” of reading a .csv or .txt file

```
# library(readr)
# myData=read_csv(file.choose())
```

You can also directly import Excel (.xlsx) file using the package “readxl”.

```
# library(readxl)
# myData=read_excel(file.choose())

myData=read.csv(file="https://www.openintro.org/data/csv/loan50.csv",header=T)
myData[1:5,]
```

```
##   state emp_length term homeownership annual_income verified_income
## 1    NJ          3    60        rent      59000     Not Verified
## 2    CA         10    36        rent      60000     Not Verified
## 3    SC          NA   36     mortgage     75000     Verified
## 4    CA          0    36        rent      75000     Not Verified
## 5    OH          4    60     mortgage     254000     Not Verified
##   debt_to_income total_credit_limit total_credit_utilized
## 1      0.5575254           95131            32894
## 2      1.3056833           51929            78341
## 3      1.0562800          301373            79221
## 4      0.5743467           59890            43076
## 5      0.2381496          422619            60490
##   num_cc_carrying_balance   loan_purpose loan_amount grade interest_rate
## 1                      8 debt_consolidation     22000     B       10.90
## 2                      2 credit_card        6000     B       9.92
## 3                     14 debt_consolidation     25000     E       26.30
## 4                      10 credit_card        6000     B       9.92
## 5                      2 home_improvement     25000     B       9.43
##   public_record_bankrupt loan_status has_second_income total_income
## 1                      0 Current             FALSE      59000
## 2                      1 Current             FALSE      60000
## 3                      0 Current             FALSE      75000
```

```

## 4          0    Current      FALSE     75000
## 5          0    Current      FALSE    254000
nrow(myData)

## [1] 50
ncol(myData)

## [1] 18

```

Wrangle data in R

This lecture introduce the basics of how to wrangle data in R. Wrangling skills will provide an intellectual and practical foundation for working with modern data.

We going to using the dplyr package. The dplyr package presents a grammar for data wrangling (H. Wickham and Francois 2020). This package is loaded when library(tidyverse) is run. You may find detailed user manual and useful information of the dplyr package here: <https://dplyr.tidyverse.org/>

The most common way to extract data from data tables is with SQL (structured query language). We'll introduce SQL in CMIS563. Once you understand data wrangling with dplyr, it's straightforward to learn SQL.

```
install.packages("tidyverse")
```

```
library(tidyverse)
```

```

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## vforcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr    1.3.0
## v purrr    1.0.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

```

select() and filter()

The function filter() returns a subset of the rows of a data frame.

```
filter(myData, term == 60)
```



```

##   state emp_length term homeownership annual_income verified_income
## 1   NJ        3    60       rent      59000    Not Verified
## 2   OH        4    60      mortgage  254000    Not Verified
## 3   FL        6    60       rent      34000    Not Verified
## 4   FL        3    60      mortgage  80000  Source Verified
## 5   TX        2    60      mortgage  98000    Verified
## 6   HI        5    60       rent      68700  Source Verified
## 7   CA        3    60       rent      43000  Source Verified
## 8   FL       10    60      mortgage  76000  Source Verified
## 9   WI       10    60      mortgage  50000    Verified
## 10  NJ        5    60      mortgage  42000  Source Verified
## 11  RI       10    60      mortgage  245000    Verified
## 12  MA        4    60      mortgage  80000    Not Verified
## 13  CA       10    60      mortgage 160000    Verified

```

```

## 14      IL       10   60          rent      71000 Not Verified
## debt_to_income total_credit_limit total_credit_utilized
## 1      0.55752542           95131      32894
## 2      0.23814961           422619      60490
## 3      0.69750000           87705      23715
## 4      0.16685417           330394      32036
## 5      1.10710204           319551      108496
## 6      1.49954876           119325      103019
## 7      0.15465278           60693      22270
## 8      0.58981579           249700      44826
## 9      0.52341667           375828      87934
## 10     0.08666667           64800       3640
## 11     0.16736327           495163      41004
## 12     0.24454128           432092      53310
## 13     0.33440000           581604      53504
## 14     0.74252113           115970      52719

## num_cc_carrying_balance      loan_purpose loan_amount grade interest_rate
## 1                           8 debt_consolidation    22000     B    10.90
## 2                           2 home_improvement    25000     B    9.43
## 3                           10 credit_card        10000     A    7.97
## 4                           4 debt_consolidation    18500     C    12.62
## 5                           6 credit_card        29400     E    24.85
## 6                           5 debt_consolidation    13500     D    18.06
## 7                           4 debt_consolidation    18200     B    10.42
## 8                           10 debt_consolidation   18000     D    19.42
## 9                           11 debt_consolidation   40000     D    20.00
## 10                          5 house            20000     B    10.91
## 11                          6 credit_card        35000     B    11.98
## 12                          5 debt_consolidation   18000     B    9.93
## 13                          3 debt_consolidation   38500     C    12.62
## 14                          8 car              24000     B    10.90

## public_record_bankrupt loan_status has_second_income total_income
## 1                      0 Current           FALSE    59000
## 2                      0 Current           FALSE   254000
## 3                      0 Current           FALSE   34000
## 4                      1 Current           TRUE    192000
## 5                      0 Fully Paid        FALSE   98000
## 6                      0 Current           FALSE   68700
## 7                      0 Current           TRUE   144000
## 8                      0 Current           FALSE   76000
## 9                      0 Fully Paid        TRUE   168000
## 10                     0 Current           FALSE   42000
## 11                     0 Current           FALSE   245000
## 12                     0 Current           TRUE   218000
## 13                     1 Current           FALSE   160000
## 14                     0 Current           FALSE   71000

```

myData %>% filter(term==60&grade=="B")



```

## state emp_length term homeownership annual_income verified_income
## 1   NJ      3   60          rent      59000 Not Verified
## 2   OH      4   60         mortgage    254000 Not Verified
## 3   CA      3   60          rent      43000 Source Verified
## 4   NJ      5   60         mortgage    42000 Source Verified
## 5   RI     10   60         mortgage    245000      Verified

```

```

## 6 MA 4 60 mortgage 80000 Not Verified
## 7 IL 10 60 rent 71000 Not Verified
## debt_to_income total_credit_limit total_credit_utilized
## 1 0.55752542 95131 32894
## 2 0.23814961 422619 60490
## 3 0.15465278 60693 22270
## 4 0.08666667 64800 3640
## 5 0.16736327 495163 41004
## 6 0.24454128 432092 53310
## 7 0.74252113 115970 52719
## num_cc_carrying_balance loan_purpose loan_amount grade interest_rate
## 1 debt_consolidation 22000 B 10.90
## 2 home_improvement 25000 B 9.43
## 3 debt_consolidation 18200 B 10.42
## 4 house 20000 B 10.91
## 5 credit_card 35000 B 11.98
## 6 debt_consolidation 18000 B 9.93
## 7 car 24000 B 10.90
## public_record_bankrupt loan_status has_second_income total_income
## 1 Current FALSE 59000
## 2 Current FALSE 254000
## 3 Current TRUE 144000
## 4 Current FALSE 42000
## 5 Current FALSE 245000
## 6 Current TRUE 218000
## 7 Current FALSE 71000

```

myData %>% filter(term==60&grade=="B"&annual_income>100000)



```

## state emp_length term homeownership annual_income verified_income
## 1 OH 4 60 mortgage 254000 Not Verified
## 2 RI 10 60 mortgage 245000 Verified
## debt_to_income total_credit_limit total_credit_utilized
## 1 0.2381496 422619 60490
## 2 0.1673633 495163 41004
## num_cc_carrying_balance loan_purpose loan_amount grade interest_rate
## 1 home_improvement 25000 B 9.43
## 2 credit_card 35000 B 11.98
## public_record_bankrupt loan_status has_second_income total_income
## 1 Current FALSE 254000
## 2 Current FALSE 245000

```

The function select() returns a subset of the columns of a data frame.



myData %>% select(term,annual_income,loan_amount)

```

## term annual_income loan_amount
## 1 60 59000 22000
## 2 36 60000 6000
## 3 36 75000 25000
## 4 36 75000 6000
## 5 60 254000 25000
## 6 36 67000 6400
## 7 36 28800 3000
## 8 36 80000 14500
## 9 60 34000 10000

```

```

## 10    60      80000     18500
## 11    36      73000     17000
## 12    36     120000     12000
## 13    36     100000     16000
## 14    36     105000     16500
## 15    36      34000      3000
## 16    60      98000     29400
## 17    60      68700     13500
## 18    60      43000     18200
## 19    36     100000     6000
## 20    60      76000     18000
## 21    36      80000      5000
## 22    60      50000     40000
## 23    36      80000      4500
## 24    36     185000     15000
## 25    60      42000     20000
## 26    36     100000     20000
## 27    36      65000     10000
## 28    36      45000      9000
## 29    60     245000     35000
## 30    36      50000     40000
## 31    36     116000    13125
## 32    36      30000     32000
## 33    60      80000     18000
## 34    36     103000     30000
## 35    36      58000     4400
## 36    36      55000      7000
## 37    36      65000     25000
## 38    36      32000      6500
## 39    36     325000     35000
## 40    36      70000      7500
## 41    60     160000    38500
## 42    36      50000     15000
## 43    60      71000     24000
## 44    36      85000     24000
## 45    36      87000     12800
## 46    36      60000      6000
## 47    36      58500     30000
## 48    36      50000      5825
## 49    36     103000     20000
## 50    36      77500     15000

```

Combining the filter() and select() commands.

```
select(filter(myData, term==60&grade=="B"), interest_rate,loan_amount)
```

```

##   interest_rate loan_amount
## 1      10.90     22000
## 2       9.43     25000
## 3      10.42     18200
## 4      10.91     20000
## 5      11.98     35000
## 6       9.93     18000
## 7      10.90     24000

```

```

myData %>%
  filter(term==60&grade=="B") %>%
  select(interest_rate,loan_amount)

##   interest_rate loan_amount
## 1      10.90     22000
## 2      9.43      25000
## 3      10.42     18200
## 4      10.91     20000
## 5      11.98     35000
## 6      9.93      18000
## 7      10.90     24000

```

mutate() and rename()



The mutate() function can be used to modify the data in an existing column, or modify the data and create a new column.

```

myData=myData %>% mutate(perc_credit_utilized=total_credit_utilized/total_credit_limit*100)
myData=myData %>% mutate(term=term/12)
myData[1:5,]

```

```

##   state emp_length term homeownership annual_income verified_income
## 1 NJ            3    5          rent      59000 Not Verified
## 2 CA            10   3          rent      60000 Not Verified
## 3 SC            NA   3        mortgage    75000 Verified
## 4 CA             0   3          rent      75000 Not Verified
## 5 OH            4    5        mortgage    254000 Not Verified
##   debt_to_income total_credit_limit total_credit_utilized
## 1 0.5575254        95131           32894
## 2 1.3056833        51929           78341
## 3 1.0562800        301373          79221
## 4 0.5743467        59890           43076
## 5 0.2381496        422619          60490
##   num_cc_carrying_balance loan_purpose loan_amount grade interest_rate
## 1                      8 debt_consolidation    22000    B      10.90
## 2                      2 credit_card       6000    B      9.92
## 3                      14 debt_consolidation   25000    E      26.30
## 4                      10 credit_card       6000    B      9.92
## 5                      2 home_improvement  25000    B      9.43
##   public_record_bankrupt loan_status has_second_income total_income
## 1                      0 Current        FALSE      59000
## 2                      1 Current        FALSE      60000
## 3                      0 Current        FALSE      75000
## 4                      0 Current        FALSE      75000
## 5                      0 Current        FALSE      254000
##   perc_credit_utilized
## 1      34.57758
## 2      150.86175
## 3      26.28669
## 4      71.92520
## 5      14.31313

myData=myData %>% mutate(adj_income = ifelse(verified_income == "Not Verified", annual_income*0.75, annual_income))
myData[1:3,]

```

```

## state emp_length term homeownership annual_income verified_income
## 1 NJ 3 5 rent 59000 Not Verified
## 2 CA 10 3 rent 60000 Not Verified
## 3 SC NA 3 mortgage 75000 Verified
## debt_to_income total_credit_limit total_credit_utilized
## 1 0.5575254 95131 32894
## 2 1.3056833 51929 78341
## 3 1.0562800 301373 79221
## num_cc_carrying_balance loan_purpose loan_amount grade interest_rate
## 1 8 debt_consolidation 22000 B 10.90
## 2 2 credit_card 6000 B 9.92
## 3 14 debt_consolidation 25000 E 26.30
## public_record_bankrupt loan_status has_second_income total_income
## 1 0 Current FALSE 59000
## 2 1 Current FALSE 60000
## 3 0 Current FALSE 75000
## perc_credit_utilized adj_income
## 1 34.57758 44250
## 2 150.86175 45000
## 3 26.28669 75000

```

The `rename()` function is used to change the name of the columns.

```
myData=myData %>% rename(term_yr=term)
```



```

## state emp_length term_yr homeownership annual_income verified_income
## 1 NJ 3 5 rent 59000 Not Verified
## 2 CA 10 3 rent 60000 Not Verified
## 3 SC NA 3 mortgage 75000 Verified
## debt_to_income total_credit_limit total_credit_utilized
## 1 0.5575254 95131 32894
## 2 1.3056833 51929 78341
## 3 1.0562800 301373 79221
## num_cc_carrying_balance loan_purpose loan_amount grade interest_rate
## 1 8 debt_consolidation 22000 B 10.90
## 2 2 credit_card 6000 B 9.92
## 3 14 debt_consolidation 25000 E 26.30
## public_record_bankrupt loan_status has_second_income total_income
## 1 0 Current FALSE 59000
## 2 1 Current FALSE 60000
## 3 0 Current FALSE 75000
## perc_credit_utilized adj_income
## 1 34.57758 44250
## 2 150.86175 45000
## 3 26.28669 75000

```

arrange()

The function that will sort a data frame is called `arrange()`. Note that the default sort order is ascending order, so we do not need to specify an order if that is what we want. Use `desc()` if we want a descending order.

```
myData1=select(filter(myData, grade=="B"), term_yr,loan_amount,interest_rate, annual_income)
myData1 %>% arrange(desc(term_yr))
```

```

##   term_yr loan_amount interest_rate annual_income
## 1      5     22000      10.90      59000
## 2      5     25000       9.43     254000
## 3      5     18200      10.42      43000
## 4      5     20000      10.91      42000
## 5      5     35000      11.98     245000
## 6      5     18000       9.93      80000
## 7      5     24000      10.90      71000
## 8      3      6000       9.92      60000
## 9      3      6000       9.92      75000
## 10     3      6400       9.92      67000
## 11     3      4500       9.44      80000
## 12     3     15000       9.92     185000
## 13     3    13125      10.91     116000
## 14     3     32000       9.44      30000
## 15     3     24000       9.93      85000
## 16     3     12800       9.44      87000
## 17     3      6000      10.42      60000
## 18     3      5825      10.91      50000
## 19     3     20000       9.43     103000

```

```
myData1 %>% arrange(desc(term_yr), loan_amount)
```

```

##   term_yr loan_amount interest_rate annual_income
## 1      5     18000       9.93      80000
## 2      5     18200      10.42      43000
## 3      5     20000      10.91      42000
## 4      5     22000      10.90      59000
## 5      5     24000      10.90      71000
## 6      5     25000       9.43     254000
## 7      5     35000      11.98     245000
## 8      3      4500       9.44      80000
## 9      3      5825      10.91      50000
## 10     3      6000       9.92      60000
## 11     3      6000       9.92      75000
## 12     3      6000      10.42      60000
## 13     3      6400       9.92      67000
## 14     3     12800       9.44      87000
## 15     3    13125      10.91     116000
## 16     3     15000       9.92     185000
## 17     3     20000       9.43     103000
## 18     3     24000       9.93      85000
## 19     3     32000       9.44      30000

```

```
myData1 %>% arrange(desc(term_yr), loan_amount,annual_income)
```

```

##   term_yr loan_amount interest_rate annual_income
## 1      5     18000       9.93      80000
## 2      5     18200      10.42      43000
## 3      5     20000      10.91      42000
## 4      5     22000      10.90      59000
## 5      5     24000      10.90      71000
## 6      5     25000       9.43     254000
## 7      5     35000      11.98     245000
## 8      3      4500       9.44      80000

```

```

## 9      3     5825    10.91    50000
## 10     3     6000     9.92    60000
## 11     3     6000    10.42    60000
## 12     3     6000     9.92    75000
## 13     3     6400     9.92    67000
## 14     3    12800     9.44    87000
## 15     3   13125    10.91   116000
## 16     3   15000     9.92   185000
## 17     3   20000     9.43   103000
## 18     3   24000     9.93    85000
## 19     3   32000     9.44    30000

```

summarize() with group_by()

summarize() collapses a data frame into a single row of a summary statistics. The function n() simply counts the number of rows.

```

myData %>% summarize(
  N=n(),
  N_current_loan = sum(loan_status=="Current"),
  max_term = max(term_yr),
  avg_loan_amount = mean(loan_amount),
  min_interest = min(interest_rate),
)

## # A tibble: 1 x 5
##   N N_current_loan max_term avg_loan_amount min_interest
##   <int>           <dbl>      <dbl>            <dbl>
## 1 50             44         5       17083        5.31

```

Use summarize() with group_by() command to specify that the rows of data frame that should be grouped by a variable when calculate the summary statistics.

```

myData %>%
  group_by(grade) %>%
  summarize(
    N=n(),
    N_current_loan = sum(loan_status=="Current"),
    max_term = max(term_yr),
    avg_loan_amount = mean(loan_amount),
    min_interest = min(interest_rate),
  )

## # A tibble: 5 x 6
##   grade     N N_current_loan max_term avg_loan_amount min_interest
##   <chr> <int>           <dbl>      <dbl>            <dbl>
## 1 A         15            13         5       15133.        5.31
## 2 B         19            17         5       16518.        9.43
## 3 C          6            6          5       21000.       12.6
## 4 D          8            7          5       16612.       17.1
## 5 E          2            1          5       27200.       24.8

```

Data wrangling on multiple tables, inner_join() and left_join()

```

data_1 = myData %>% filter(term_yr==5) %>% select(grade,annual_income,interest_rate,loan_amount,total_credit_limit,loan_status)
##   grade annual_income interest_rate loan_amount total_credit_limit loan_status

```

```

## 1     B      59000    10.90    22000      95131    Current
## 2     B     254000     9.43    25000     422619    Current
## 3     A     34000     7.97   10000      87705    Current
## 4     C     80000    12.62   18500     330394    Current
## 5     E     98000    24.85   29400     319551 Fully Paid
## 6     D     68700    18.06   13500     119325    Current
## 7     B     43000    10.42   18200      60693    Current
## 8     D     76000    19.42   18000     249700    Current
## 9     D     50000    20.00   40000     375828 Fully Paid
## 10    B     42000    10.91   20000      64800    Current
## 11    B    245000    11.98   35000     495163    Current
## 12    B     80000     9.93   18000     432092    Current
## 13    C    160000    12.62   38500     581604    Current
## 14    B     71000    10.90   24000     115970    Current

nrow(data_1)

## [1] 14

data_2 = myData %>% filter(term_yr==5&grade=="B") %>% select(annual_income,total_credit_limit,homeownership)
data_2

##   annual_income total_credit_limit homeownership total_credit_utilized
## 1      59000           95131        rent            32894
## 2     254000          422619      mortgage         60490
## 3      43000           60693        rent            22270
## 4      42000           64800      mortgage           3640
## 5    245000          495163      mortgage         41004
## 6      80000          432092      mortgage         53310
## 7      71000           115970        rent            52719

nrow(data_2)

## [1] 7

The function inner_join() merges two data frame by matching up rows. The result set contains only those rows that have matches in both tables. The number of rows in the joined data is the number of matched rows only. Other rows are discarded.

data_joined = data_1 %>%
  inner_join(data_2, by = c("total_credit_limit" = "total_credit_limit"))
data_joined

##   grade annual_income.x interest_rate loan_amount total_credit_limit
## 1     B      59000     10.90     22000      95131
## 2     B     254000     9.43     25000     422619
## 3     B     43000    10.42    18200      60693
## 4     B     42000    10.91    20000      64800
## 5     B    245000    11.98    35000     495163
## 6     B     80000     9.93    18000     432092
## 7     B     71000    10.90    24000     115970
##   loan_status annual_income.y homeownership total_credit_utilized
## 1   Current      59000        rent            32894
## 2   Current     254000      mortgage         60490
## 3   Current      43000        rent            22270
## 4   Current      42000      mortgage           3640
## 5   Current    245000      mortgage         41004

```

```

## 6      Current      80000     mortgage      53310
## 7      Current      71000       rent        52719
nrow(data_joined)

```

```
## [1] 7
```

Another commonly-used type of join is a `left_join()`. Here the rows of the first table are always returned, regardless of whether there is a match in the second table. We retrieve all of the rows of the first data. NA's are inserted into the columns where no matched data was found. The number of rows of the resulting data frame is always the same as the first data.

```

data_leftjoined = data_1 %>%
  left_join(data_2, by = c("total_credit_limit" = "total_credit_limit"))
data_leftjoined

```

```

##   grade annual_income.x interest_rate loan_amount total_credit_limit
## 1    B      59000      10.90      22000          95131
## 2    B     254000      9.43      25000         422619
## 3    A      34000      7.97      10000          87705
## 4    C      80000      12.62      18500         330394
## 5    E      98000      24.85      29400         319551
## 6    D     68700      18.06      13500         119325
## 7    B     43000      10.42      18200          60693
## 8    D     76000      19.42      18000         249700
## 9    D     50000      20.00      40000         375828
## 10   B     42000      10.91      20000          64800
## 11   B    245000      11.98      35000         495163
## 12   B     80000      9.93      18000         432092
## 13   C    160000      12.62      38500         581604
## 14   B     71000      10.90      24000         115970
##   loan_status annual_income.y homeownership total_credit_utilized
## 1   Current      59000       rent        32894
## 2   Current     254000     mortgage      60490
## 3   Current        NA      <NA>        NA
## 4   Current        NA      <NA>        NA
## 5 Fully Paid        NA      <NA>        NA
## 6   Current        NA      <NA>        NA
## 7   Current     43000       rent        22270
## 8   Current        NA      <NA>        NA
## 9 Fully Paid        NA      <NA>        NA
## 10  Current     42000     mortgage      3640
## 11  Current     245000     mortgage      41004
## 12  Current     80000     mortgage      53310
## 13  Current        NA      <NA>        NA
## 14  Current     71000       rent        52719
nrow(data_leftjoined)

```

```
## [1] 14
```



STAT 560 Lecture 5: Summarizing Data with Graphs

Beidi Qiang

SIUE

Types of Data

This lecture focuses on the mechanics and construction of summary graphs. We use statistical software for generating the graphs in practice. In this lecture we take our time to detail how to create them. Graphing techniques for summarizing numerical and categorical variables are different.

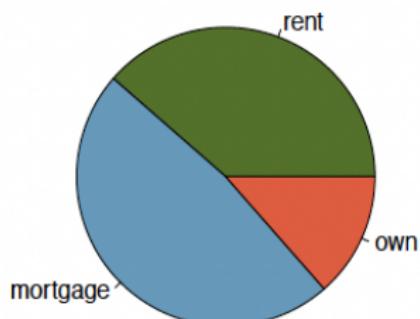
- ▶ **Categorical variables:** bar graph, pie chart,
- ▶ **Quantitative variables:** histogram, dotplot, boxplot (lecture 6), line graph (time series data), scatterplot (paired data)

Pie Charts and Bar Plots

- ▶ Pie Charts and Bar Plots are common ways to present a graphical summary of categorical data.
- ▶ Pie Charts present proportions as wedges in a circle.
- ▶ Bar Plots present proportions as vertical bars of various heights.

Pie Chart and Bar Plot Example

Pie Chart and Bar plot of the homeownership variable

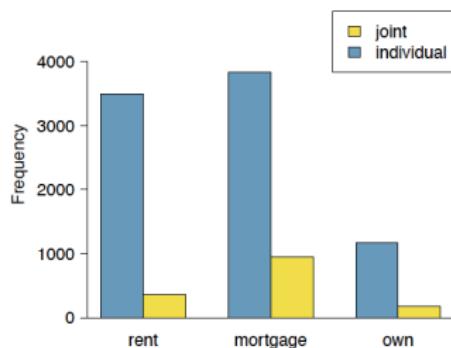
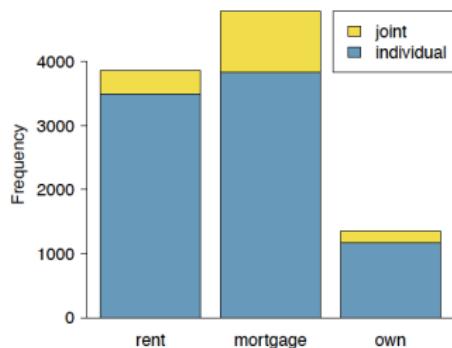


Pie Charts v.s. Bar Plot

- ▶ Pie charts can be useful for giving a high-level overview to show how a set of cases break down. However, it is also difficult to decipher details in a pie chart.
- ▶ We prefer bar plots for their ease in comparing groups.
- ▶ One may use a bar plot comparing two variables

Stacked and Side-by-side Bar plot

Bar plot of the homeownership variable and then divided each group by the levels of app type.



Dot plot

Dot plot provides the most basic of displays of numerical data. A dot on the number line represent an observation. You may use a stacked dot plot to see a rough shape of the data.

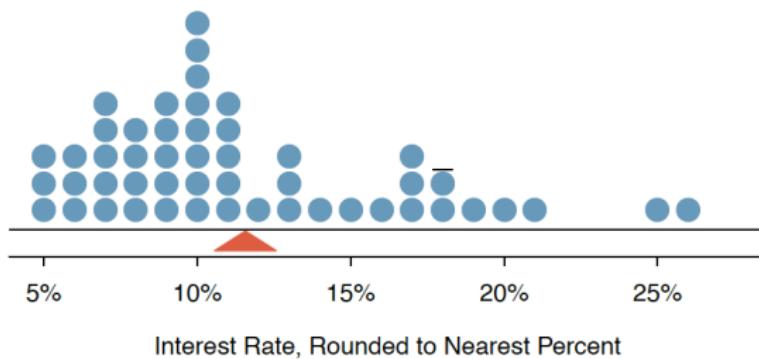


Figure 2.4: A stacked dot plot of `interest_rate` for the `loan50` data set. The rates have been rounded to the nearest percent in this plot, and the distribution's mean is shown as a red triangle.

Histogram

Dot plots show the exact value for each observation. This is useful for small data sets, but they can become hard to read with larger samples.

Histograms are a convenient way to show a large data by grouping values into “bins”.

Here are the steps:

- ▶ Divide the observations into intervals on the real number line (classes); keep the intervals of equal width.
 - ▶ Count the number of observations in each interval; make a table of these counts.
 - ▶ Record intervals on the horizontal axis and plot the counts on the vertical axis.

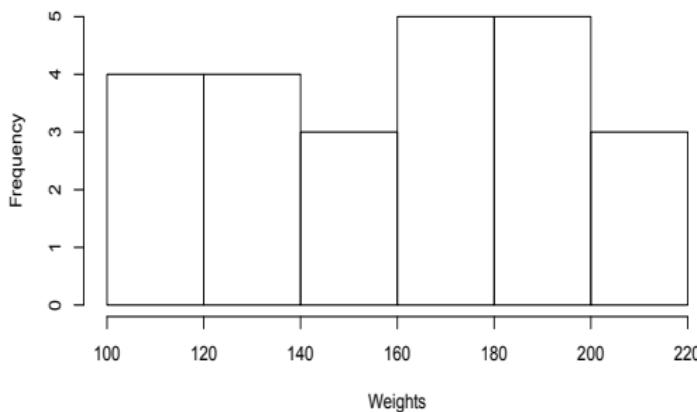
Histogram Example

Weights of a sample of 24 statistics students:

125 135 185 120 155 101 194 110 165 185 220 180
128 212 175 140 187 180 119 203 157 148 200 165

Bin (Class)	100-120	121-140	141-160	161-180	181-200	201-220
Frequency	4	4	3	5	5	3
Rel. Freq	16.67%	16.67%	12.5%	20.83 %	20.83%	12.5%

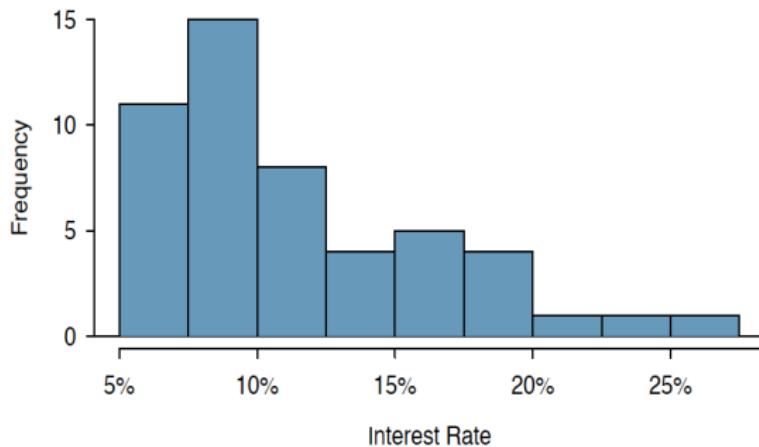
Histogram of Weights in Lb



Histogram Example

Interest Rate	5.0% - 7.5%	7.5% - 10.0%	10.0% - 12.5%	12.5% - 15.0%	...	25.0% - 27.5%
Count	11	15	8	4	...	1

Figure 2.5: Counts for the binned `interest_rate` data.



- ▶ Histograms provide a view of the data density. Higher bars represent where the data are relatively more common.
- ▶ How does this differ from a bar graph?
- ▶ We don't want too many or too few bins good software programs usually pick a reasonable number (\approx 6 to 12 classes).

Interpreting Histograms

Histograms are used to show the distribution of observations recorded for a quantitative variable. We will focus on the following characteristics when we examine and describe histograms:

- ▶ Overall pattern of the distribution
- ▶ Deviations from the overall pattern (e.g., outliers, etc.)

Overall Pattern

We determine the overall pattern based on the bulk of the data.

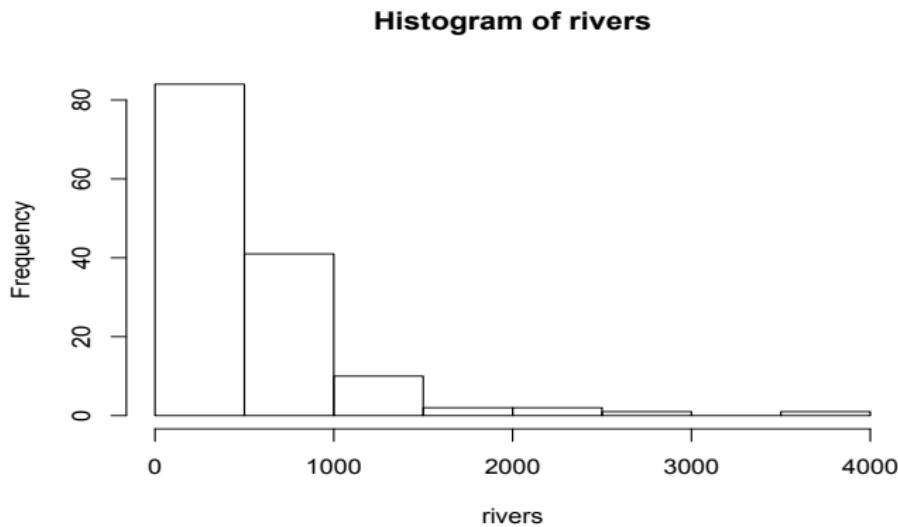
- ▶ Center: Where does the center of the distribution fall approximately?
- ▶ Spread: How much variation is in the distribution? How spread out is it? What is the range of possible values?
- ▶ Shape: What type of shape does the distribution have?

Shapes of Distributions

- ▶ A distribution is **symmetric** if the left and right sides of the histogram are approximately mirror images. Otherwise, the distribution is called skewed.
- ▶ A distribution is **skewed to the right** if the right side of the histogram extends much farther out than the left side ("long right tail").
- ▶ A distribution is **skewed to the left** if the left side of the histogram extends much farther out than the right side ("long left tail").

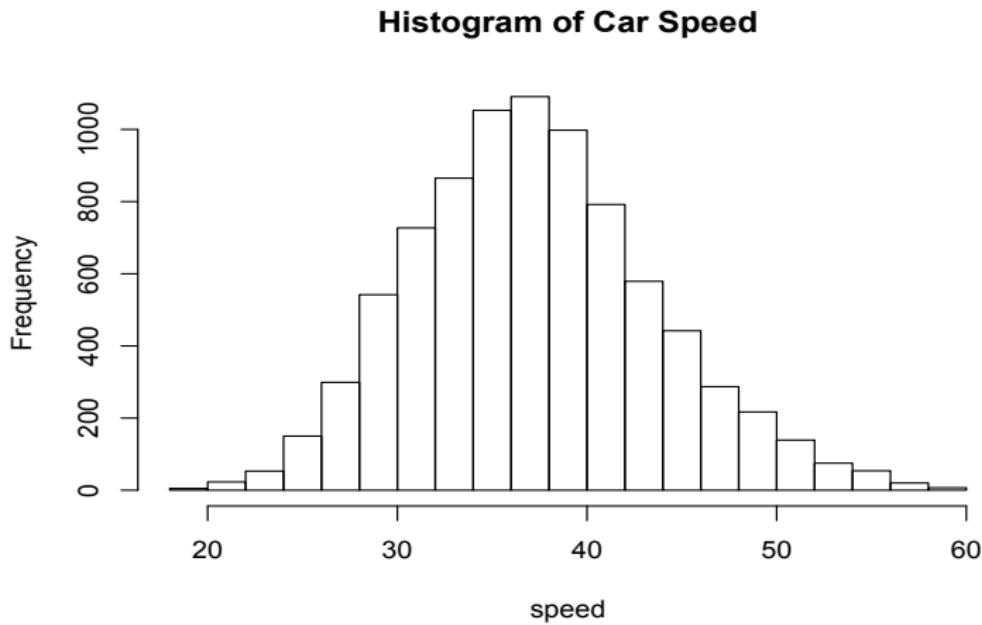
Examples

Histogram of the Lengths of Major North American Rivers:



Examples

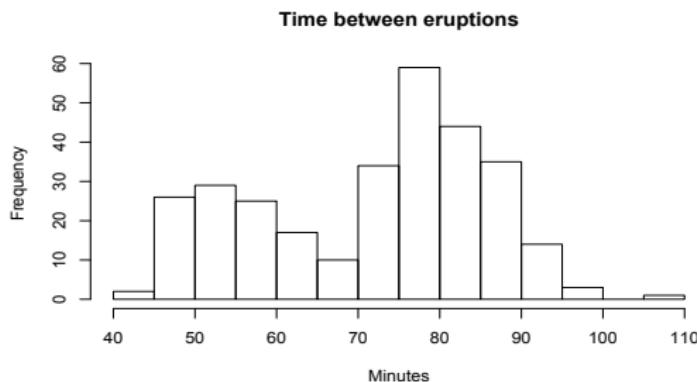
Histogram of the car speed at a crossroad:



Shapes of Distributions Cont'd

- ▶ A mode is represented by a prominent peak in the distribution.
- ▶ A distribution is unimodal if the histogram shows one dominant peak.
- ▶ A distribution is bimodal if the histogram shows two separate peaks.

Old Faithful geyser data

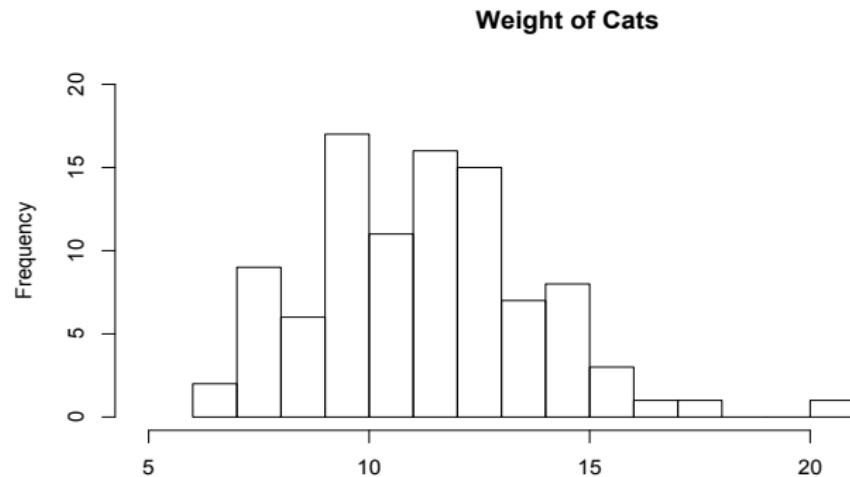


Definition: An **outlier** is an individual observation that falls outside the overall pattern of the distribution.

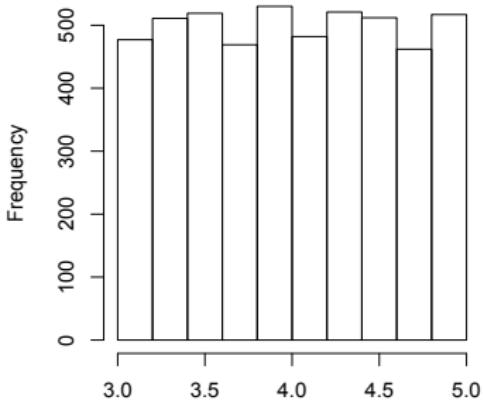
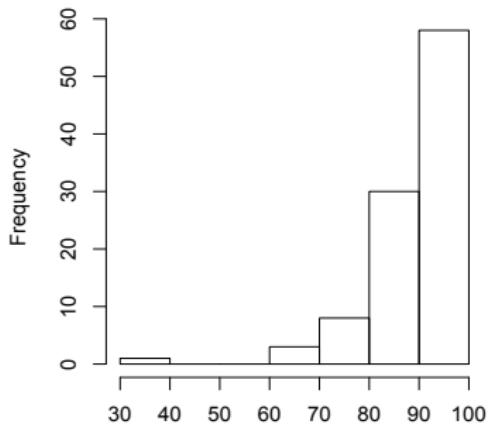
- ▶ May be a naturally occurring unusual value
- ▶ May be due to a recording error or measurement error
- ▶ May be an observation from a fundamentally different population
- ▶ When you see an outlier, go back and investigate that observation!

More Examples of Histogram

Weight Data for Domestic Cats:



More Examples of Histogram



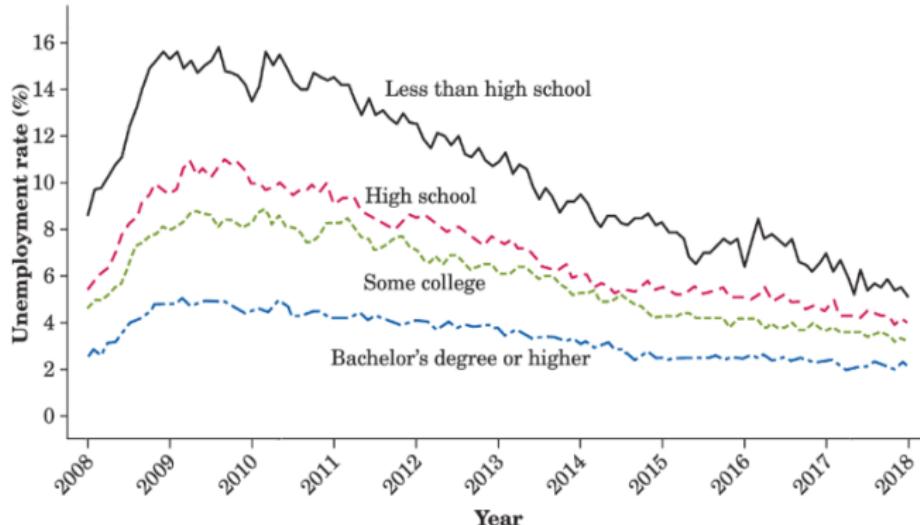
It is very common to observe quantitative data over time. Data that arise over time are called **time series** data. For example:

- ▶ Daily gas prices (in dollars)
 - ▶ Monthly sales (in dollars)
 - ▶ Daily high temperatures (in deg F)
 - ▶ Number of crime cases per month in the United States

Line Graphs show the pattern of variation of some variable over time

- ▶ Time (in years? months? days? hours?) is always plotted on the horizontal axis.
- ▶ The value of the variable is plotted on the vertical axis.

Line Graphs Example 2



- ▶ **Trends:** a long-term upward or downward movement over time.
Increasing trend? Decreasing trend? Constant trend?
- ▶ **Sharp deviations:** unusual observations or pattern show deviation from the overall trend.
- ▶ **Seasonal variation:** patterns that repeat themselves over time.
especially common when data are plotted monthly or weekly

More on Line Graphs

- ▶ Why are line graphs useful? We can look for trends and other patterns in the data. This information can be useful for making **predictions**.
- ▶ To predict the future number, your prediction should account for the overall trend and seasonal variation.
- ▶ If a variable is known vary seasonally, it's more instructive to first study the pattern after removing the seasonal variation.

Scatterplots for Paired Data

A **scatterplot** is a graphical display that shows the relationship between two numerical variables measured on the same individuals.

- ▶ Each individual in the data set has two variables measured on it.
- ▶ The values of one variable appear on the horizontal axis; the values of the other variable appear on the vertical axis.
- ▶ On the plot, there is a dot for each observation in the data set.
- ▶ Scatterplots give a visual impression of how the two variables behave together.

Example:

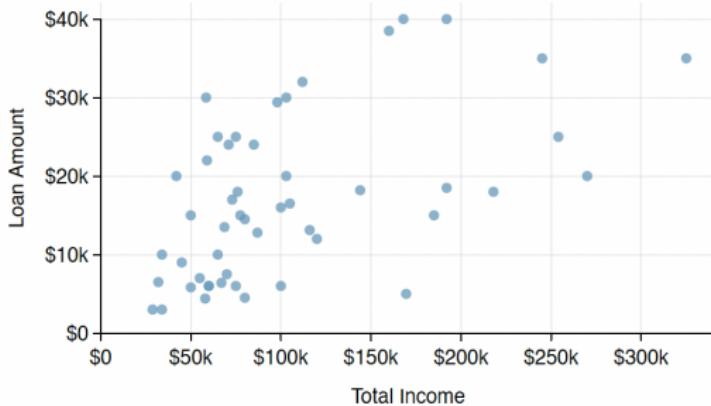


Figure 2.1: A scatterplot of `total_income` versus `loan_amount` for the `loan50` data set.

Explanatory v.s. Response Variables

- ▶ We are studying the relationship of 2 variables.
- ▶ Recall, sometimes one variable (called the explanatory variable) may naturally explain or predict the value of the other variable (called the response variable).
- ▶ If so, conventionally, the explanatory variable is denoted X and plotted on the X-axis. The response variable is denoted Y and plotted on the Y-axis.
- ▶ Other times, there is no natural explanatory-response relationship between the two variables. (e.g. Height/Weight)

Interpreting Scatterplots

We will focus on the following characteristics when we examine and describe scatterplots

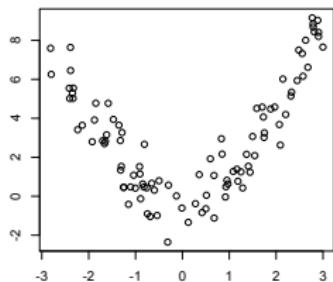
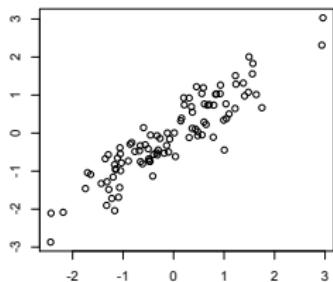
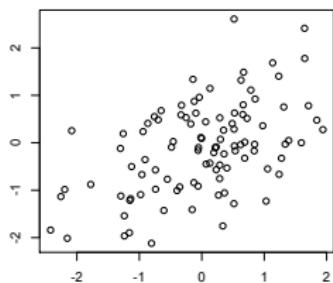
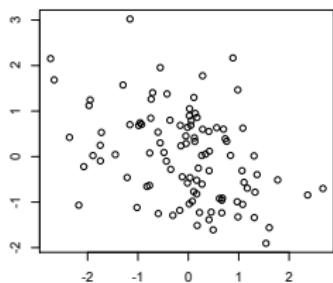
- ▶ **Form:** Are there straight-line (linear) patterns or curved patterns? Do observations tend to fall into clusters?
- ▶ **Direction:** Are the variables positively related or negatively related?
- ▶ **Strength:** Is the relationship strong, moderate, or mild? Perhaps there is no relationship (e.g., a random scatter of points)?

- ▶ Two quantitative variables are positively related when an increase in one variable tends to accompany an increase in the other.
- ▶ Two quantitative variables are negatively related when an increase in one variable tends to accompany a decrease in the other.
- ▶ The scatterplot for positively associated variables has a pattern that slopes upward from left to right. The scatterplot for negatively associated variables has a pattern that slopes downward.

Form and Strength of Association

- ▶ The **form** of the relationship between two variables may be approximately linear or curved.
- ▶ Once we identify the form of the relationship, we can characterize the **strength** of the relationship between the two variables.
- ▶ The relationship is **strong** if most of the data values closely follow the major trend in the plot.
- ▶ The relationship is **weak** if the data values show a great deal of random scatter around the major trend in the plot.

Examples of Scatterplots



STAT 560 Lecture 6 Summarizing Data with Numbers

Beidi Qiang

SIUE

We can also use numbers (statistics) to describe the **center** and **spread** of a sample data. In this lecture,

- ▶ We introduce commonly used numerical summaries of quantitative data
- ▶ We also introduce a new graph to display quantitative distributions - the **boxplot**.
- ▶ We introduce the Contingency tables for summarizing categorical variables.

Definition: The **median** (denoted M) of a data set is a numerical measure of the midpoint.

- ▶ Half of the observations are smaller than M ; half are larger.
- ▶ The median measures the **center** of a distribution.

Finding Median

- ▶ First order the data from smallest to largest.
- ▶ Find the midpoint of the ordered data.
- ▶ If the number of data values is odd, median is the data value in the midpoint position.
- ▶ If the number of data values is even, median is the average of the two data values around the midpoint.

Example: Finding Median

Trunk circumferences (in mm) of orange trees:

30 58 87 33 69 111 30 51 75 32 62 112 30 49 81

Using R:

```
> tree=c(30,58,87,33,69,111,30,51,75,32,62,112,30,49,81)
> sort(tree)
[1] 30 30 30 32 33 49 51 58 62 69 75 81 87 111 112
> median(tree)
[1] 58
```

The **quartiles** (denoted Q1, Q2, Q3) are numbers that divide the data set into quarters.

- ▶ Q2 is simply the median.
- ▶ The **first quartile**(Q1) is the median of the lower half of the observations.
- ▶ The **third quartile**(Q3) is the median of the upper half.

Example: Finding Quartiles

Trunk circumferences (in mm) of orange trees:

30 58 87 33 69 111 30 51 75 32 62 112 30 49 81

Sorted values:

30 30 30 32 33 49 51 58 62 69 75 81 87 111 112

Using R:

```
> quantile(tree,type=2)  
0% 25% 50% 75% 100%  
30 32 58 81 112
```

The Five-Number Summary

The 5-number summary consists of: The minimum value; Q1; the median; Q3; and the maximum value.

- ▶ The **minimum** (Min) is the smallest observation
- ▶ The **maximum** (Max) is the largest.
- ▶ These 5 numbers summarize information about **the center**, **the spread**, and **the tails** of the distribution.
- ▶ The median describes the **center** of the distribution.
- ▶ The distance between Q1 and Q3 describes the **spread** of the middle 50% of the data.(Interquartile Range)
- ▶ The minimum and maximum give information about the “**tails**” and possible outliers.

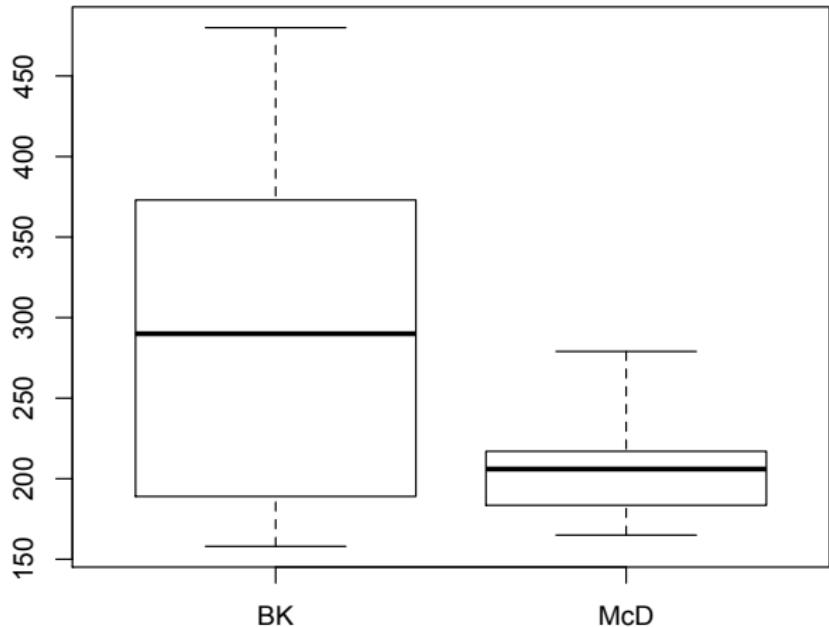
Example: Calorie data of fast food Sandwiches/Hamburgers

- ▶ McDonald's (sorted):
165 169 198 206 214 220 279
- ▶ Burger King (sorted):
158 164 170 189 190 250 290 311 315 373 398 456 480
- ▶ (See the complete data at <http://www.acaloriecounter.com/fast-food.php>)

A boxplot is a graphical presentation of the 5-number summary.

- ▶ A central box spans the quartiles Q1 and Q3.
- ▶ A solid line marks the median M.
- ▶ Lines extend from the box to the minimum and maximum values.
- ▶ Often multiple boxplots are placed in the same graph (using same axes) to compare numerical data across groups.

Example: Hamburgers Data



Example: Side-by-side Boxplot and Hollow Histograms

Besides side-by-side box plot, another useful plotting method uses hollow histograms to compare numerical data across groups. These are just the outlines of histograms of each group put on the same plot.

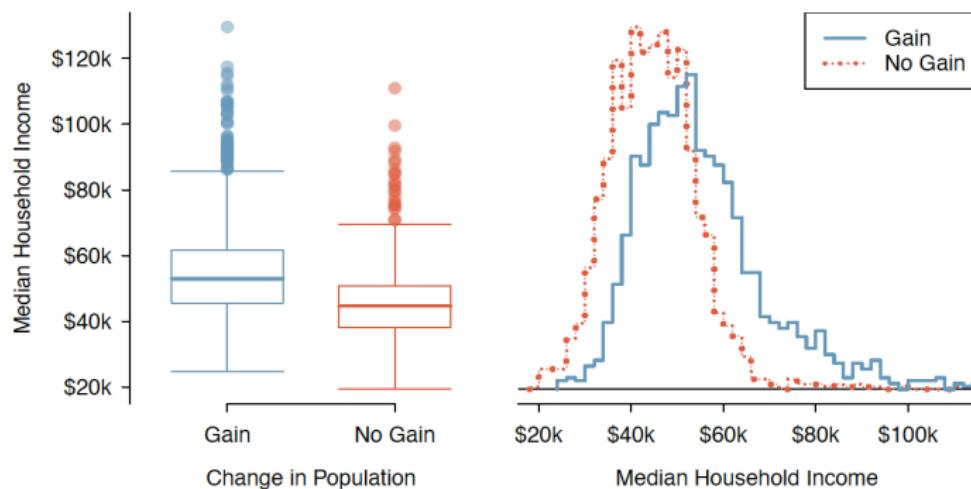


Figure 2.28: Side-by-side box plot (left panel) and hollow histograms (right panel) for `med_hh_income`, where the counties are split by whether there was a population gain or loss.

Interquartile Range

Definition: The **interquartile range** (IQR) of a distribution is the difference between the 1st and 3rd quartiles

$$\text{IQR} = Q_3 - Q_1$$

- ▶ The IQR measures the “spread” in the middle 50% of a distribution.
- ▶ The IQR is simply the length of the “box” part of a boxplot.

Example:

5-number summary of McDonald's: 165 169 206 220 279

5-number summary of Burger King: 158 189 290 373 480

Outliers

Typically an observation is labeled an outlier if it lies more than $1.5 \times \text{IQR}$ above Q3 or below Q1.

- ▶ Some computer packages produce boxplots whose extra lines ("whiskers") only extend to the "non-outlying values"
- ▶ The outliers may be marked with separate symbols on the plot.

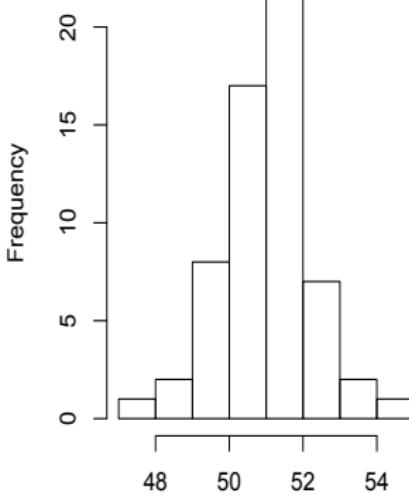
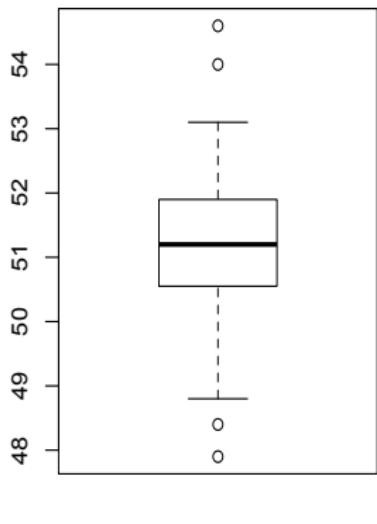
Example: Temperature in New Haven

The mean annual temperature in degrees Fahrenheit in New Haven, Connecticut (60 observations):

49.9 52.3 49.4 51.1 49.4 47.9 49.8 50.9 49.3 51.9 50.8 49.6 49.3 50.6 48.4 50.7 50.9 50.6
51.5 52.8 51.8 51.1 49.8 50.2 50.4 51.6 51.8 50.9 48.8 51.7 51.0 50.6 51.7 51.5 52.1 51.3
51.0 54.0 51.4 52.7 53.1 54.6 52.0 52.0 50.9 52.6 50.2 52.6 51.6 51.9 50.5 50.9 51.7 51.4
51.7 50.8 51.9 51.8 51.9 53.0

- ▶ 5-number summary: 47.90 50.55 51.20 51.90 54.60
- ▶ $1.5 \text{IQR} = 1.5 \times (51.9 - 50.55) = 2.025$
- ▶ $Q1 - 2.025 = 48.525$ and $Q3 + 2.025 = 53.925$

Example: Temperature in New Haven Cont'd



More on Boxplots

- ▶ A boxplot can indicate whether a distribution is symmetric or skewed.
- ▶ Does one half of the boxplot extend farther out than the other half?
- ▶ It's typically easier to determine shape and symmetry/skewness using a histogram.
- ▶ Side-by-side boxplots are good for comparisons of distributions (comparing center, spread, etc).

Other Measures of Center and Spread

- ▶ The **mean** is a measure of a data set's center, like the median.
- ▶ The **standard deviation** is a measure of a data set's spread, like the IQR.
- ▶ The variance is the square of the standard deviation. It is also a measure of the spread.

Mean

Definition: With a sample of observations x_1, x_2, \dots, x_n , the **mean** (\bar{x}) is the average of the n values.

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n}$$

- ▶ The mean \bar{x} is the balancing point of a distribution.
- ▶ McDonald's Data: 165 169 198 206 214 220 279

- ▶ Use R:
> mean(McD)
[1] 207.2857

Standard Deviation

Definition: With a sample of observations x_1, x_2, \dots, x_n , the **standard deviation** (s) is the average of the n values.

$$s = \sqrt{\frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}{n - 1}}$$

- ▶ The standard deviation measures how far observations are from the mean, on average.

- ▶ The hamburger example: (Use R)

```
> sd(McD)  
[1] 38.0601  
> sd(BK)  
[1] 112.814
```

Calculating Standard Deviation by Hand

Weights of a sample of 10 statistics students:

Individual	Observation	Distance from the mean	Squared Distance
1	125	$125 - 158 = -33$	1089
2	135	$135 - 158 = -23$	529
3	185	$185 - 158 = 27$	729
4	120	$120 - 158 = -38$	1444
5	155	$155 - 158 = -3$	9
6	110	$110 - 158 = -48$	2304
7	165	$165 - 158 = 7$	49
8	185	$185 - 158 = 27$	729
9	220	$220 - 158 = 62$	3844
10	180	$180 - 158 = 22$	484
Sum	1580	0	11210

$$s = \sqrt{11210/9} = 35.29243$$

- ▶ The larger the standard deviation, the more variation (spread) in the distribution.
- ▶ The smallest the standard deviation can be is 0. This occurs when all the observations are the same. There is **no spread**.
- ▶ The standard deviation is measured in the original units of the data.
- ▶ The **variance** is defined as the square of standard deviation. Not in the original units. Less meaningful.

We have discussed two statistics that describe the center of a distribution:
Median = midpoint of the distribution
Mean = average of the observations.

- ▶ The value of the mean can be heavily influenced by outliers.
- ▶ Outliers (on either side) do not affect the median greatly (robust).

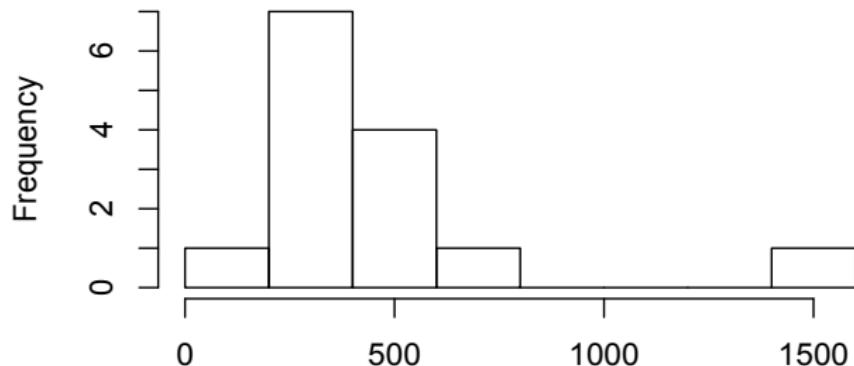
Standard Deviation v.s. IQR

Similarly, we have two statistics (standard deviation and IQR) to describe the spread

- ▶ The value of the standard deviation is more affected by outliers
- ▶ Outliers do not affect the IQR greatly (robust).

Example: Rivers in North America

The lengths (in miles) of some "major" rivers in North America:
735 320 325 392 524 450 1459 135 465 600 330 336 280 315



Example: Rivers in North America Cont'd

The lengths of rivers:

735 320 325 392 524 450 1459 135 465 600 330 336 280 315

- ▶ Center: mean=476.1, median=364
- ▶ Spread: std.dev=319.6, IQR=204.
- ▶ min=135, Q1=320, Q2=364, Q3=524, max=1459
- ▶ Outlier: < 14 or > 830

Example: Rivers in North America Cont'd

The lengths of rivers (outlier removed):

735 320 325 392 524 450 135 465 600 330 336 280 315

- ▶ Center: mean=400.5, median=336
- ▶ Spread: std.dev=154.7, IQR=145.

Frequency tables

A frequency table shows overall percentages or proportions for each categories of a categorical variable. Example:

homeownership	Count
rent	3858
mortgage	4789
own	1353
Total	10000

Contingency tables

A contingency table summarizes data for two categorical variables. Each value in the table represents the number of times a particular combination of variable outcomes occurred.

app_type	homeownership			Total
	rent	mortgage	own	
individual	3496	3839	1170	8505
joint	362	950	183	1495
Total	3858	4789	1353	10000

	rent	mortgage	own	Total
individual	0.411	0.451	0.138	1.000
joint	0.242	0.635	0.122	1.000
Total	0.386	0.479	0.135	1.000

STAT560 - Foundation of Data Science - Lecture 7

Using ggplot2 package

In this lecture, we illustrate the composition of a graph using the ggplot2 package. You may also create most of the commonly used graph with just the base graphics package in R. We employ the ggplot2 system because it is one of the most widely-used R packages, and it provides a comprehensive grammar for describing and specifying graphics. Detailed user manual and other useful information can be found on the website: <https://ggplot2.tidyverse.org/>

To use ggplot2, we first need to install (if you have not done so) and load the tidyverse package.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## vforcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyverse 1.3.0
## v purrr    1.0.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

We demonstrate the functions with the loan50 data that we introduced before.

```
myData=read.csv(file="https://www.openintro.org/data/csv/loan50.csv",header=T)
```

Composition of Graph

A data table provides the basis for drawing a data graphic. The relationship between a data table and a graphic is simple: Each case (row) in the data table becomes a mark in the graph. As the designer of the graphic, you choose which variables the graphic will display and how each variable is to be represented graphically: position, size, color, and so on.

Visual cues: graphical elements that draw the eye, such as position coordinates, shape, color, shade, size, etc.

Coordinate systems: Cartesian (the usual x-y coordinates), Polar, Geographic

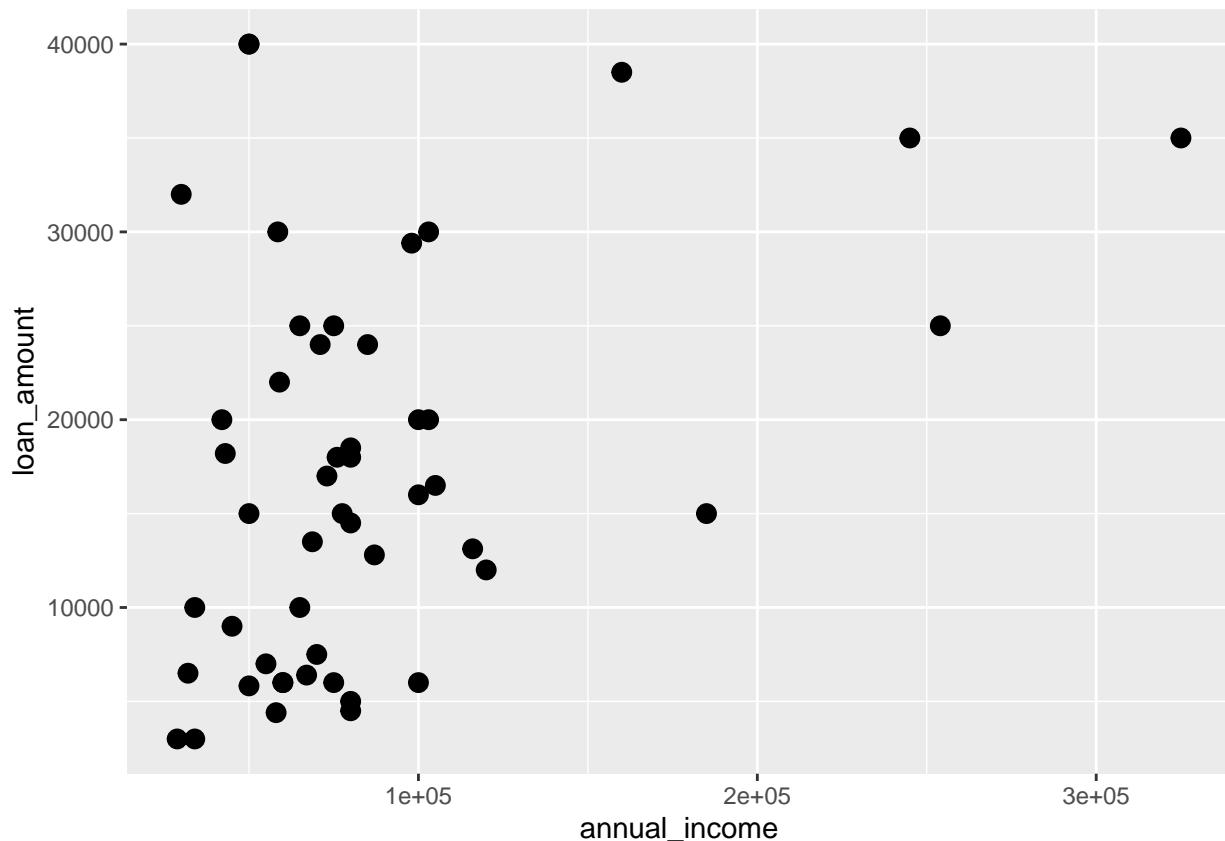
Scale: numeric (linear, logarithmic, or percentage) scale, Categorical (no ordering or ordinal), time

Context: annotations on titles or subtitles, axis labels or reference points or lines. Also called guides or legends.

Aesthetics in ggplots

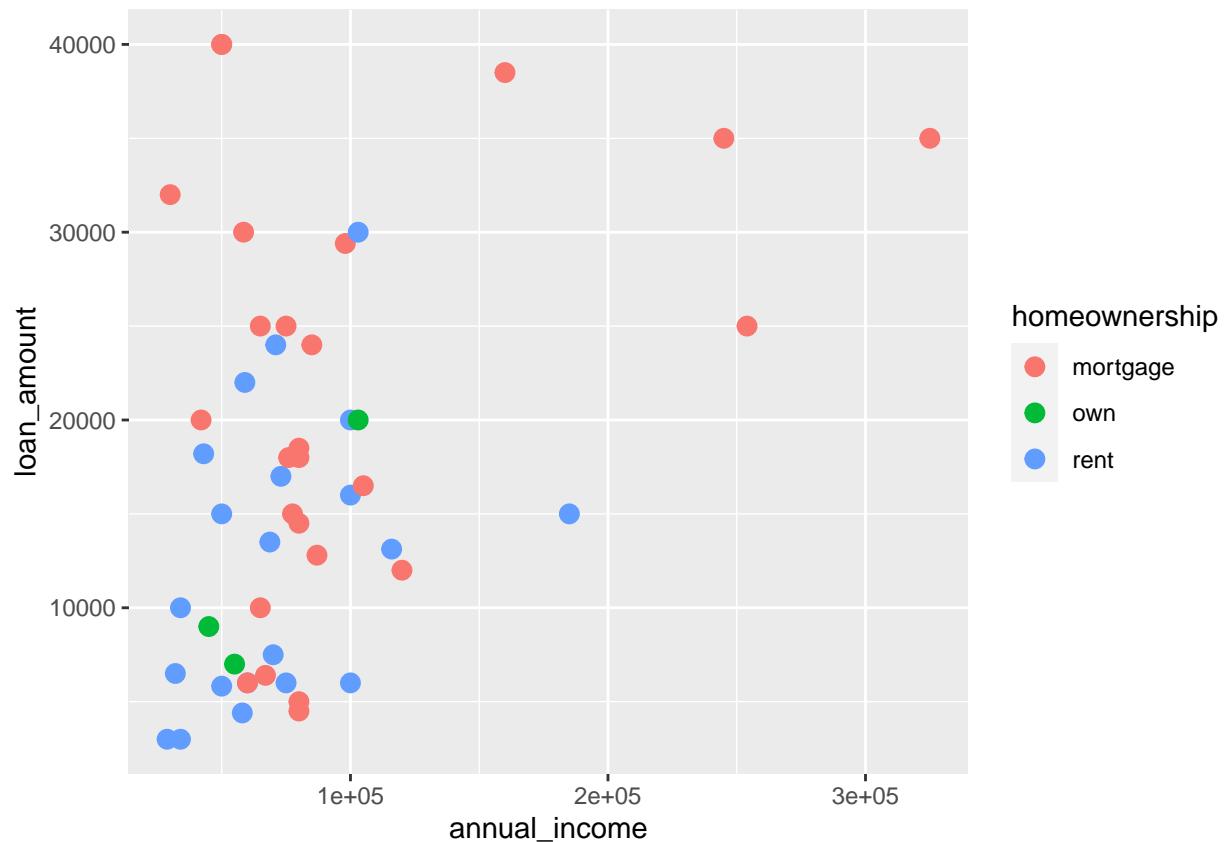
In ggplot2, an aesthetic is an explicit mapping between a variable and the visual cues that represent its values. Here the two aesthetics map the vertical coordinate to the loan_amount variable, and the horizontal coordinate to the annual_income variable.

```
g = ggplot(data = myData, aes(y = loan_amount, x = annual_income))  
g + geom_point(size = 3)
```



You may add additional aesthetics. Here we map color of the dot to homeownership:

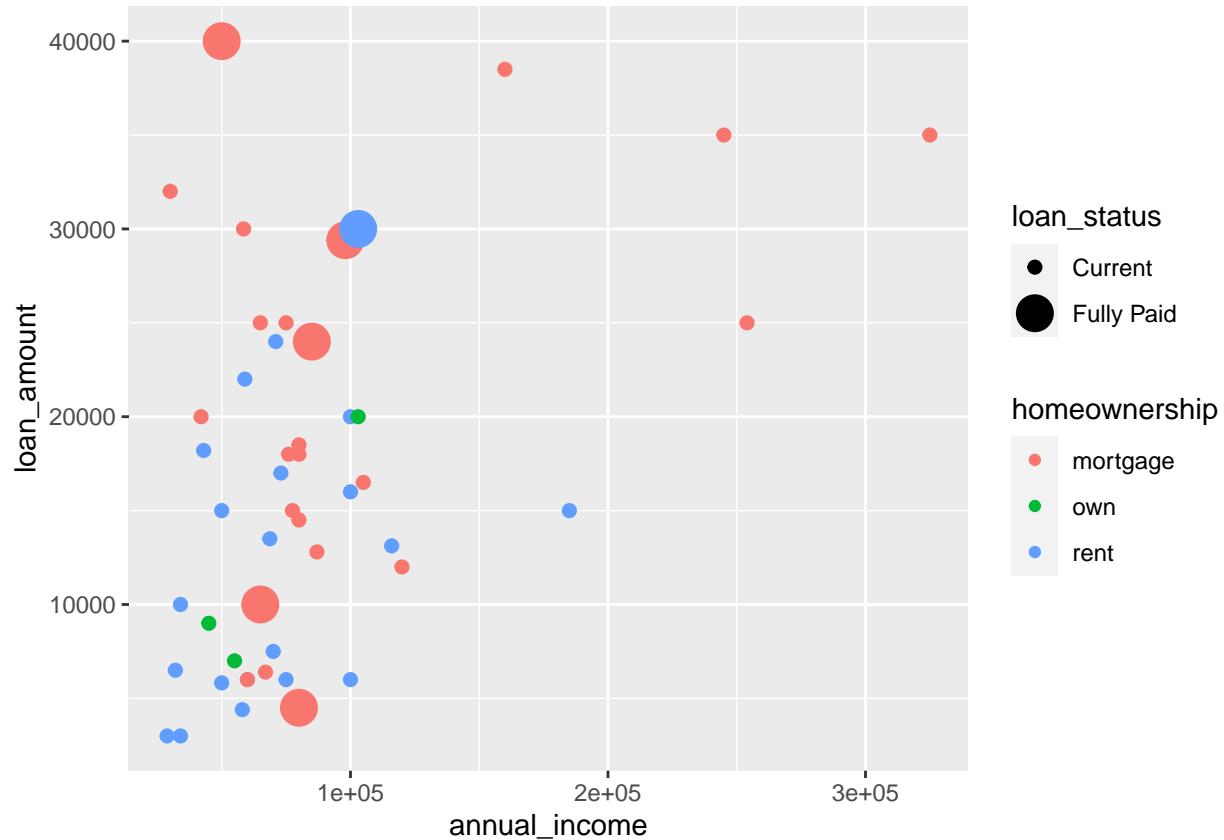
```
g + geom_point(aes(color = homeownership), size = 3)
```



You may employ multiple aesthetics:

```
g + geom_point(aes(color = homeownership, size = loan_status))
```

Warning: Using size for a discrete variable is not advised.



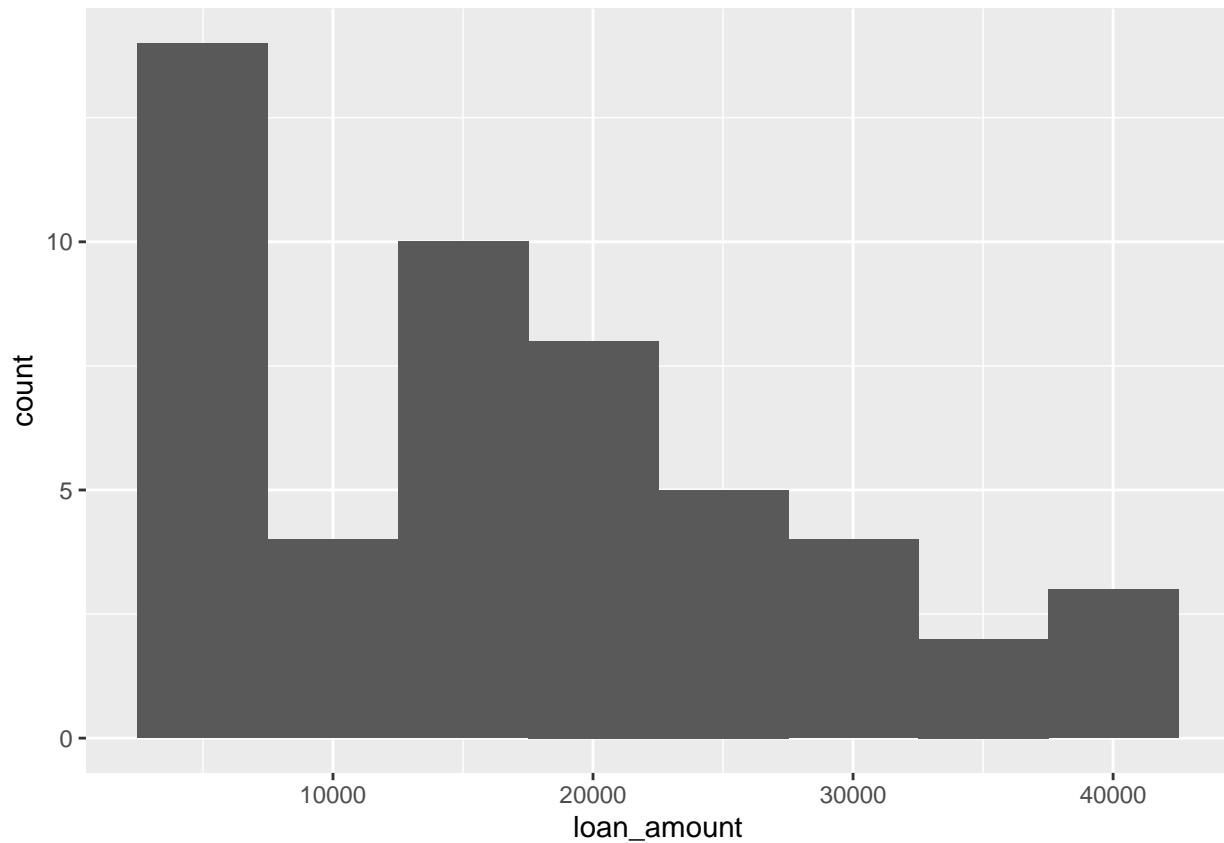
You may also change the type of the marks:

```
g + geom_text(aes(label=state ,color = homeownership), size = 4)
```



You may also plot just one variable:

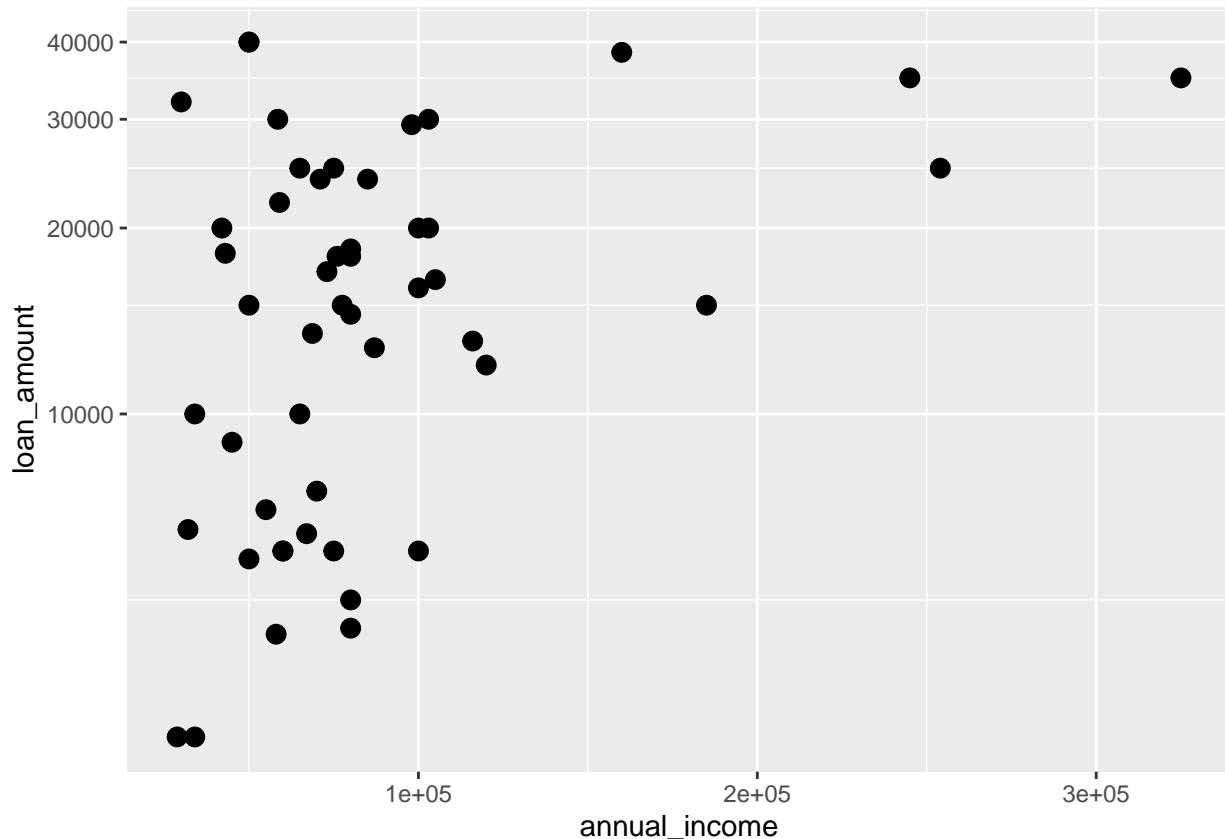
```
hg = ggplot(data=myData,aes(x=loan_amount))  
hg + geom_histogram(binwidth=5000)
```



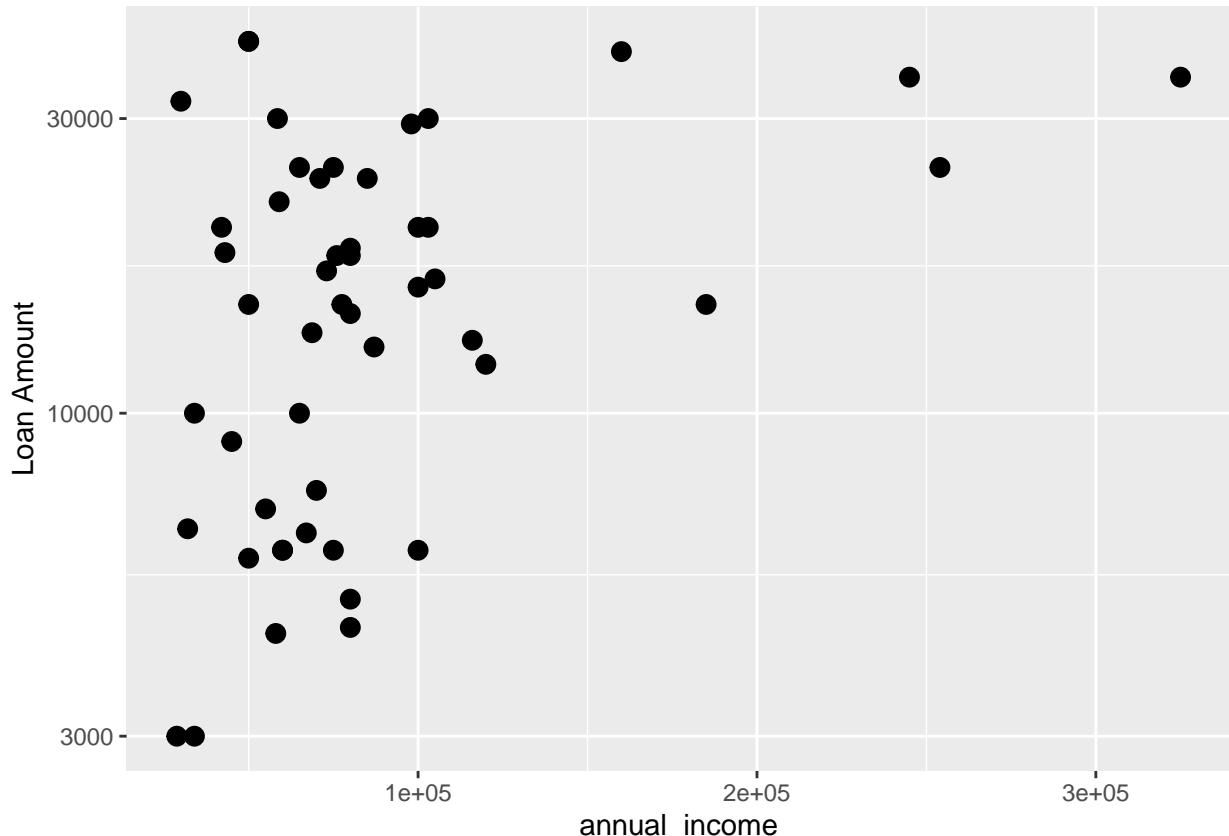
Scales

You may use the commands like `coord_trans()`, `scale_y_continuous()` or `scale_x_continuous()` in `ggplot2` to transform the scale of axis. The difference is how and where the major tick marks and axis labels are drawn. We prefer to use `coord_trans()`.

```
g = ggplot(data = myData, aes(y = loan_amount, x = annual_income))
g + geom_point(size = 3) + coord_trans(y = "log10")
```



```
g + geom_point(size = 3) +
  scale_y_continuous(
    name = "Loan Amount",
    trans = "log10"
  )
```



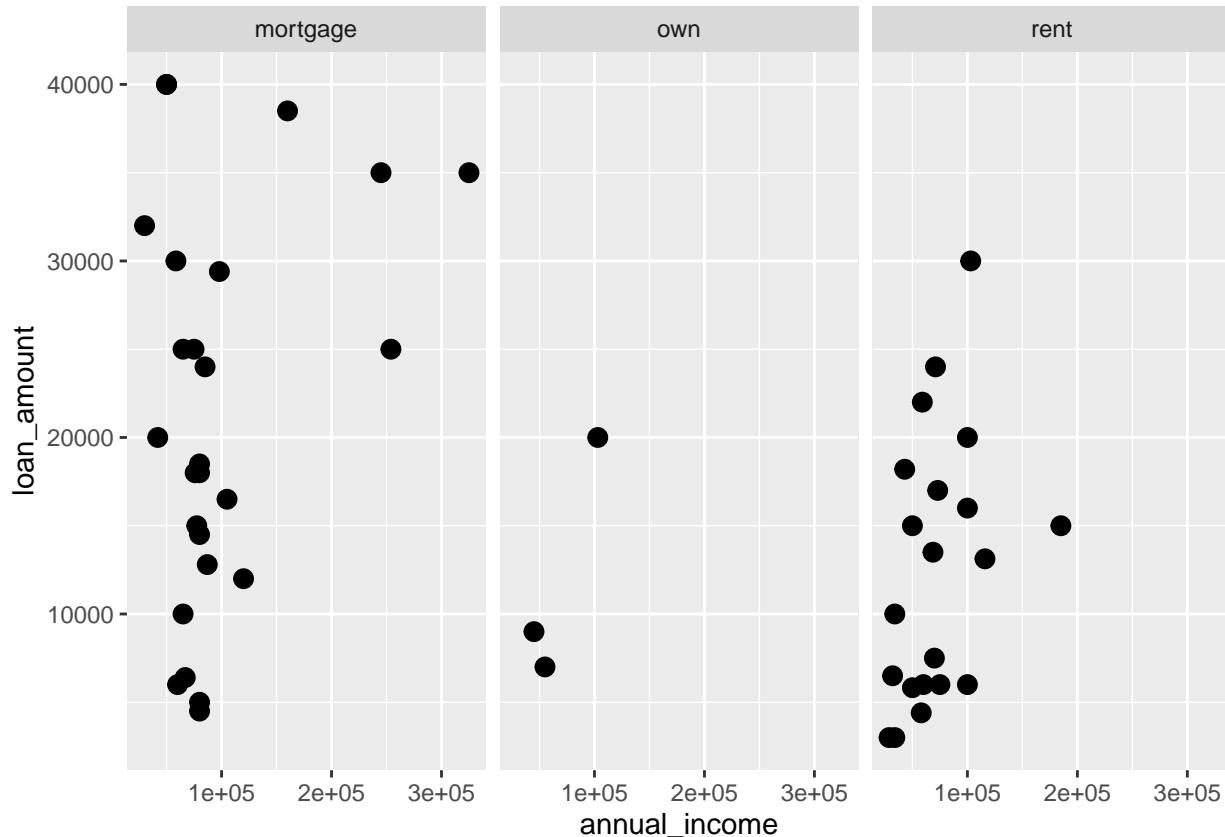
Facets

Facets are collections several small multiples of the same plot, with one (discrete) variable changing in each of the small sub-images.

Using many aesthetics to display multiple variables on a same graph can make the graph hard to read. We may want to create separate facets by the levels of a categorical or discrete variable. There are two functions that create facets: `facet_wrap()` and `facet_grid()`.

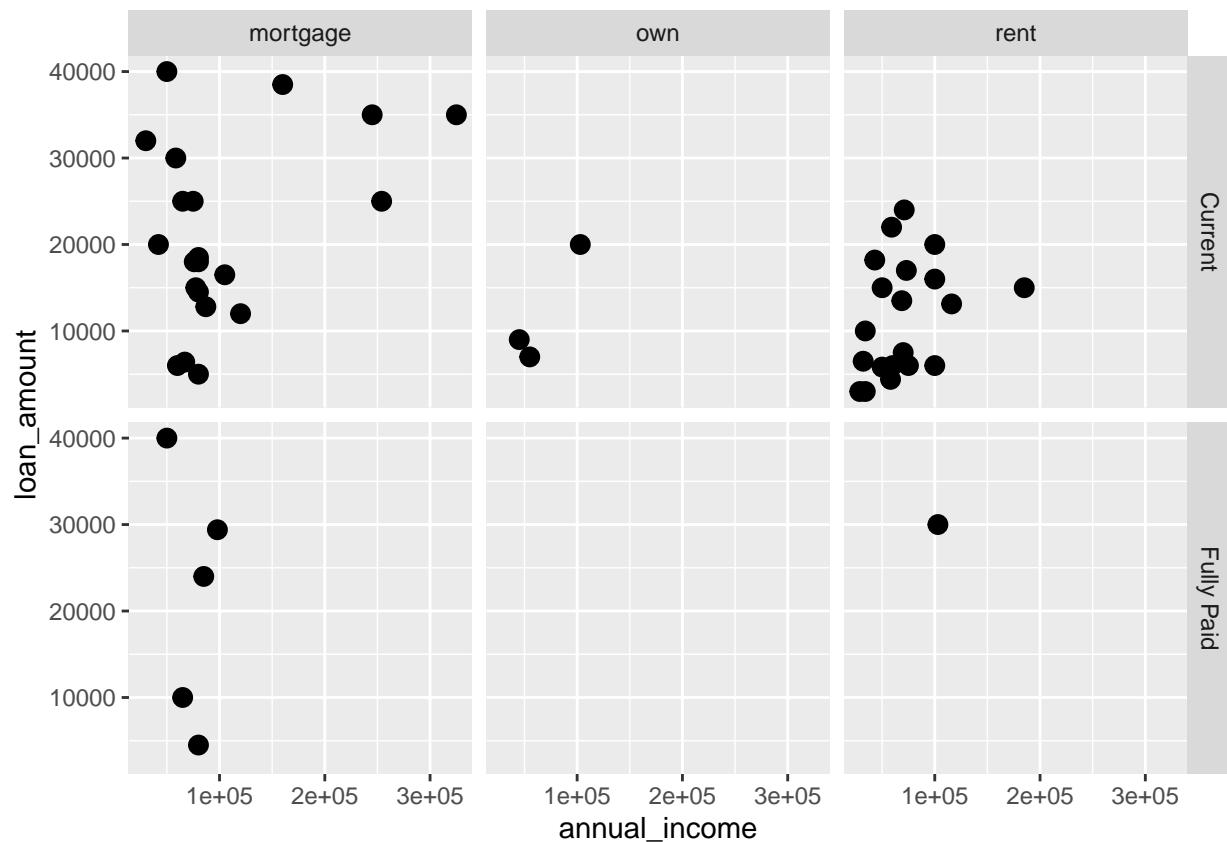
`facet_wrap()` creates a facet for each level of a single categorical or discrete variable,

```
g = ggplot(data = myData, aes(y = loan_amount, x = annual_income))
g + geom_point(size = 3) + facet_wrap(~homeownership, nrow = 1)
```



`facet_grid()` creates a facet for each combination of two categorical variables, arranging them in a grid.

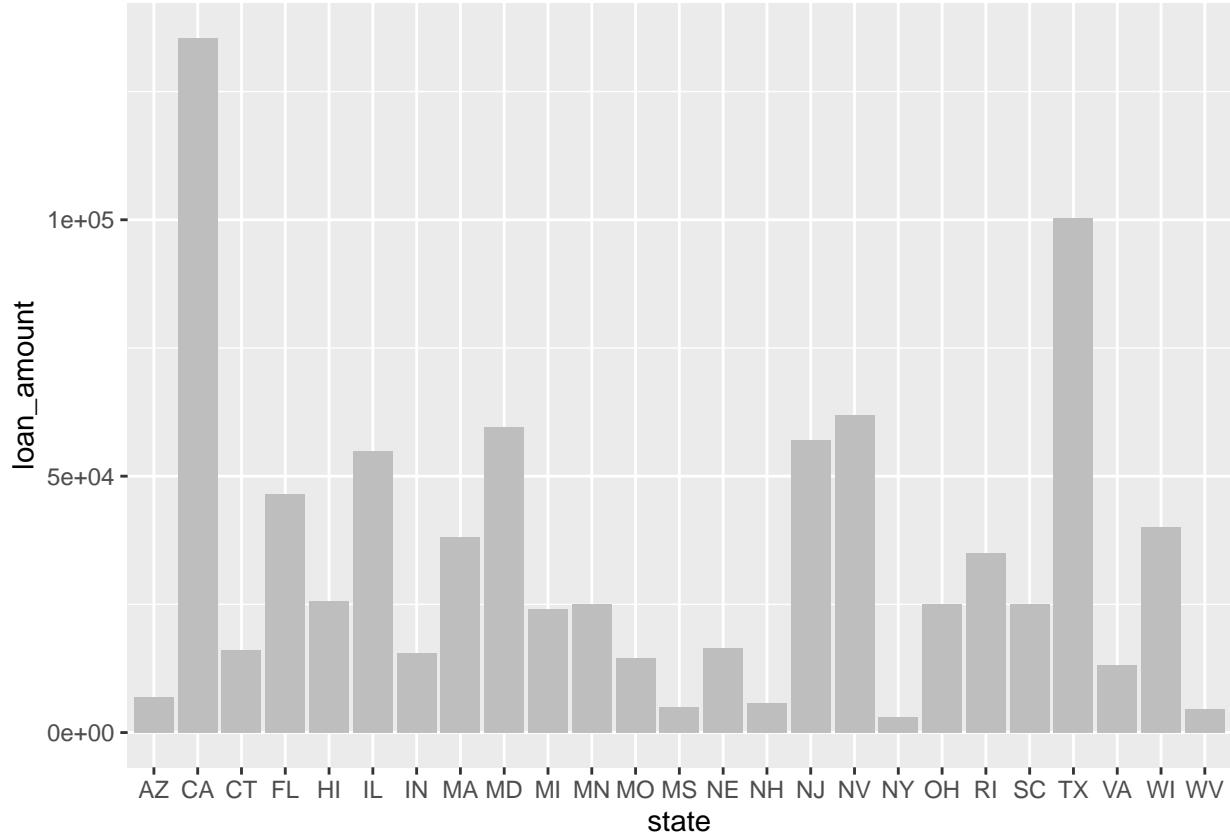
```
g + geom_point(size = 3) + facet_grid(loan_status~homeownership)
```



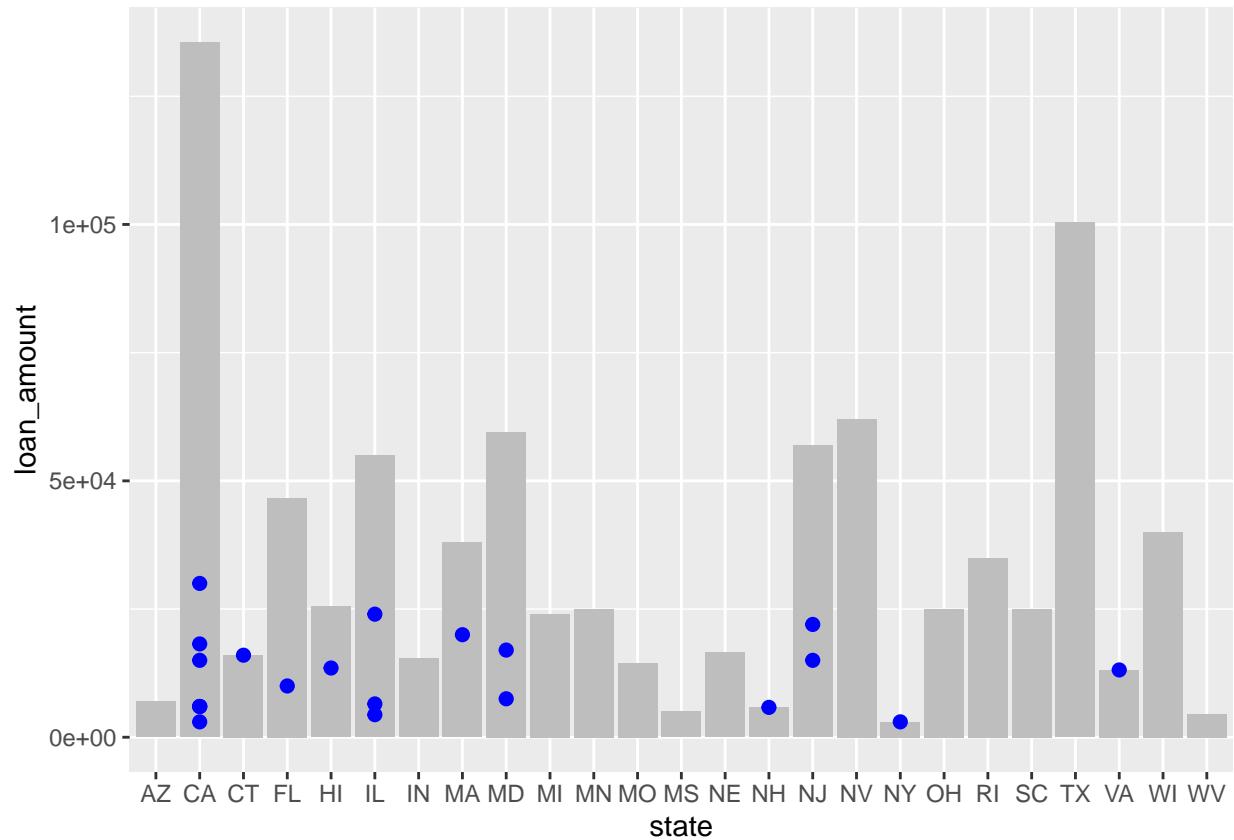
Layers

Layers are new graphs on top of an existing data graphic. You may use ggplots() to graph data from more than one data table on top of each other.

```
data2<-myData %>% filter(homeownership=="rent") %>% select(state,loan_amount)
g = ggplot(
  data = myData,
  aes(x = state , y = loan_amount)) +
  geom_col(fill = "gray")
g
```



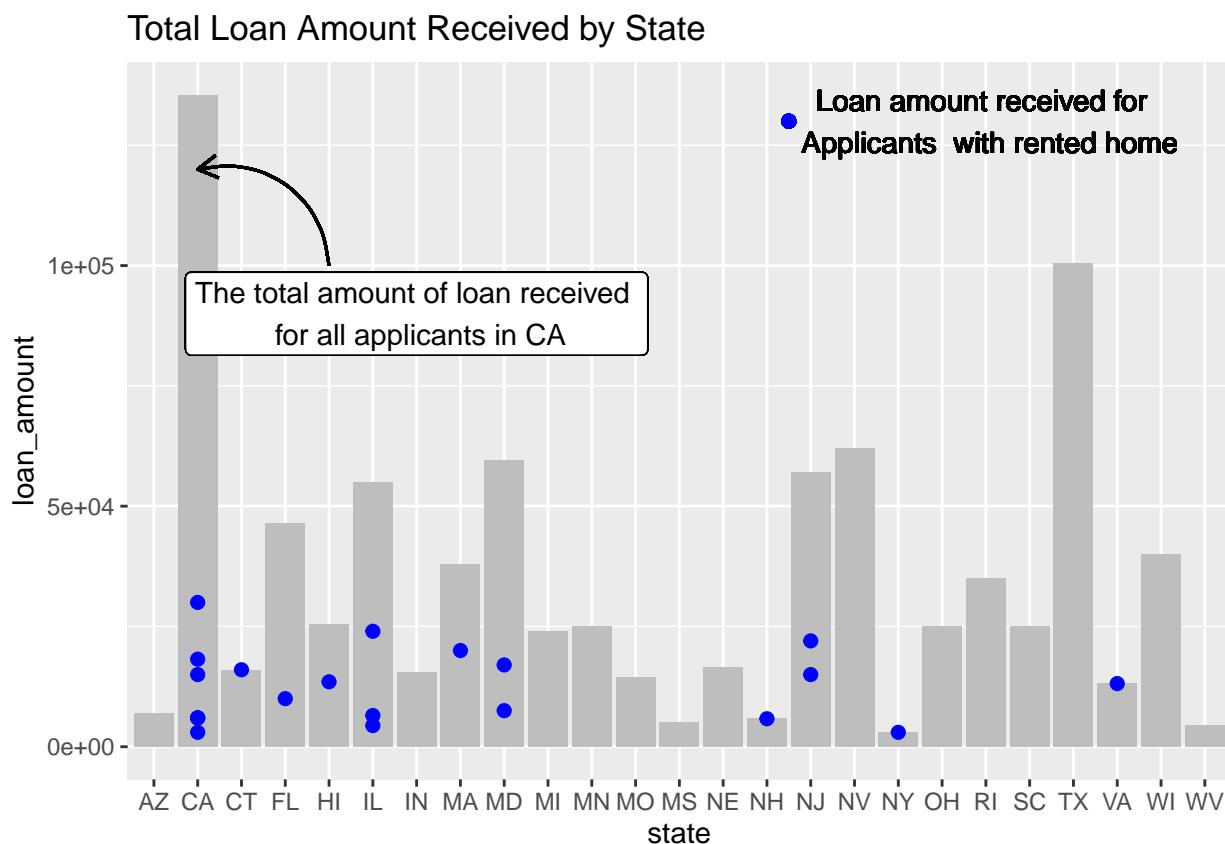
```
p=g+geom_point(data=data2, size = 2, color="blue")  
p
```



Guides (legends)

The geom_text() and geom_label() functions can also be used to provide specific textual annotations on the plot.

```
p+geom_point(  
  aes(y = 130000, x = 15.5),  
  color = "blue",  
  size = 2,  
) +  
  geom_text(aes(y=130000,x = 20,  
                label = "Loan amount received for \n Applicants with rented home")) +  
  ggtitle("Total Loan Amount Received by State") +  
  geom_curve(  
    x = 5, xend = 2, y = 100000, yend = 120000,  
    arrow = arrow(length = unit(0.3, "cm")), curvature = 0.5  
) +  
  geom_label(y=90000,x = 7,  
             label = "The total amount of loan received \n for all applicants in CA")
```



STAT560 - Foundation of Data Science - Lecture 8

Creating Standard Data Graphics

Over time, statisticians have developed standard data graphics for specific use cases. We've introduced the mechanics of those commonly used plots, such as histogram, barplot, boxplot, scatterplot in lecture 5 and 6. In this lecture, we will see how we can use R to generate those standard plots.

The standard plots we are going to create in this lecture are:

Pie Chat, Bar Graph, Dot Plot, Histogram, Boxplot, Scatterplot, Line Graph, Mosaic plot

You may create most of these plots using just the base package in R, or through the functions of ggplot2.

We again use loan50 data for demonstration.

```
myData=read.csv(file="https://www.openintro.org/data/csv/loan50.csv",header=T)
```

Pie Chat (univariate, categorical variable)

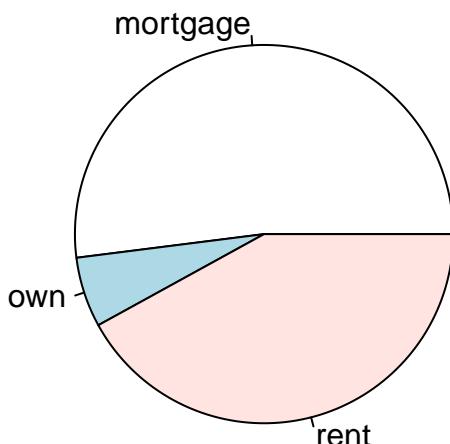
Pie chat can be used to display the distribution of a categorical variable. You may use pie() function in base R package to create a pie chat. You will need to generate a table to compute the counts or proportions first.

```
ho_count=table(myData$homeownership)
ho_count

## 
## mortgage      own       rent
##      26         3        21

pie(ho_count,main="Pie chat for categories of homeownership")
```

Pie chat for categories of homeownership

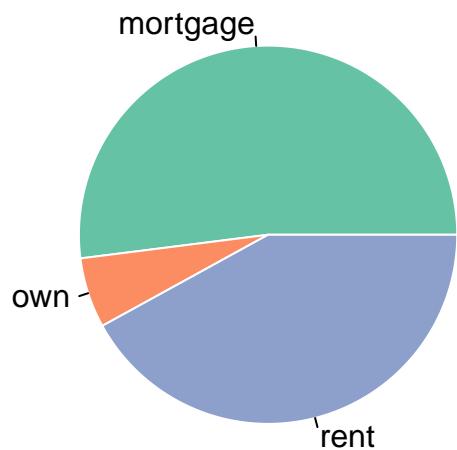


You may change label context with label, group color with col, and border color with border. Here, the RcolorBrewer package is used to build a nice color palette.

```

library(RColorBrewer)
myColor <- brewer.pal(3, "Set2")
pie(ho_count, border="white", col=myColor)

```



You may also use ggplot2 to with polar coordinates to create a pie chat.

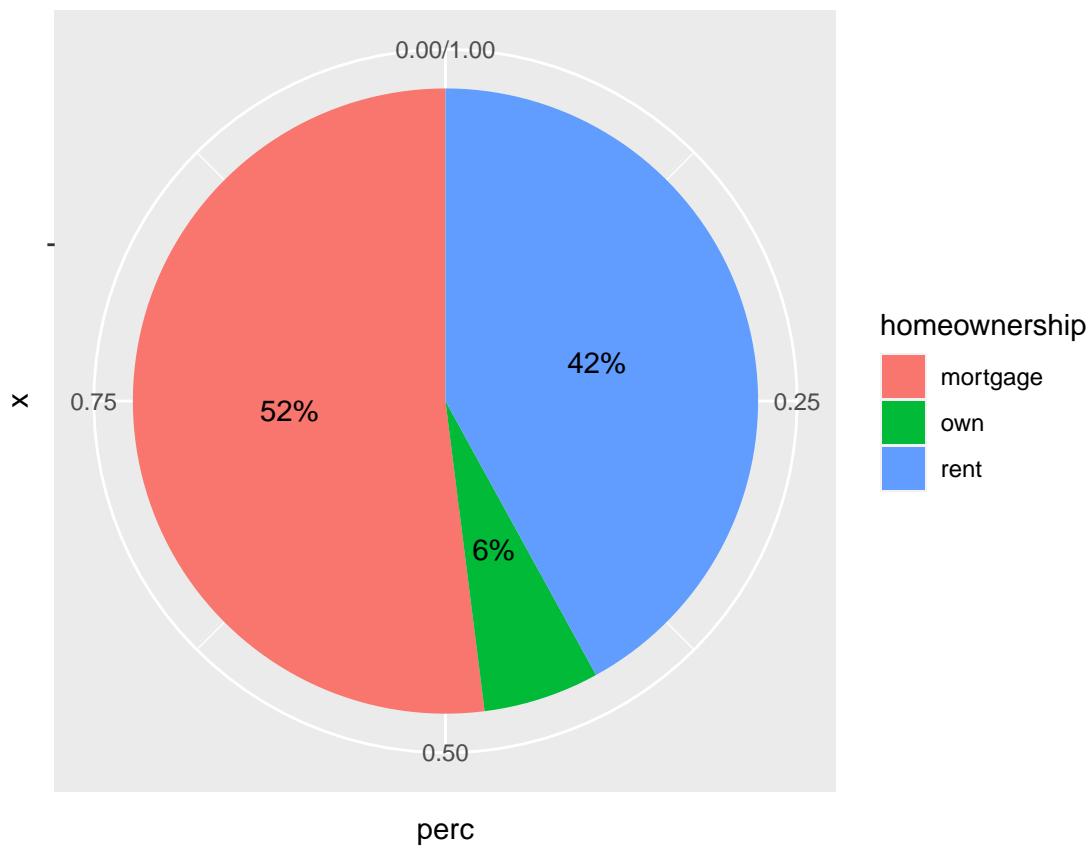
```

library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr    1.3.0
## v purrr    1.0.1
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
df = myData %>%
  group_by(homeownership) %>%
  summarise(n = n()) %>%
  mutate(perc = n/ sum(n)) %>%
  mutate(labels = scales::percent(perc))

g = ggplot(data=df, aes(x="",y=perc,fill=homeownership))
g + geom_col() + coord_polar(theta = "y") + geom_text(aes(label = labels),
  position = position_stack(vjust = 0.5))

```

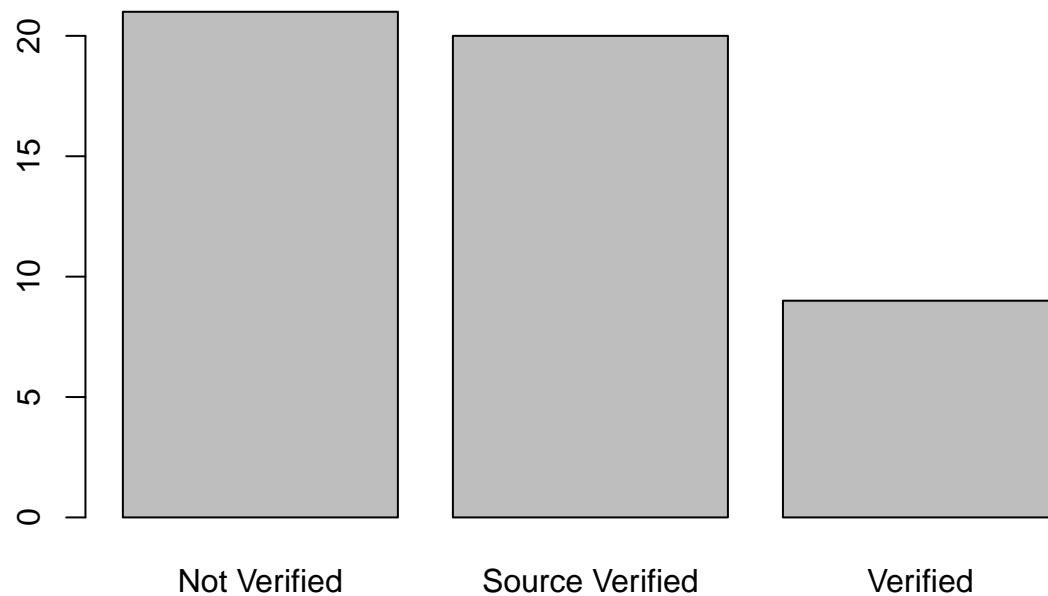


Bar Graph (categorical variables)

Bar graph can also be used display the distribution of a categorical variable. It is often preferred than pie chat. Bar graph is more effective in presenting information and can be used as stacked or side-by-side for comparison among groups.

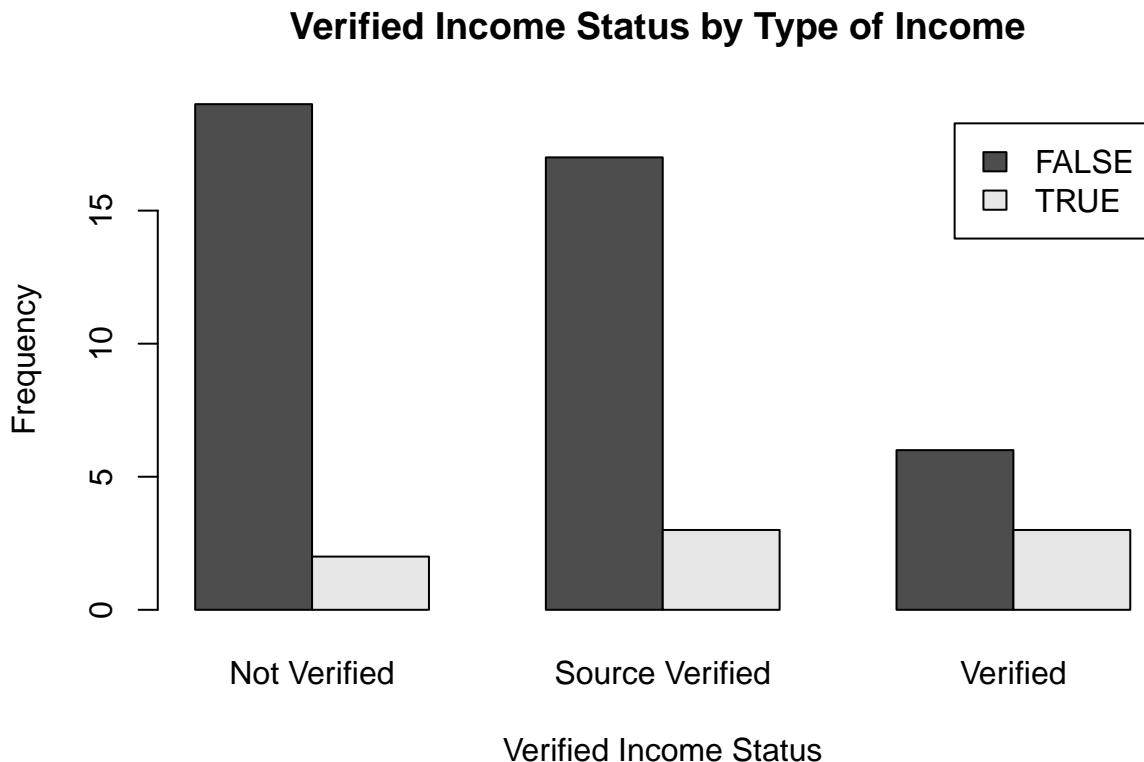
You may use barplot() function in base R package to create a bar graph. You will again need to generate a table to compute the counts or proportions first.

```
barplot(table(myData$verified_income))
```



You may also create side-by-side bar plot.

```
barplot(table(myData$has_second_income, myData$verified_income),
       beside = T,
       legend.text = T,
       xlab = "Verified Income Status",
       ylab = "Frequency",
       main = "Verified Income Status by Type of Income")
```

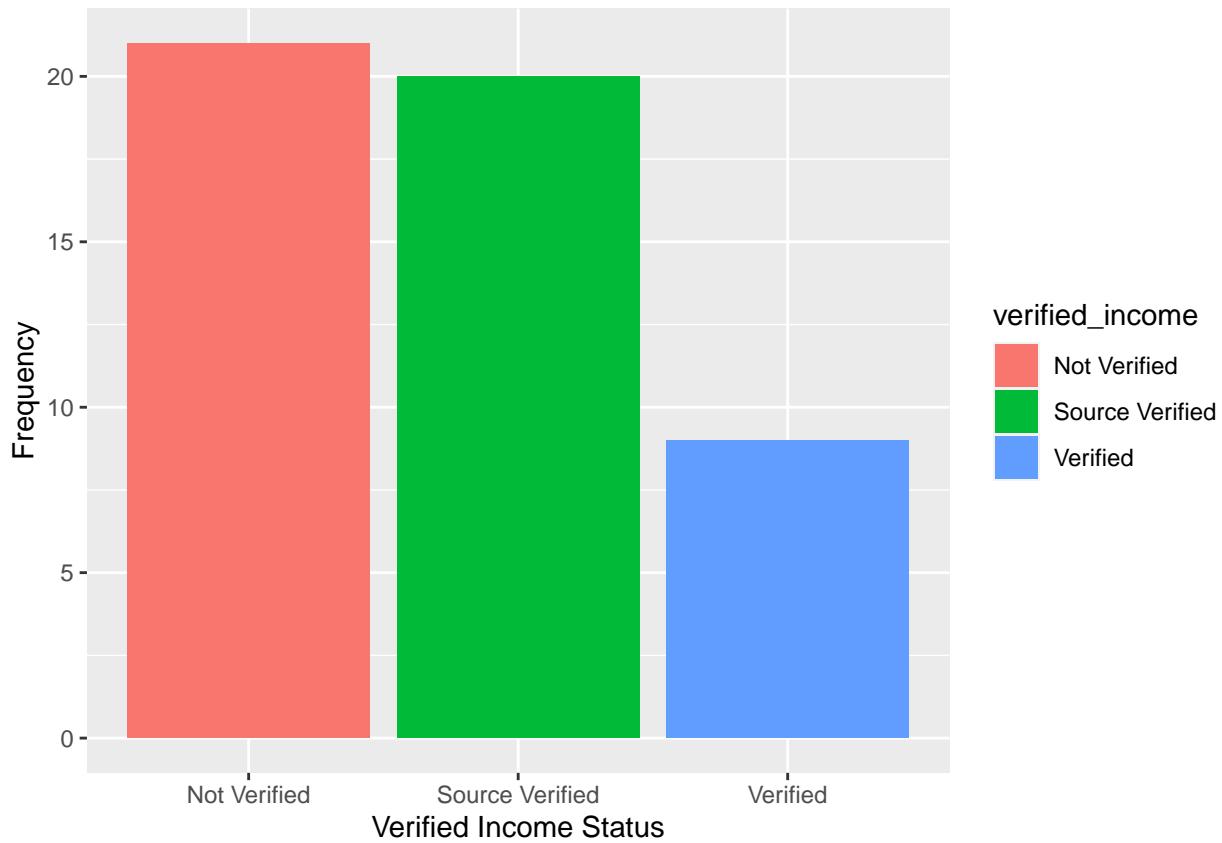


You may use geom_bar or geom_col in ggplot2 to create bar plots.

```

g = ggplot(data=myData,aes(x=verified_income))
g+ geom_bar(aes(fill=verified_income)) +
  labs(y = "Frequency", x= "Verified Income Status")

```

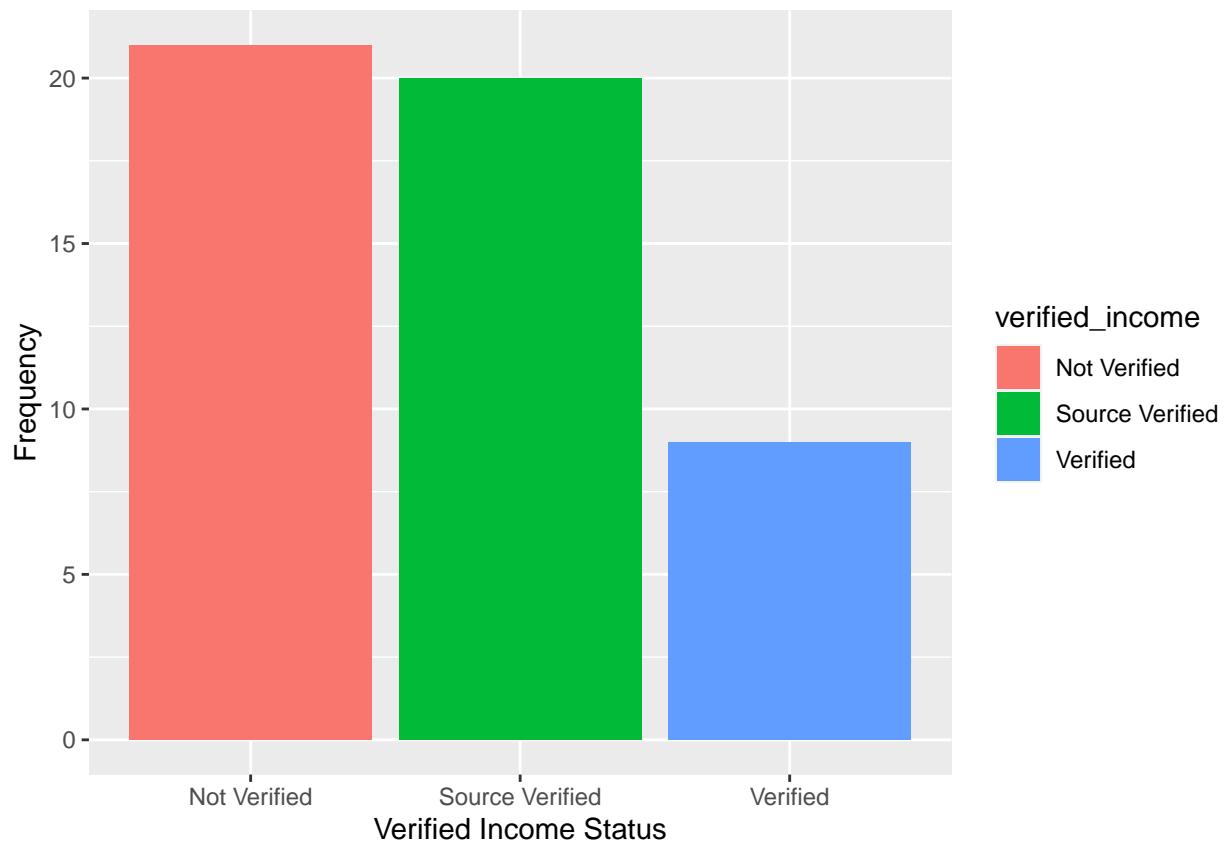


To use geom_col, you will need to first create a data frame to compute the counts or proportions in each category.

```

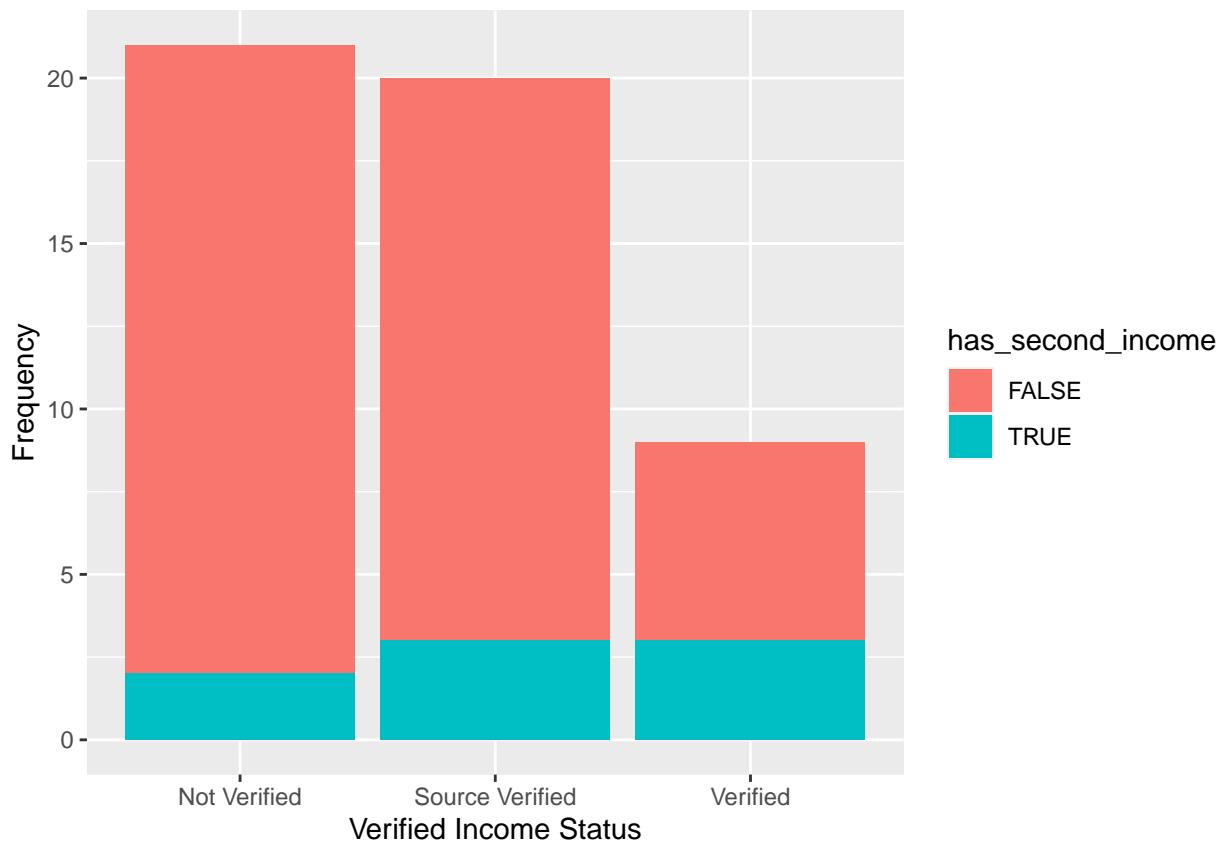
df = myData %>%
  group_by(verified_income) %>%
  summarise(n = n()) %>%
  mutate(perc = n/ sum(n))
ggplot(data=df,aes(x=verified_income, y=n))+ 
  geom_col(aes(fill=verified_income))+ 
  labs(y = "Frequency", x= "Verified Income Status")

```

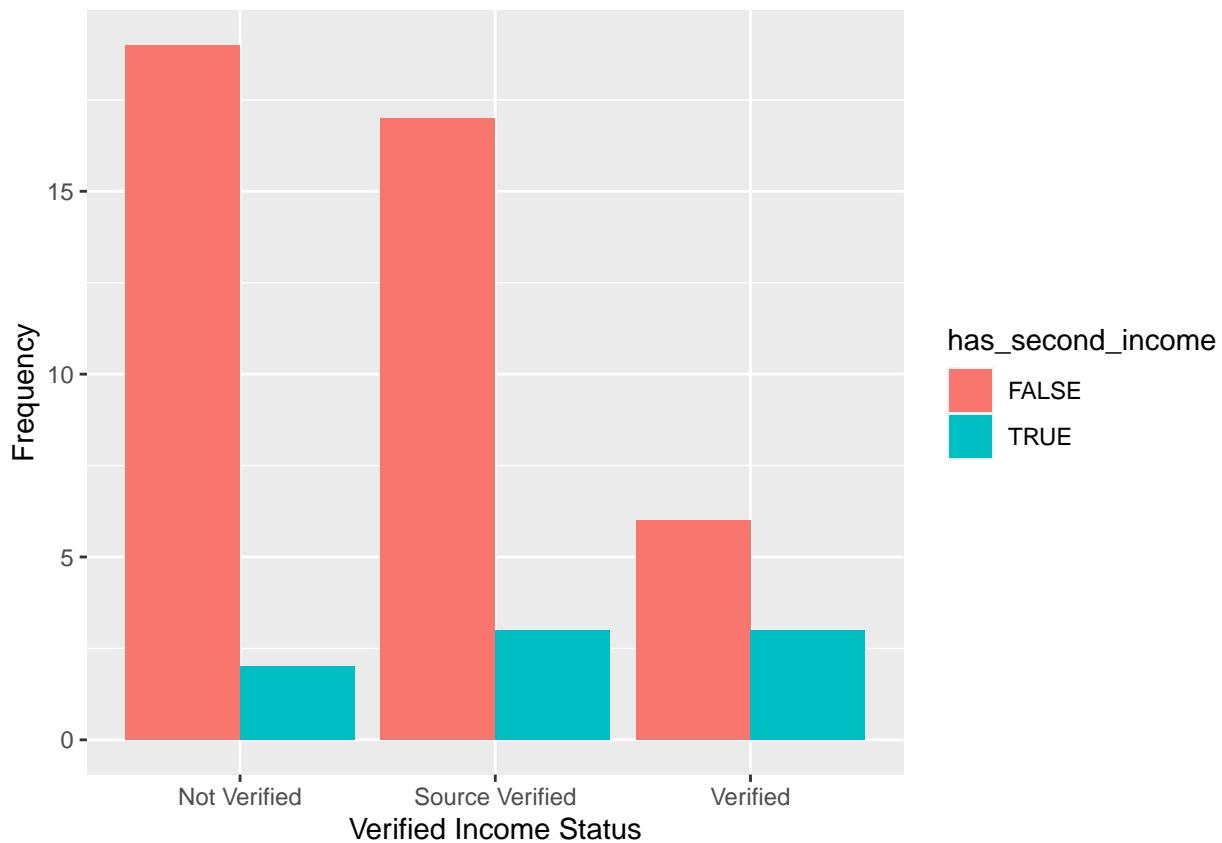


You can also create side-by-side or stacked bar plots using ggplot2.

```
#Stacked
g = ggplot(data=myData,aes(x=verified_income,fill=has_second_income))
g+ geom_bar() +labs(y = "Frequency", x= "Verified Income Status")
```



```
#side-by-side
g = ggplot(data=myData,aes(x=verified_income,fill=has_second_income))
g+ geom_bar(position = "dodge") +
  labs(y = "Frequency", x= "Verified Income Status")
```

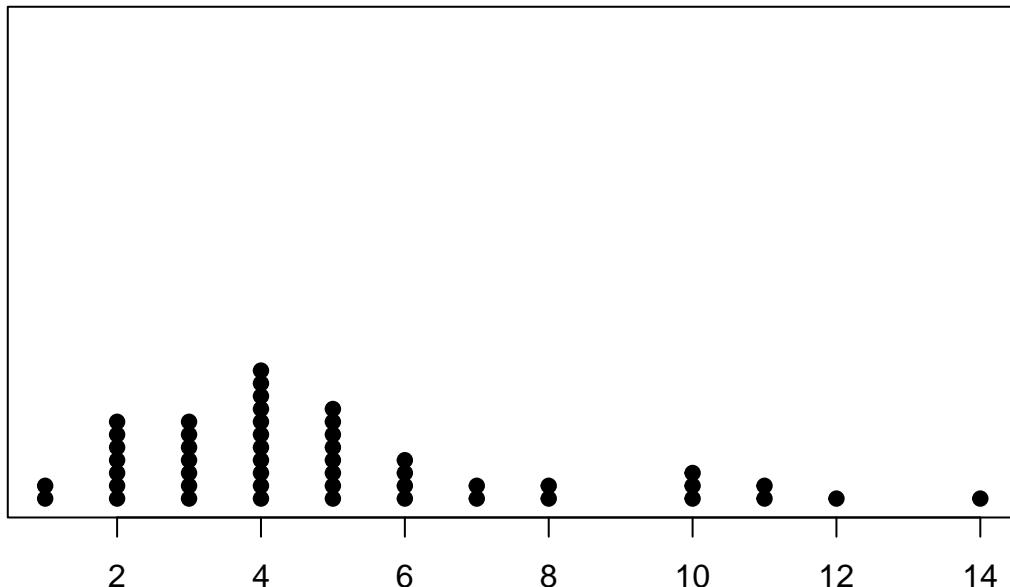


Dot Plot (univariate, numerical variable in small samples)

Dot plot can be used display the distribution of a numerical variable. Dots are stacked, with each dot representing one observation. The graph is useful in small samples to identify possible outliers.

You may create a dotplot with `stripchart()` function in base R package, or `geom_dotplot()` in `ggplot2`.

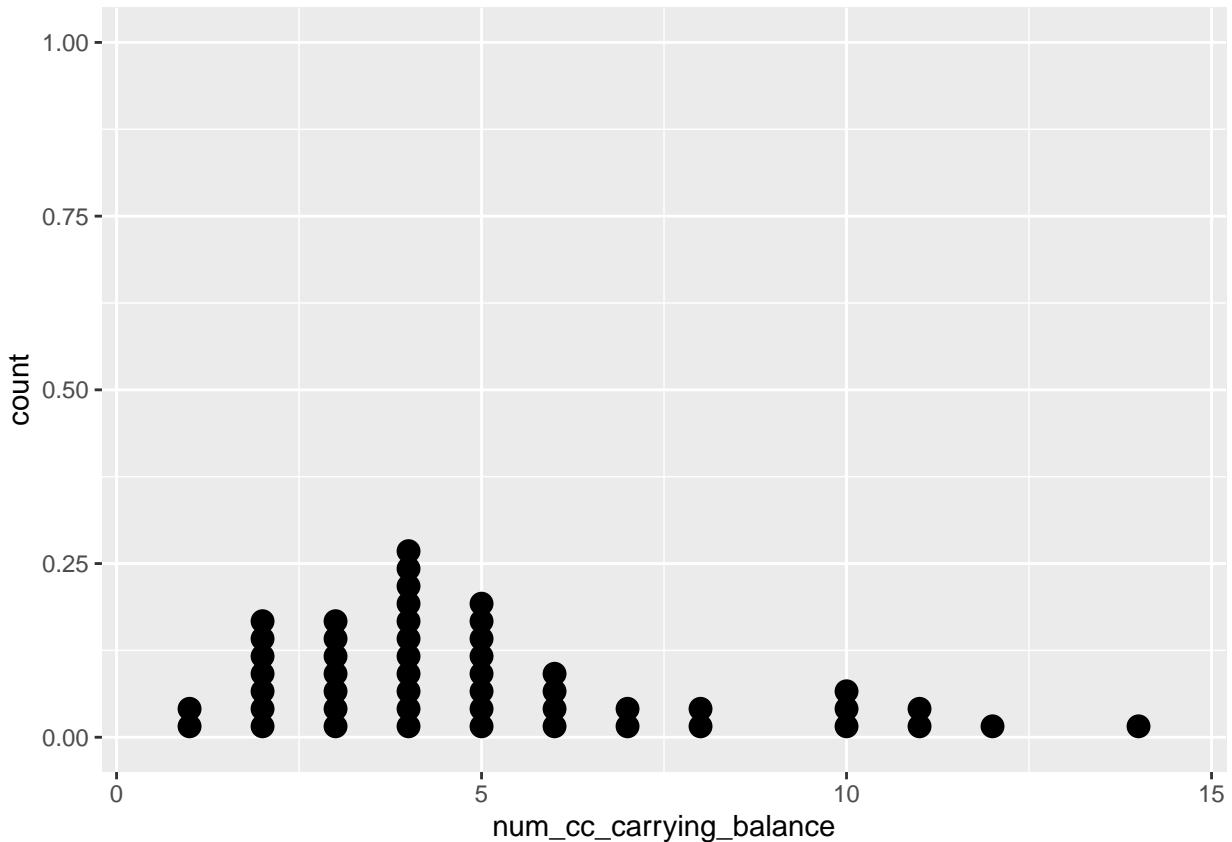
```
stripchart(myData$num_cc_carrying_balance, method = "stack", at = 0, pch = 19,
           col = "black")
```



```

g = ggplot(data=myData,aes(x=num_cc_carrying_balance))
g+geom_dotplot(fill = "black",dotsize = 0.3, stackratio = 0.8,binwidth=1)

```



Histogram (univariate, numerical variable suitable for big data)

A histogram is appropriate to summary the distribution of a numerical variable. It groups data values into bins and then plot the counts. Histogram works very well with large datasets.

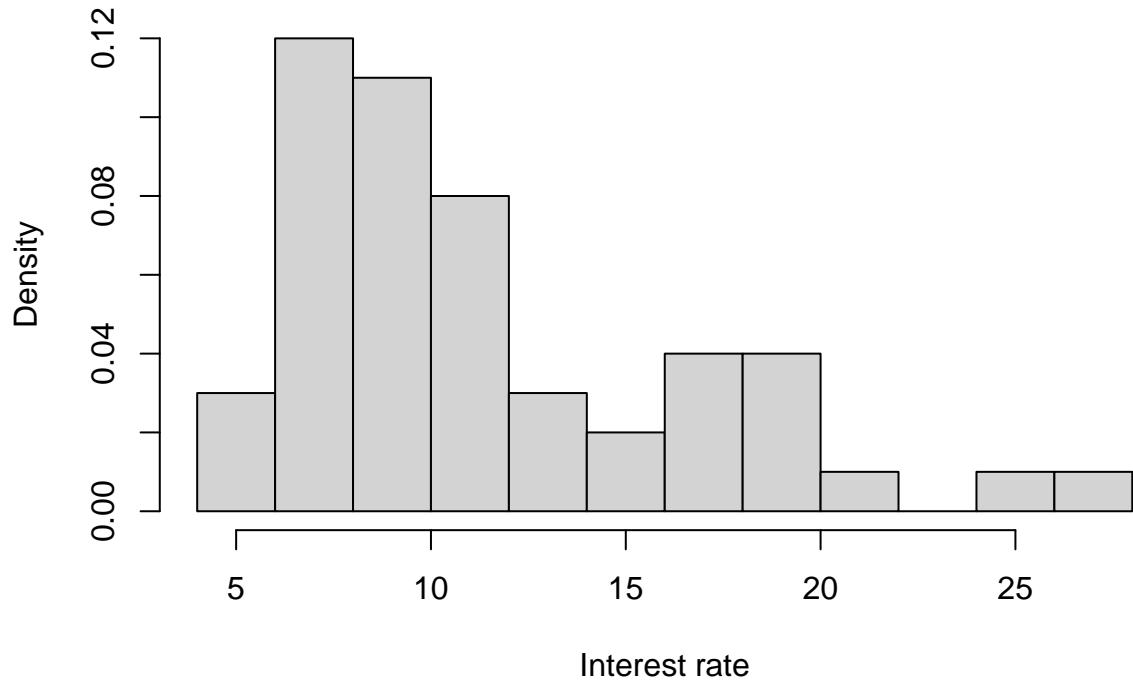
You may construct histogram with the `hist()` function in R base package.

```

hist(myData$interest_rate, prob=T,
      xlab = "Interest rate",
      main = "Probability histogram of interest rate", breaks=10)

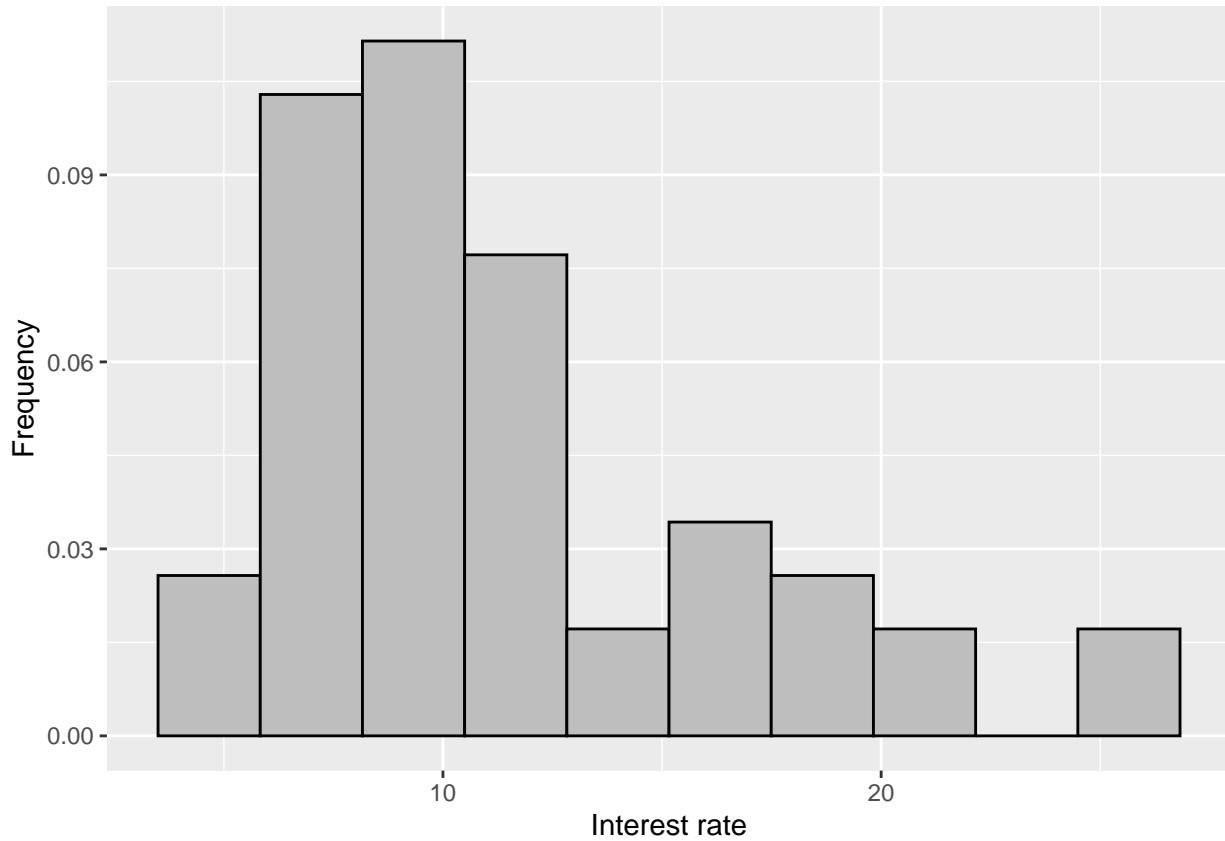
```

Probability histogram of interest rate



You may also construct histogram with the `geom_hist()` in `ggplot2` package.

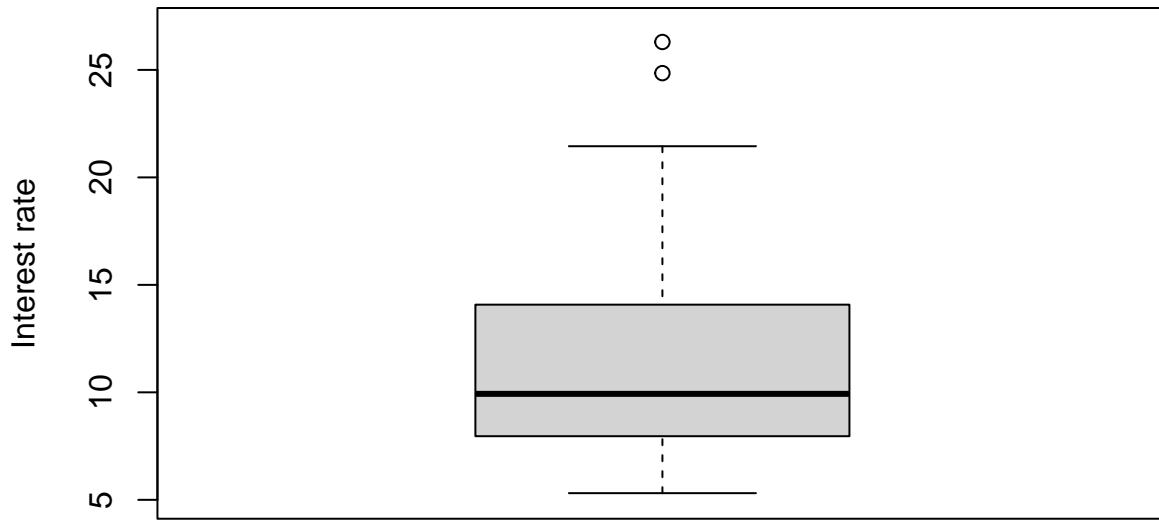
```
ggplot(data=myData,aes(x = interest_rate)) +  
  geom_histogram(aes(y = after_stat(density)),bins=10, color="black",fill="grey") +  
  labs(y = "Frequency", x = "Interest rate")
```



Boxplot (univariate numerical variable or compare numerical variable across categorical levels)

A boxplot is appropriate to summary the distribution of a numerical variable. It presents the five-number-summary of the data in a plot. You may create boxplot use boxplot() command in R base package or the geom_boxplot() function in ggplot2.

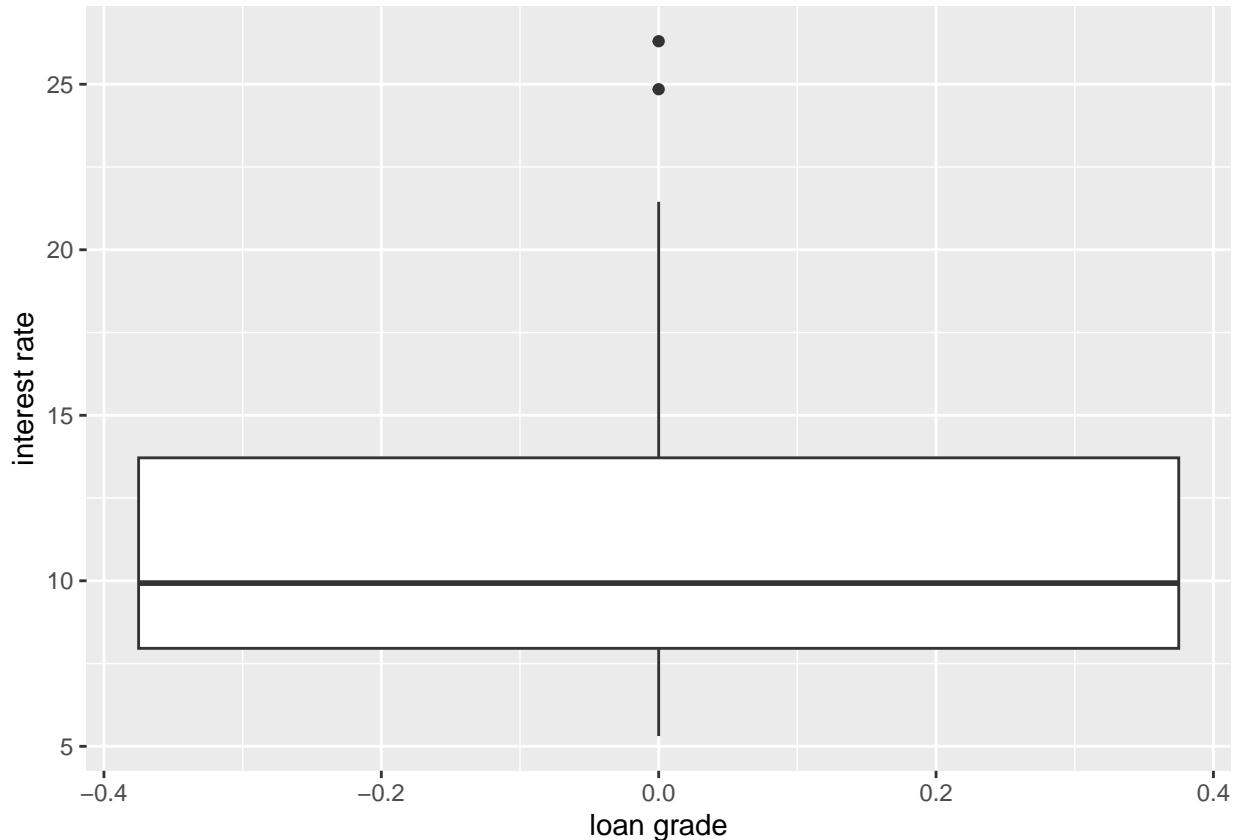
```
boxplot(myData$interest_rate, ylim=c(5,27), ylab="Interest rate")
```



```

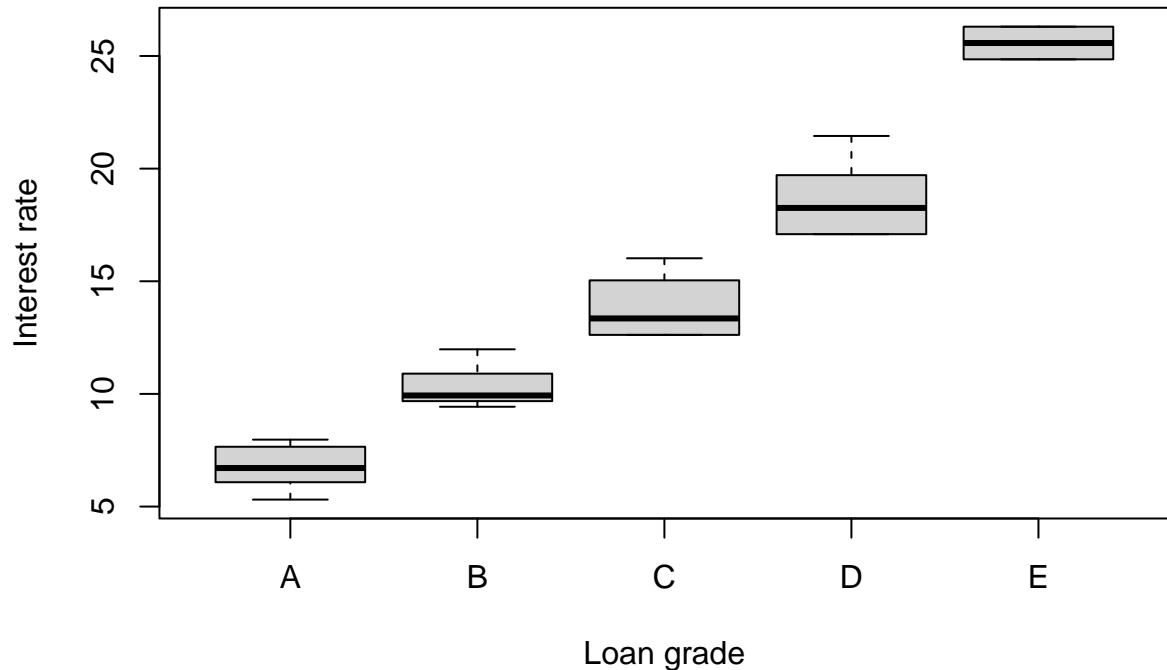
quantile(myData$interest_rate) #five number summary used for the boxplot
##      0%     25%     50%     75%    100%
## 5.310  7.960  9.930 13.715 26.300
ggplot(data=myData,aes(y = interest_rate)) +
  geom_boxplot() +
  labs(y = "interest rate", x = "loan grade")

```

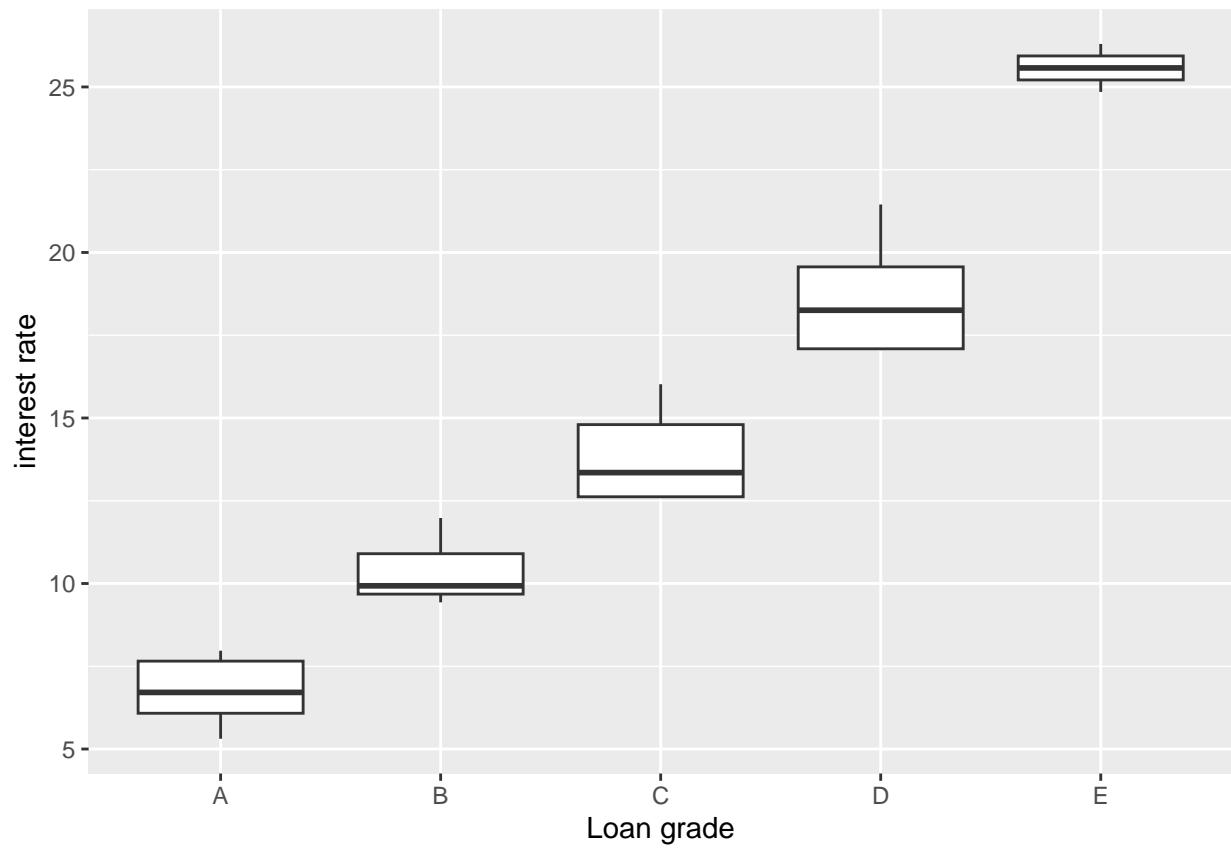


Side by side boxplot are also commonly used to compare a numerical variable in different groups of a categorical variable.

```
boxplot(myData$interest_rate~myData$grade,ylab="Interest rate",xlab="Loan grade")
```



```
ggplot(data=myData,aes(y = interest_rate,x=grade)) +
  geom_boxplot() +
  labs(y = "interest rate",x = "Loan grade")
```

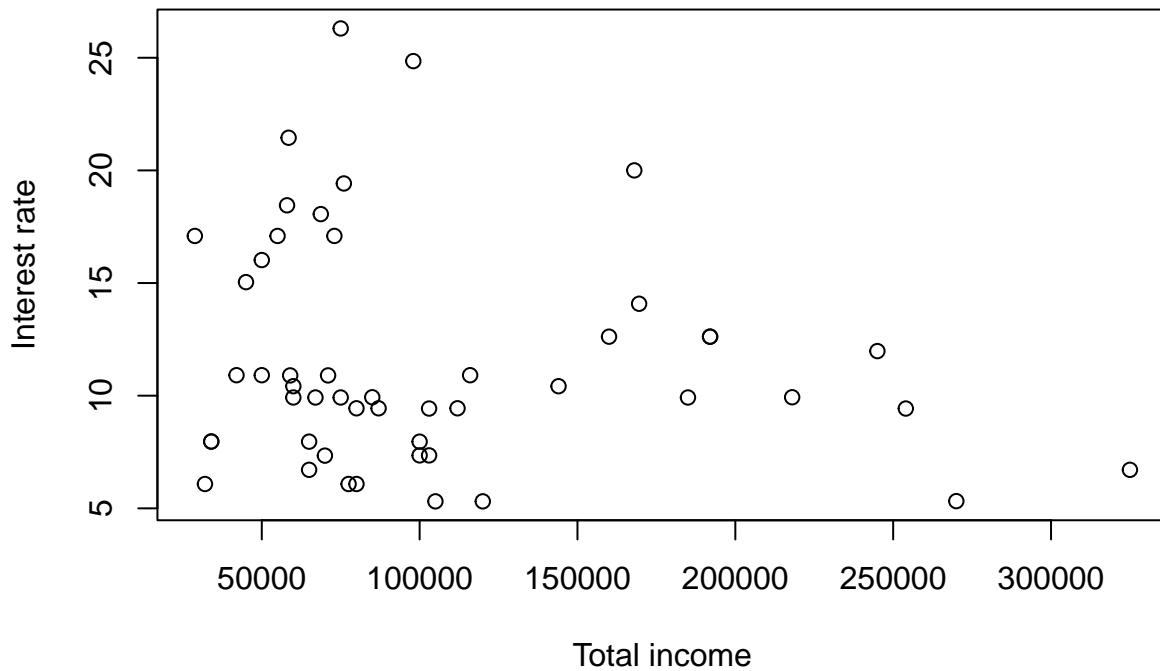


Scatterplot (bivariate numerical variable)

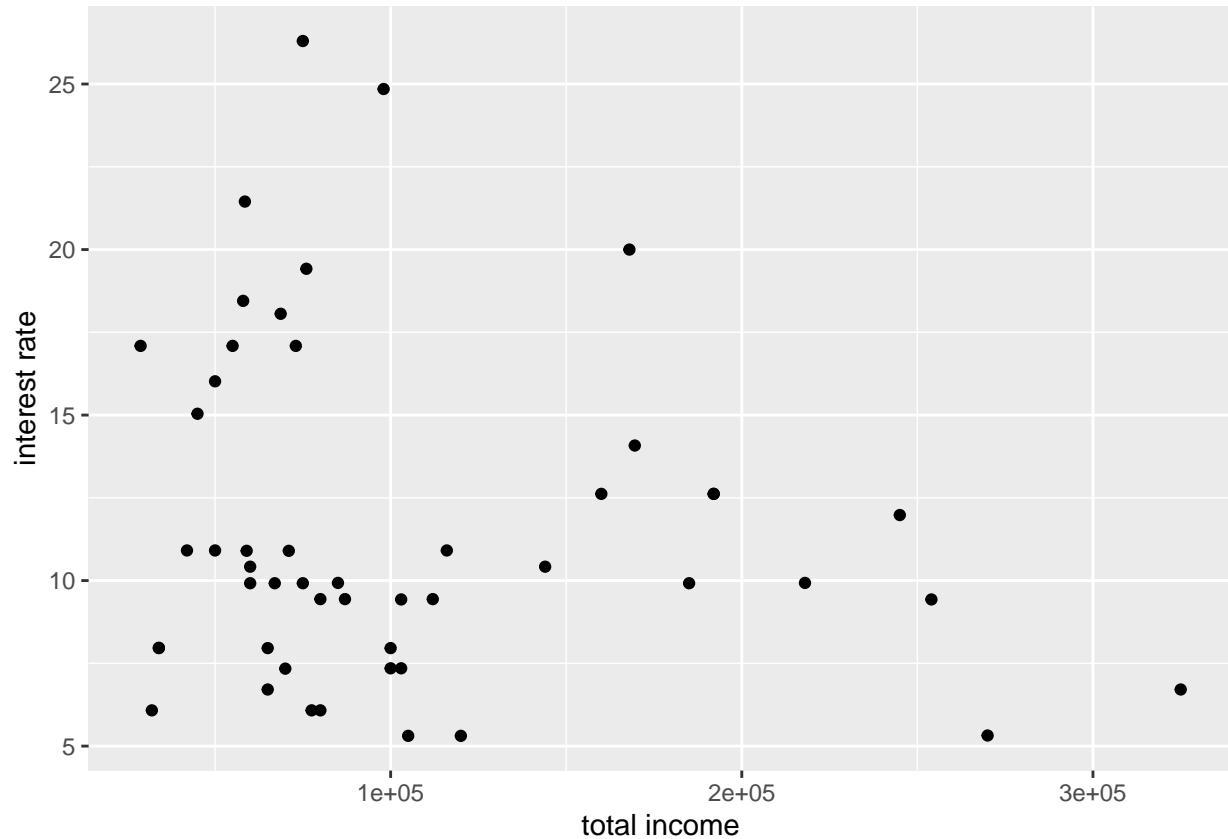
Scatterplots are appropriate for examining the relationship between 2 numerical variables. The main purpose of a scatterplot is to show the relationship between two variables across cases. Often, there is a Cartesian coordinate system in which the x-axis represents explanatory (independent) variable and the y-axis the value of response (dependent) variable.

You may construct scatterplots using plot() in R base package or geom_point() in ggplot2.

```
plot(myData$interest_rate~myData$total_income,ylab="Interest rate",xlab="Total income")
```

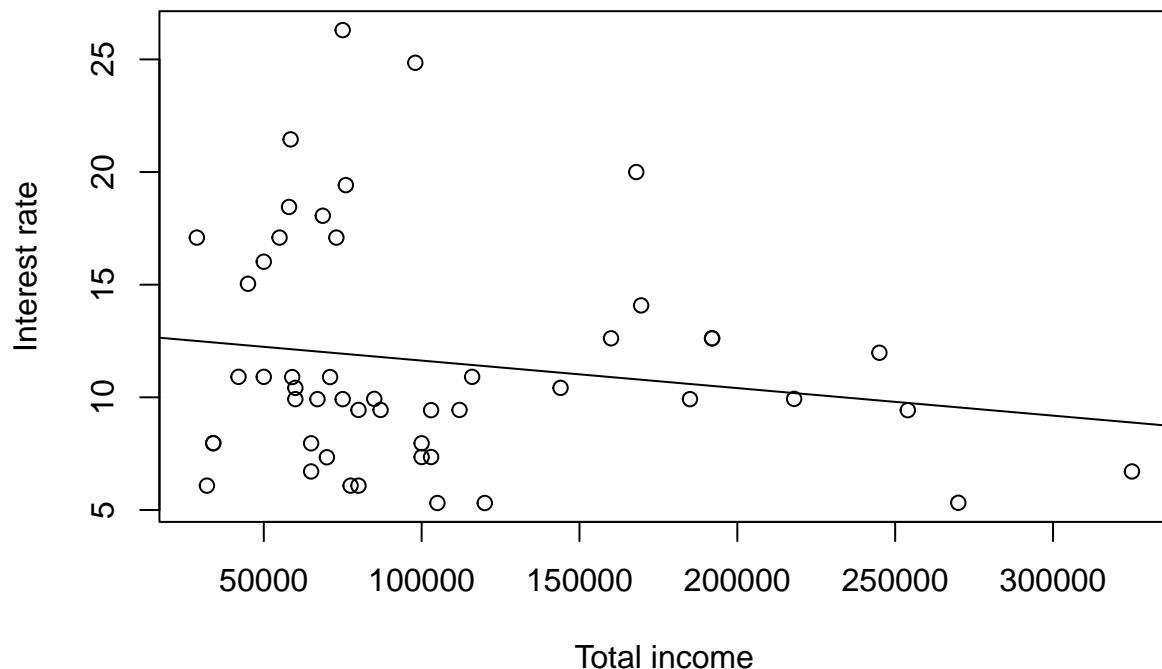


```
ggplot(data=myData,aes(y = interest_rate,x = total_income)) +  
  geom_point() +  
  labs(y = "interest rate",x="total income")
```



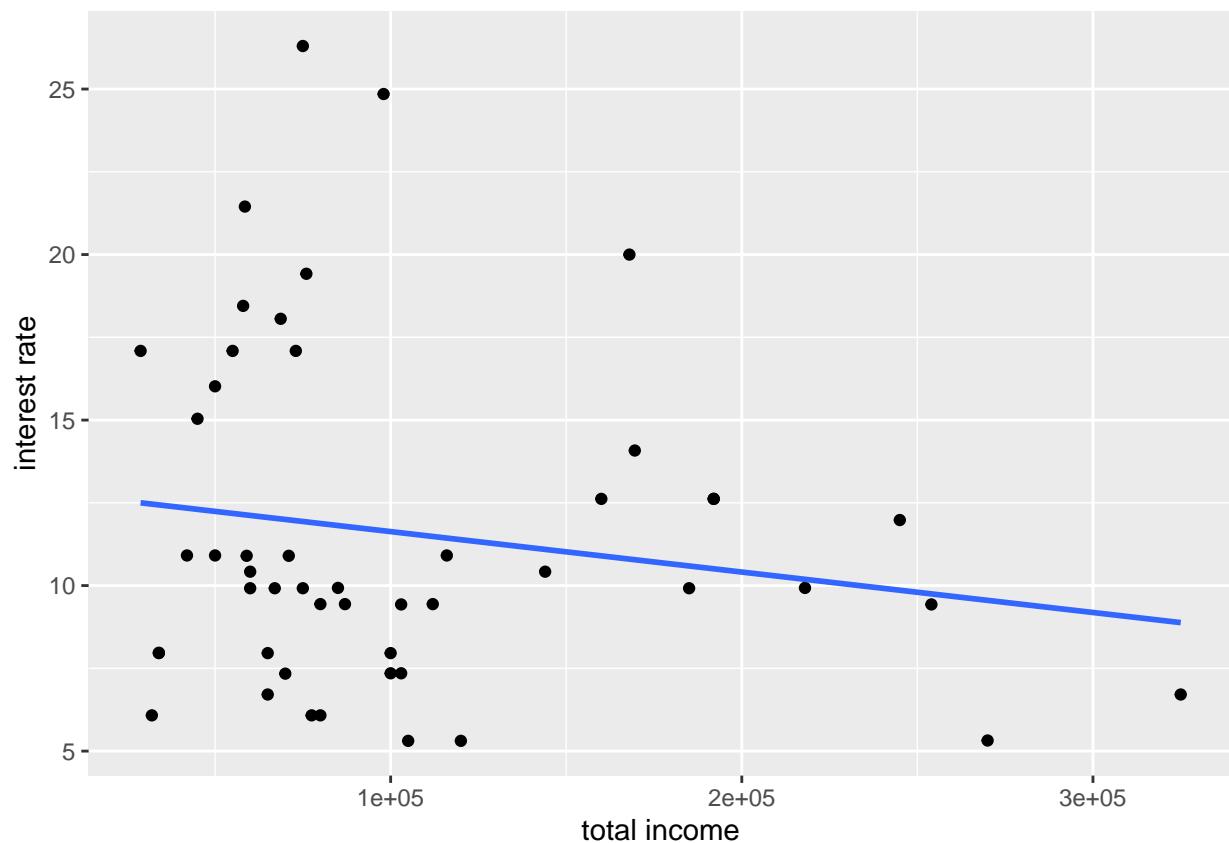
We can add a smooth trend line through the points. We use abline() in base R package or the geom_smooth() function in ggplot2 order to plot the simple linear regression line along with the points.

```
plot(myData$total_income,myData$interest_rate,ylab="Interest rate",xlab="Total income")
abline(lm(myData$interest_rate~myData$total_income))
```



```
ggplot(data=myData,aes(y = interest_rate,x = total_income)) +
  geom_point() + geom_smooth(method = "lm", se=F) +
  labs(y = "interest rate",x="total income")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Line Graph (Time series data)

A time series (line) graph is just a scatterplot with time on the horizontal axis and points connected by lines to indicate temporal continuity. You may construct a line graph just the same way as a scatterplot by plotting time in the x-axis.

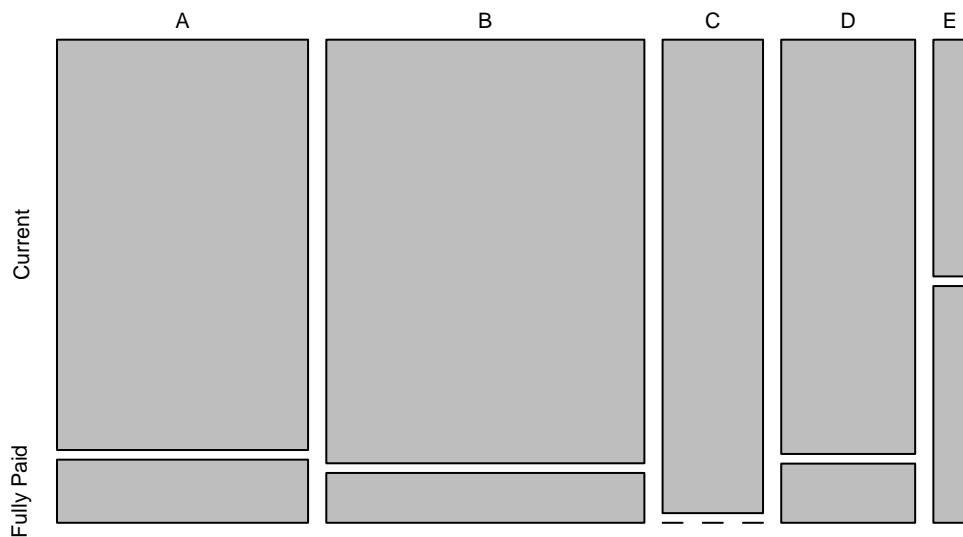
Mosaic plot

When both the explanatory and response variables are categorical, we present the relationship with a mosaic plot. In the mosaic plot the number of observations in each cell is proportional to the area of the box.

You may construct a mosaic plot with `mosaicplot()` in R base package or `geom_mosaic()` in `ggmosaics` package.

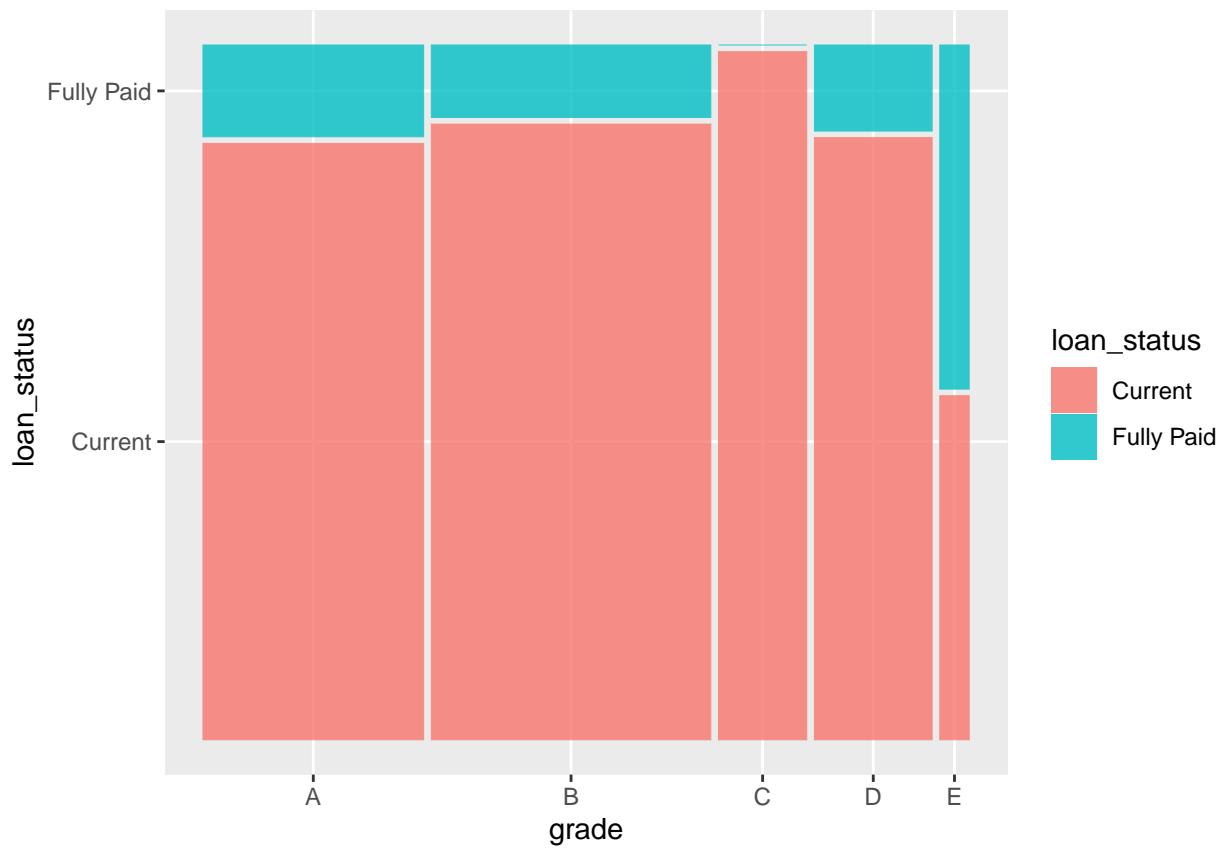
```
mosaicplot(table(myData$grade,myData$loan_status))
```

table(myData\$grade, myData\$loan_status)



```
library(ggmosaic)
ggplot(data=myData) +
  geom_mosaic(aes(x=product(grade), fill=loan_status))

## Warning: `unite_()` was deprecated in tidyverse 1.2.0.
## i Please use `unite()` instead.
## i The deprecated feature was likely used in the ggmosaic package.
##   Please report the issue at <https://github.com/haleyjeppson/ggmosaic>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



STAT 560 Lecture 9: Probability

Beidi Qiang

SIUE

Overview of Probability

The subject of probability theory is the foundation upon which all of statistics is built, providing a means for modeling populations, experiments, or almost anything else that could be considered a random phenomenon.

Definition: The probability of an outcome is the proportion of times the outcome would occur if we observed the random process an infinite number of times.

- ▶ A probability is a number between 0 (or 0%) and 1 (or 100%).
- ▶ The higher the probability for a particular outcome, the more likely it is to occur.
- ▶ The probability of an outcome refers to what happens in the long run. As more observations are collected, the proportion of occurrences with a particular outcome converges to the probability of that outcome. (Law of Large Numbers).

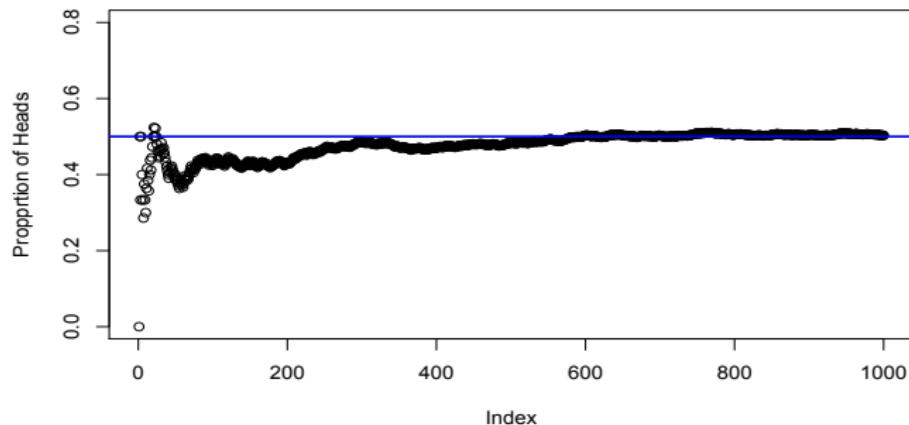
Example: Coin Toss

Consider a coin toss.

- ▶ There are 2 possible outcomes in a coin toss (H, T) and both outcomes have the same chance of occurring (a **fair** coin toss).
- ▶ We say the probability of coming up "heads" is 0.5. We use notation $P(H) = 0.5$
- ▶ This means if we toss the coin many times, the proportion of times it comes up heads becomes close to 0.5.
- ▶ Note: this probability is 0.5 not simply because the coin has two sides and one must turn up!

Coin Toss Simulation

The figure below shows the simulation results of 1000 coin tosses. On the vertical axis is the proportion of heads, charted over the number of tosses.



R code to generate this simulation:

```
coin=c("H", "T")
toss1000=sample(coin,1000,replace=T)
proportion=vector(length=1000)
for (i in 1:1000) {proportion[i]=sum(toss1000[1:i]=="H")/i}
plot(1:1000, proportion); abline(h=0.5)
```

The set of *all possible outcomes* for a given random experiment is called the **sample space**, denoted by S . (discrete vs. continuous)

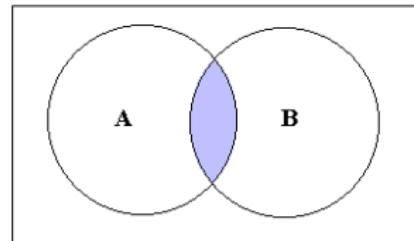
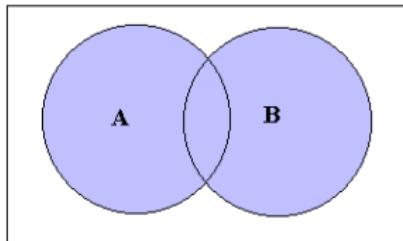
- ▶ Tossing a coin once. $S = \{H, T\}$.
- ▶ Tossing a coin three times.
 $S = \{HHH, HHT, HTH, HTT, THH, THT, TTH, TTT\}$
- ▶ Observing the lifetime of an electronic component.
 $S = \{\text{any number in } (0, +\infty)\}$

The element of the sample space is called **outcome**.

- ▶ An **event** is a set of possible outcomes that is of interest. We use capital letters, such as A, B, C to denote an event.
- ▶ We would like to develop a mathematical framework so that we can assign probability to an event A . This will quantify how likely the event is. The probability that the event A occurs is denoted by $P(A)$.
- ▶ Example:
Random experiment: toss a coin once;
Event: $H=$ toss a coin once and the head is up;
Probability: $P(H)=0.5$.

Union and Intersection of Events

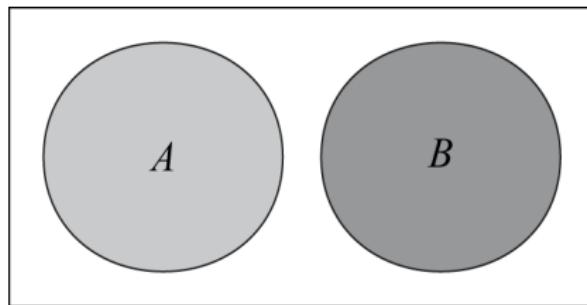
- ▶ The **union** of two events A and B is the set of all outcomes in either event or both. We use $P(A \text{ or } B)$ to denote the probability of the union.
- ▶ The **intersection** of two events A and B is the set of all outcomes in the both events. We use $P(A \text{ and } B)$ to denote the probability of the intersection.



Mutually Exclusive Events

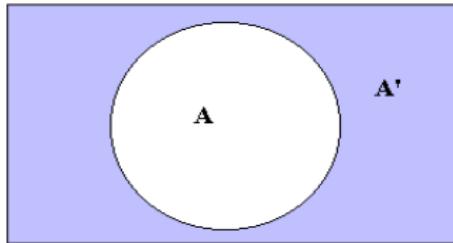
- ▶ **Mutually exclusive** events **can not** occur at the same time. If events A and B are mutually exclusive, then they do not share any common outcomes.
- ▶ If there are many disjoint outcomes A_1, \dots, A_k , then the probability that one of these outcomes will occur is

$$P(A_1 \text{ or } A_2 \text{ or } \dots \text{ or } A_k) = P(A_1) + P(A_2) + \dots + P(A_k)$$



- ▶ **Definition:** The **complement** of an event is every outcome not included in the event, but still part of the sample space. The complement of A is denoted by A^c or A' .
- ▶ Relationship between the probability of an event and its complement:

$$P(A) + P(A^c) = 1$$



- ▶ A conditional probability is the probability that an event will occur, when another event is known to occur or to have occurred

- ▶ **Definition:** Let A and B be events in a sample space S with

 $P(B) > 0$. The **conditional probability** of A , given that B has occurred, is

$$P(A|B) = \frac{P(A \text{ and } B)}{P(B)}.$$

- ▶ **Multiplication rule:** Solving the conditional probability formula for the probability of the intersection of A and B yields

$$P(A \text{ and } B) = P(A|B) \times P(B).$$

Exploring probabilities with a contingency table

Data scientists have been working to improve a classifier for whether the photo is about fashion or not. Each photo gets two classifications: the first is called machine learned and gives a classification from a machine learning (ML) system of either predicted fashion or not. Each of these 1822 photos have also been classified carefully by a team of people, which we take to be the source of truth; this variable is called truth and takes values fashion and not.

		truth		Total
		fashion		
mach_learn	pred_fashion	197	22	219
	pred_not	112	1491	1603
	Total	309	1513	1822

- ▶ Marginal probabilities (probabilities based on a single variable):
 $P(\text{pred_fashion}) = 219/1822 = 0.12$
- ▶ Joint probabilities (probabilities of outcomes for two or more variables):
 $P(\text{pred_fashion and truth is fashion}) = 197/1822 = 0.11$
 $P(\text{pred_not, truth is fashion}) = 112/1822 = 0.06$
- ▶ Conditional probabilities (probabilities of outcomes for one variable under a condition of another variable):
 $P(\text{pred_fashion} \mid \text{truth is fashion}) = 197/309 = 0.638$

- ▶ **Definition:** When $P(A|B) = P(A)$, we say that events A and B are **independent**. If A and B are not independent, they are said to be **dependent** events.
- ▶ This definition is symmetric, $P(A|B) = P(A) \iff P(B|A) = P(B)$.
- ▶ Under independence assumption,
$$P(A \text{ and } B) = P(A|B)P(B) = P(B|A)P(A) = P(A)P(B)$$

Bayes' Theorem: Inverting Probabilities

In many instances, we are given a conditional probability of the form $P(A|B)$ but we would really like to know the inverted conditional probability $P(B|A)$. In these cases, we can apply Bayes' Theorem. To derive the Bayes' Theorem:

1. Multiplicative Rule:

$$P(A \text{ and } B) = P(A|B)P(B) = P(B|A)P(A).$$

2. Law of Total Probability:


$$P(A) = P(A \text{ and } B) + P(A \text{ and } B^c) = P(A|B)P(B) + P(A|B^c)P(B^c).$$

3. Bayes' rule:

$$P(B|A) = \frac{P(A \text{ and } B)}{P(A)} = \frac{P(A|B)P(B)}{P(A|B)P(B) + P(A|B^c)P(B^c)}.$$

In Canada, about **0.35% of women over 40 will develop breast cancer** in any given year. A common screening test for cancer is the mammogram, but this test is not perfect. In about **11% of patients with breast cancer, the test gives a false negative**: it indicates a woman does not have breast cancer when she does have breast cancer. Similarly, the test gives a **false positive in 7% of patients who do not have breast cancer**: it indicates these patients have breast cancer when they actually do not. **If we tested a random woman over 40 for breast cancer using a mammogram and the test came back positive (that is, the test suggested the patient has cancer) what is the probability that the patient actually has breast cancer?**

Breast Cancer Example Continued

Define the events:

$$A = \{\text{the patient is tested positive using a mammogram}\}$$

$$B = \{\text{the patient actually has breast cancer}\}$$

We want to calculate $P(B|A)$.

From the information in the problem, we have

$$P(B) = 0.35\%, P(A^c|B) = 11\%, \text{ and } P(A|B^c) = 7\%.$$

Use property of complement event:

$$P(B^c) = 1 - P(B) = 99.65\% \text{ and } P(A|B) = 1 - P(A^c|B) = 89\%$$

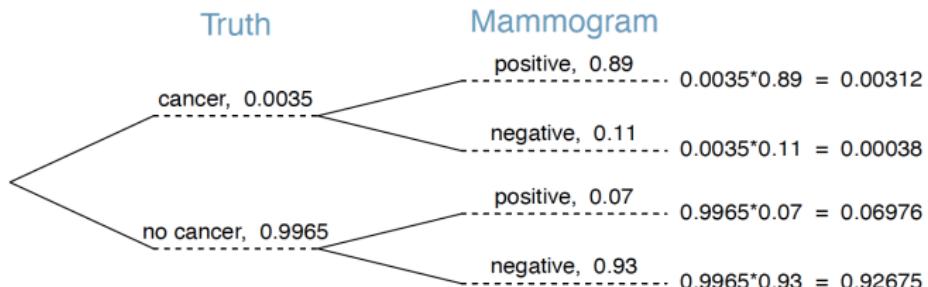
Use Bayes' Theorem:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A|B)P(B) + P(A|B^c)P(B^c)} = \frac{0.89 \times 0.0035}{0.89 \times 0.0035 + 0.07 \times 0.9965} = 0.0428.$$

Tree diagrams

Tree diagrams are a tool to organize outcomes and probabilities around the structure of the data. They are most useful when two or more processes occur in a sequence.

- ▶ Tree diagram splits the data into groups the first (primary branch) with respective marginal probabilities.
- ▶ The secondary branches are conditioned on the first, so we assign conditional probabilities to these branches.
- ▶ We may construct joint probabilities at the end of each branch in our tree by multiplying the numbers we come across (Multiplication Rule).



Calculate Probability through Simulation

We learned probability is the proportion of times the outcome would occur if we repeat the random process many times.

- ▶ View probability as long-run proportions allow us to calculate the probabilities of complicated events through simulation.
- ▶ We can use a computer to “simulate” many repetitions of a complicated event.
- ▶ Remember, probability is just the proportion of occurrence in many many repetitions!
- ▶ We will introduce simulation in a following lecture.

Example of Simulation

Morris's kidneys have failed and he is awaiting a kidney transplant. His doctor gives him this information for patients in his condition:

- ▶ 90% survive the transplant operation, and 10% die.
- ▶ The transplant succeeds in 60% of those who survive, and the other 40% must return to kidney dialysis.
- ▶ The proportions who survive for at least five years are 70% for those with a new kidney and 50% for those who return to dialysis.

Morris want to know what is the chance that he will survive for at least five years if he choose to do the operation.

Simulation Results

Steps of Simulation:

- ▶ Stage 1 (transplant operation): randomly generate an integer in $0 \sim 9$.
0=die;
1,2,3,4,5,6,7,8,9=survive (go to stage 2).
- ▶ Stage 2 (transplant success): randomly generate an integer in $0 \sim 9$.
0,1,2,3,4,5=transplant succeeds (go to 3A); 6,7,8,9=return to dialysis (go to 3B).
- ▶ Stage 3A (survive with transplant): randomly generate an integer in $0 \sim 9$.
0,1,2,3,4,5,6=survive; 7,8,9=die.
- ▶ Stage 3B (survive with dialysis): randomly generate an integer in $0 \sim 9$.
0,1,2,3,4=survive; 5,6,7,8,9=die.

Result of the first 4 repetitions:

	Repetition 1	Repetition 2	Repetition 3	Repetition 4
Stage 1	3	Survive	4	Survive
Stage 2	8	Dialysis	8	Dialysis
Stage 3	4	Survive	4	Survive

From a long simulation of 1000, we find that Morris survived in 558 of those which gives a probability 0.558 of living for at least five years.

STAT560 Lecture 10: Random Variable and Probability Distribution

Beidi Qiang

SIUE

- ▶ So far, we have already known how to calculate probabilities of events.
- ▶ Suppose we toss a fair coin three times, we know that the probability of getting three heads in a row is $1/8$. However, people maybe not interested in the probability directly related to the sample space, instead, some numerical summaries are of our interest.
- ▶ Suppose we are interested in the number of heads in an experiment of tossing a fair coin three times. The possible values are 0, 1, 2, and 3. The corresponding probabilities can be calculated.

Definition of Random Variable:

We call a variable or process with a numerical outcome a random variable, and we usually represent this random variable with a capital letter such as X, Y , or Z.

- ▶ The set of possible distinct values of the random variable is called its range.
- ▶ A outcome of the random variable is usually denoted by the lower-case letters $x, y,,$ and with possible subscripts $x_1, x_2.$

Type of Random Variables

- ▶ A **discrete** random variable can take one of a countable list of distinct values. Usually, counts are discrete.
- ▶ A **continuous** random variable can take any value in an interval of the real number line. Usually, measurements are continuous.
- ▶ Examples:
 - ▶ The number of times a transistor in computer memory changes state in one operation. (discrete)
 - ▶ The volume of gasoline that is lost to evaporation during the filling of a gas tank. (continuous)

Example of Discrete Random Variable

- Let $X = \text{Number of heads when tossing a coin 3 times}$

x	0	1	2	3
$P(X = x)$	1/8	3/8	3/8	1/8

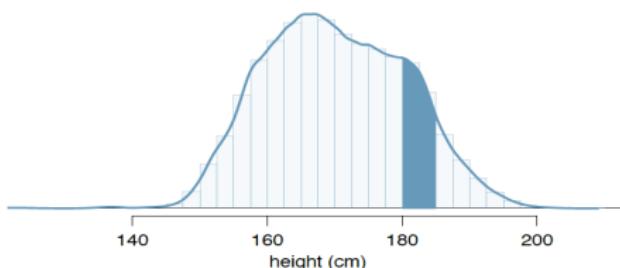
- Let $X = \text{Number of broken eggs in a half-dozen pack}$

x	0	1	2	3	4	5	6
$P(X = x)$	0.85	0.05	0.04	0.03	0.01	0.01	0.01

- Note 1:** $0 \leq P(X = x) \leq 1$, for all possible values of X
- Note 2:** The sum of the probabilities, taken over all possible values of X , must equal 1.

Example of Continuous Random Variable

The continuous probability distribution of heights for US adults:



- ▶ This smooth curve overlaid on the histogram is called a probability density function. It represents the probability distribution of a continuous random variable.
- ▶ Density function is non-negative. The total area under the curve is 1.
- ▶ The area under the density curve over a given range represents the probability that the random variable fall in the range.
For example: the probability that a randomly selected US adult is between 180 and 185 cm is the shaded area in the graph, which is approximately 0.1157.

Expected Value of a Random Variable

- ▶ **Definition:** Let X be a discrete random variable, the expected value of X , denoted as μ_X , or $E(X)$ is the sum of each outcome multiplied by its corresponding probability.

$$\mu_X = E(X) = x_1 \times P(X = x_1) + x_2 \times P(X = x_2) + \cdots = \sum x_i P(X = x_i)$$

The expected value for a discrete random variable X is simply a weighted average of the possible values of X . Each value x is weighted by its probability.

- ▶ In statistical applications, $\mu = E(X)$ is commonly called the **population mean**.

Example: Raffle

The campus raffle has two prizes: a \$200 gift certificate and a \$50 gift certificate. 1000 raffle tickets will be sold (at \$1 apiece). The possible returns and the corresponding probabilities are

Outcome	0	50	200
Prob	998/1000	1/1000	1/1000



- ▶ The expected winnings
 $= 0 \times 0.998 + 50 \times 0.001 + 200 \times 0.001 = 0.25.$
- ▶ A purchaser of one ticket has an "expected winnings" of 25 cents.
- ▶ The campus organization is expect to profit 75 cents per ticket on average (\$750 total).

- ▶ The expected value may not in fact be one of the possible values of the variable.
Raffle example: Expected value is 0.25, while the possible values that the buyer could win are 0, 50, 200.
- ▶ Expected value is a long-run average: If the experiment were repeated many times, the average value of the variable across those repetitions would be the expected value.
Raffle example: If the buyer bought many tickets, he would win about \$0.25 for each ticket purchased, on average.

Finding Expected Values with Simulation

- ▶ It is also possible to compute the expected value of a continuous random variable. We replace the summation in the previous definition by an integral over the range. This requires a little calculus. With the help of a computer, It's often easier to estimate the expected value through simulation.
- ▶ With complicated models, the calculations of expected value can become very difficult any way, so we again seek help from simulation.
- ▶ If a random phenomenon is repeated many times, the sample mean of these many outcomes will be close to the expected value (**Law of Large Numbers**).

Variance of a Random Variable

- ▶ **Definition:** Let X be a discrete random variable with expected value μ_X , then the variance of X , denoted by $\text{Var}(X)$ or the symbol σ_X^2 is

$$\sigma^2 \equiv \text{Var}(X) = \sum_{\text{all } x_i} (x_i - \mu_X)^2 P(X = x_i).$$



Note: Variance is **always** non-negative!

- ▶ **Definition:** The **standard deviation** of X is the positive square root of the variance:

$$\sigma_X = \sqrt{\sigma_X^2} = \sqrt{\text{Var}(X)}.$$

- ▶ To compute the variance of a continuous random variable, we again replace the summation in this definition by an integral over the range.

Calculate Variance: Raffle example

The campus raffle has two prizes: a \$200 gift certificate and a \$50 gift certificate. 1000 raffle tickets will be sold (at \$1 apiece). The possible returns and the corresponding probabilities are

Outcome	0	50	200
Prob	998/1000	1/1000	1/1000

$$\mu_X = 0 \times 0.998 + 50 \times 0.001 + 200 \times 0.001 = 0.25$$

$$\begin{aligned}\sigma_X^2 &= (0 - 0.25)^2 \times 0.998 + (50 - 0.25)^2 \times 0.001 + (200 - 0.25)^2 \times 0.001 \\ &= 0.0625 \times 0.998 + 2475.0625 \times 0.001 + 39900.0625 \times 0.001\end{aligned}$$

$$= 42.4375$$

$$\sigma_X = \sqrt{\text{Var}(X)} = \sqrt{42.4375} = 6.514407$$

Properties of Expected Value and Variance for Linear Combinations of Random Variables

A **linear combination** of two random variables X and Y is given by

$$aX + bY,$$

where a and b are some fixed and known numbers (constants).

Properties:

1. $E(aX + bY) = a \times E(X) + b \times E(Y)$.
2. $\text{Var}(aX + bY) = a^2 \times \text{Var}(X) + b^2 \times \text{Var}(Y)$ when X, Y are independent.
3. These properties hold the same for linear combination of 3 or more random variables.

STAT560 Lecture 11: Normal Distribution

Beidi Qiang

SIUE

- ▶ Most widely used continuous distribution.
- ▶ **Central Limit Theorem** (more later): Whenever a random experiment is replicated, the random variable that equals the average (or total) result over the replicates tends to have a normal distribution as the number of replicates becomes large.
- ▶ Other names: **Gaussian distribution**, “bell-shaped distribution” or “bell-shaped curve.”

Density of Normal Distribution

- ▶ A random variable X with probability density function (pdf)

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty$$

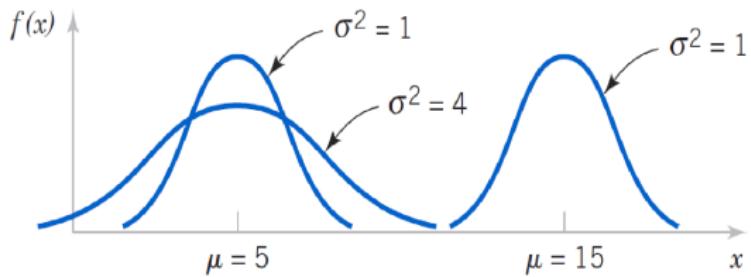
is a **normal random variable** with parameters μ and σ , where $-\infty < \mu < \infty$, and $\sigma > 0$. Also,

$$\text{E}(X) = \mu \quad \text{and} \quad \text{Var}(X) = \sigma^2.$$

- ▶ We use $X \sim \mathcal{N}(\mu, \sigma^2)$ to denote the distribution.
- ▶ Our objective now is to calculate probabilities (of intervals) for a normal random variable through R or normal probability table.

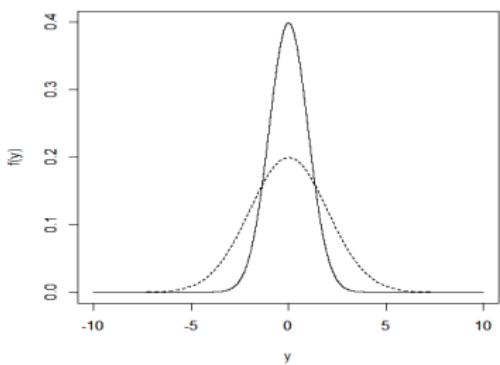
Density of Normal Distribution Cont'd

- The plot of normal distribution with different parameter values:



Characteristics of Normal Pdf

- ▶ Bell-shaped curve
- ▶ $-\infty < x < \infty$, i.e., the range of X is the whole real line
- ▶ μ determines distribution location and is the highest point on curve
- ▶ Curve is symmetric about μ
- ▶ σ determines distribution spread



Example: Strip of Wire

Assume that the current measurements in a strip of wire follow a normal distribution with a mean of 10 milliamperes and a variance of 4 (milliamperes)². What is the probability that a measurement exceeds 13 milliamperes?

Solution: Let X denotes the measure in that strip of wire. Then, $X \sim \mathcal{N}(10, 4)$. We want to calculate

$$P(X > 13) = 1 - P(X \leq 13).$$

```
> 1-pnorm(13,10,2)
[1] 0.0668072
```



Note code of calculating $F(x) = P(X \leq x)$ is of the form `pnorm(x, mu, sigma)`. The probabilities of the form $F(x) = P(X \geq x)$ is called the cumulative probability function (CDF).

Empirical Rule

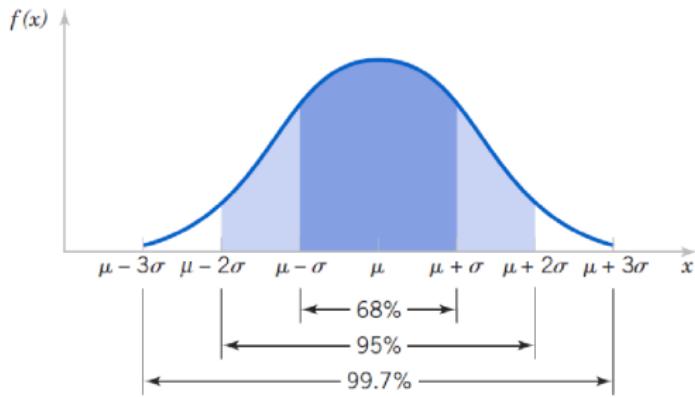
For any normal random variable X ,

$$P(\mu - \sigma < X < \mu + \sigma) = 0.6827$$

$$P(\mu - 2\sigma < X < \mu + 2\sigma) = 0.9543$$

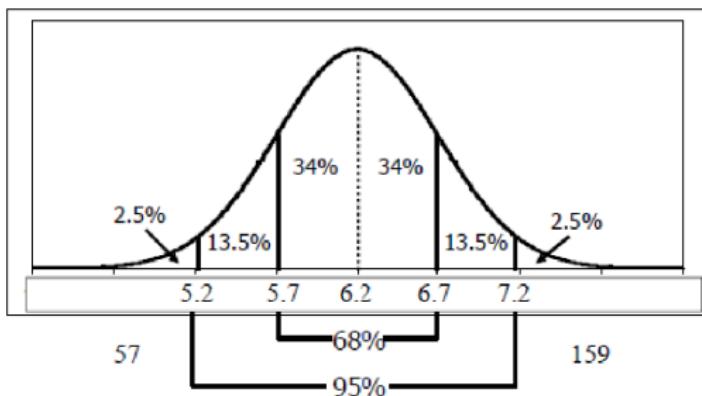
$$P(\mu - 3\sigma < X < \mu + 3\sigma) = 0.9973$$

These are summarized in the following plot:



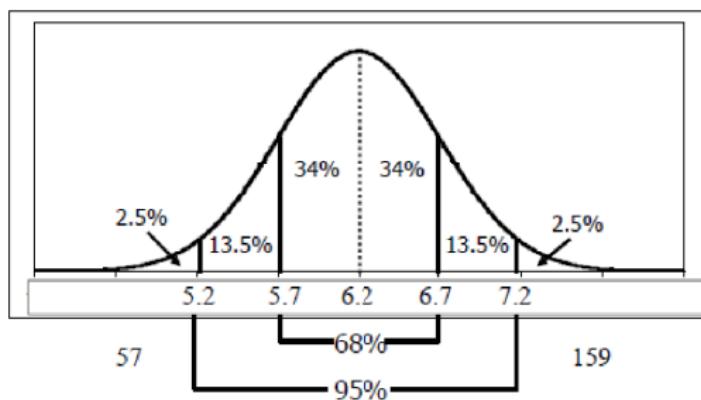
Earthquakes in a California Town

- ▶ Since 1900, the magnitude of earthquakes that measure 0.1 or higher on the Richter Scale in a certain location in California is distributed approximately normally, with $\mu = 6.2$ and $\sigma = 0.5$, according to data obtained from the United States Geological Survey.
- ▶ Earthquake Richter Scale Readings



Earthquakes in a California Town Cont'd

- ▶ Approximately what percent of the earthquakes are above 5.7 on the Richter Scale?
- ▶ What is the highest an earthquake can read and still be in the lowest 2.5%?
- ▶ What is the approximate probability an earthquake is above 6.7?



Standard Normal Distribution

- ▶ If X is a normal random variable with $E(X) = \mu$ and $\text{Var}(X) = \sigma^2$, the random variable

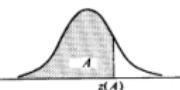
$$Z = \frac{X - \mu}{\sigma}$$

is a normal random variable with $E(Z) = 0$ and $\text{Var}(Z) = 1$. That is, Z is a standard normal random variable.

- ▶ Creating a new random variable by this transformation is referred to as standardizing.
- ▶ Z is traditionally used as the symbol for a standard normal random variable.

Normal Probability Table

Entry is area A under the standard normal curve from $-\infty$ to $z(A)$



z	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
.0	.5000	.5040	.5080	.5120	.5160	.5199	.5239	.5279	.5319	.5359
.1	.5399	.5438	.5478	.5517	.5557	.5596	.5636	.5675	.5714	.5753
.2	.5793	.5832	.5871	.5910	.5948	.5987	.6026	.6064	.6103	.6141
.3	.6179	.6217	.6255	.6293	.6331	.6368	.6406	.6443	.6480	.6517
.4	.6554	.6591	.6628	.6664	.6700	.6736	.6772	.6808	.6844	.6879
.5	.6915	.6950	.6985	.7019	.7054	.7088	.7123	.7157	.7190	.7224
.6	.7257	.7291	.7324	.7357	.7389	.7422	.7454	.7486	.7517	.7549
.7	.7580	.7611	.7642	.7673	.7704	.7734	.7764	.7794	.7823	.7852
.8	.7881	.7910	.7939	.7967	.7995	.8023	.8051	.8078	.8106	.8133
.9	.8159	.8186	.8212	.8238	.8264	.8289	.8315	.8340	.8365	.8389
1.0	.8413	.8438	.8461	.8485	.8508	.8531	.8554	.8577	.8599	.8621
1.1	.8643	.8665	.8686	.8707	.8729	.8749	.8770	.8790	.8810	.8830
1.2	.8849	.8869	.8888	.8907	.8925	.8944	.8962	.8980	.8997	.9015
1.3	.9032	.9049	.9066	.9082	.9099	.9115	.9131	.9147	.9162	.9177
1.4	.9192	.9207	.9222	.9236	.9251	.9265	.9279	.9292	.9306	.9319
1.5	.9332	.9345	.9357	.9370	.9382	.9394	.9406	.9418	.9429	.9441
1.6	.9452	.9463	.9474	.9484	.9495	.9505	.9515	.9525	.9535	.9545
1.7	.9554	.9564	.9573	.9582	.9591	.9599	.9608	.9616	.9625	.9633
1.8	.9644	.9649	.9656	.9664	.9671	.9678	.9686	.9693	.9699	.9706
1.9	.9713	.9719	.9726	.9732	.9738	.9744	.9750	.9756	.9761	.9767
2.0	.9772	.9778	.9783	.9788	.9793	.9798	.9803	.9808	.9812	.9817
2.1	.9821	.9826	.9830	.9834	.9838	.9842	.9846	.9850	.9854	.9857
2.2	.9861	.9864	.9868	.9871	.9875	.9878	.9881	.9884	.9887	.9890
2.3	.9893	.9896	.9898	.9901	.9904	.9906	.9909	.9911	.9913	.9916
2.4	.9918	.9920	.9922	.9925	.9927	.9929	.9931	.9932	.9934	.9936
2.5	.9938	.9940	.9941	.9943	.9945	.9946	.9948	.9949	.9951	.9952
2.6	.9953	.9955	.9956	.9957	.9959	.9960	.9961	.9962	.9963	.9964
2.7	.9965	.9966	.9967	.9968	.9969	.9970	.9971	.9972	.9973	.9974
2.8	.9974	.9975	.9976	.9977	.9977	.9978	.9979	.9979	.9980	.9981
2.9	.998	.9982	.9983	.9984	.9984	.9984	.9985	.9985	.9986	.9986
3.0	.9987	.9987	.9987	.9988	.9988	.9989	.9989	.9990	.9990	.9990
3.1	.9991	.9991	.9991	.9991	.9992	.9992	.9992	.9992	.9993	.9993
3.2	.9993	.9993	.9994	.9994	.9994	.9994	.9994	.9995	.9995	.9995
3.3	.9995	.9995	.9996	.9996	.9996	.9996	.9996	.9996	.9996	.9997
3.4	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9998

Normal Probability Table Cont'd

- With the help of normal probability table, we can calculate the probabilities for nonstandard normal distribution through **standardizing**.
- Suppose $X \sim N(10, 4)$, we want to calculate $P(X > 13)$.

$$\begin{aligned} P(X > 13) &= P\left(\frac{X - 10}{\sqrt{4}} > \frac{13 - 10}{\sqrt{4}}\right) \\ &= P(Z > 1.5) \\ &= 1 - P(Z \leq 1.5) \\ &= 1 - 0.9332 \text{ (from table)} \\ &= 0.0668 \end{aligned}$$

Normal Probability Table Cont'd

- ▶ If we want to calculate $P(X < 7)$,

$$\begin{aligned} P(X < 7) &= P\left(\frac{X - 10}{\sqrt{4}} < \frac{7 - 10}{\sqrt{4}}\right) \\ &= P(Z < -1.5) \\ &= 0.0668 \end{aligned}$$

- ▶ If we want to calculate $P(X > 7)$,

$$\begin{aligned} P(X > 7) &= P\left(\frac{X - 10}{\sqrt{4}} > \frac{7 - 10}{\sqrt{4}}\right) \\ &= P(Z > -1.5) \\ &= P(Z < 1.5) \text{ (by symmetry)} \\ &= 0.9332 \end{aligned}$$

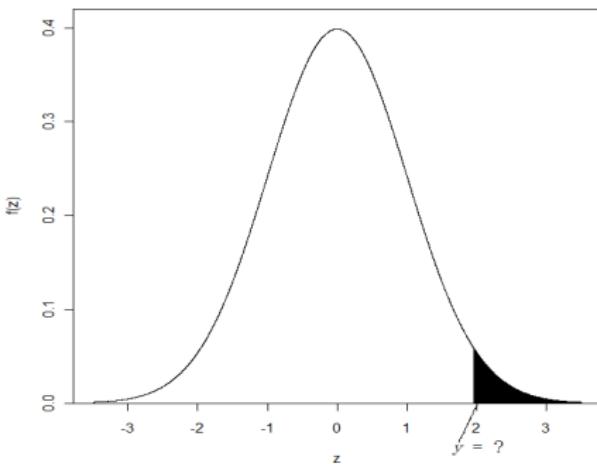
Example: Steel Bolt

- ▶ The thickness of a certain steel bolt that continuously feeds a manufacturing process is normally distributed with a mean of 10.0 mm and standard deviation of 0.3 mm. Manufacturing becomes concerned about the process if the bolts get thicker than 10.5 mm or thinner than 9.5 mm.
- ▶ Find the probability that the thickness of a randomly selected bolt is greater than 10.5 or smaller than 9.5 mm.

Solution:

Inverse Normal Probabilities

Sometimes we want to answer a question which is the reverse situation.
We know the probability, and want to find the corresponding value of Y .



Inverse Normal Probabilities Cont'd

What is the cutoff value that approximately 2.5% of the bolts produced will have thicknesses less than this value?

Solution: We need to find the z value such that $P(Z < z) = 0.025$. We can transform back to the original cutoff value from z . The R code of finding z is

```
> qnorm(0.025)   
[1] -1.959964
```

Or, $z = -1.96$ by table. It follows that

$$\begin{aligned} P(Z < -1.96) = 0.025 &\iff P\left(\frac{X - 10}{0.3} < -1.96\right) = 0.025 \\ &\iff P(X < 9.412) = 0.025 \end{aligned}$$

Therefore, the cutoff value is 9.412.

Inverse Normal Probabilities Cont'd

What is the cutoff value that approximately 1% of the bolts produced will have thicknesses greater than this value?

Solution:

STAT560 Lecture 12: Discrete Distributions - Binomial, Negative Binomial, Poisson

Beidi Qiang

SIUE

- ▶ **Definition:** A sequence of “trials”, are called **Bernoulli trials** if
 1. Each trial results in only two possible outcomes, labelled as “success” and “failure”.
 2. The trials are independent.
 3. The probability of a success in each trial, denoted as p , remains constant.

- ▶ **Definition:** Suppose that n Bernoulli trials are performed. Define

$Y = \text{the number of successes (out of } n \text{ trials performed).}$

We say that the random variable Y has a **binomial distribution** with number of trials n and success probability p .

- ▶ Shorthand notation is $Y \sim b(n, p)$.
- ▶ Range of Y is $0, 1, 2, \dots, n$.

Probability Mass Function of Binomial R.V.

- ▶ Suppose $Y \sim b(n, p)$.
- ▶ The pmf of Y is given by

$$p(y) = \begin{cases} \binom{n}{y} p^y (1-p)^{n-y} & \text{for } y = 0, 1, 2, \dots, n, \\ 0 & \text{otherwise.} \end{cases}$$



- ▶ $\binom{n}{r}$ is the number of ways to choose r distinct **unordered** objects from n distinct objects:

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}.$$

- ▶ $E(Y) = np$
- ▶ $\text{Var}(Y) = np(1-p)$.

Example: Radon Levels

Question: Historically, 10% of homes in Florida have radon levels higher than that recommended by the EPA. In a random sample of 20 homes, find the probability that exactly 3 have radon levels higher than the EPA recommendation. Assume homes are independent of one another.

Solution: Checking each house is a Bernoulli trial, which satisfies

- ✓ Two outcomes: higher than EPA recommendation or satisfies EPA recommendation.
- ✓ Homes are independent of one another.
- ✓ The case that the radon level is higher than the recommendation is considered as a “success”. The success probability remains 10%.

Example: Radon Levels Cont'd

So, the binomial distribution is applicable in this example. Define

$Y = \text{number of home having radon level higher than EPA.}$

We have $Y \sim b(20, 0.1)$.



$$P(Y = 3) = \binom{20}{3} 0.1^3 0.9^{20-3} = 0.1901.$$

Doing calculation by R is much easier:

```
> dbinom(3, 20, 0.1)  
[1] 0.1901199
```



Example: Radon Levels Cont'd

You can also calculate the probability that at least 6 homes out of the sample having higher radon level than recommended:

$$P[Y \geq 6] = P[Y = 6] + P[Y = 7] + \cdots + P[Y = 20]$$

$$P[Y \geq 6] = 1 - P[Y \leq 5] = 1 - (P[Y = 0] + P[Y = 1] + \cdots + P[Y = 5]).$$

Using R,

```
> 1-pbinom(5,20,0.1)
[1] 0.01125313
```

Question

A manufacturing process has a 0.03 defective rate. If we randomly sample 25 units

$$Y \sim b(25, 0.03)$$

- (a) What is the probability that less than 6 will be defective?

$$P(Y < 6) = P(Y \leq 5)$$

In R: `pbinom(5, 25, 0.03)`



- (b) What is the probability that 4 or more are defective?

$$P(Y \geq 4) = P(Y \leq 3)$$

In R: `pbinom(3, 25, 0.03)`



- (c) What is the probability that between 2 and 5, inclusive, are defective?

$$P(2 \leq Y \leq 5) = P(Y = 2) + P(Y = 3) + P(Y = 4) + P(Y = 5)$$

In R: `dbinom(2, 25, 0.03)+dbinom(3, 25, 0.03)+dbinom(4, 25, 0.03)+dbinom(5, 25, 0.03)`

$$\text{Alternatively, } P(2 \leq Y \leq 5) = P(Y \leq 5) - P(Y \leq 1)$$

In R: `pbinom(5, 25, 0.03)-pbinom(1, 25, 0.03)`



Negative Binomial Distribution and Geometric Distribution



Setting: The experiment consists of a sequence of independent trials. Each trial can result in either a success (S) or a failure (F). The probability of success is constant from trial to trial, denoted by p . The experiment continues (trials are performed) until a total of r successes have been observed.

- ▶ Let Y =number of trials until the r th success. Y is called a negative binomial random variable.
- ▶ Shorthand notation is $Y \sim nb(r, p)$.
- ▶ The pmf of Y is

$$P(Y = y) = \binom{y-1}{r-1} p^{y-r} (1-p)^r \quad y = r, r+1, r+2, \dots$$

- ▶ When $r=1$, Y is called a geometric random variable. Notation $Y \sim geo(p)$

Example: Negative Binomial Distribution

The probability that a wafer contains a large particle of contamination is 0.01. If it is assumed that the wafers are independent, what is the probability that exactly 125 wafers need to be analyzed until 5 large particle is detected?

Solution: Let X denote the number of samples analyzed until 5 large particle is detected. Then X is a negative binomial random variable with $p = 0.01$ and $r = 5$. The requested probability is

$$P(X = 125) = \binom{125 - 1}{5 - 1} p^{125 - 5} (1 - p)^5 = 0.00028$$

Mean and Variance of Negative Binomial Distribution

Suppose $Y \sim \text{nb}(y; r, p)$, the expected value of Y is

$$E(Y) = \frac{r}{p}.$$

$$\text{Var}(Y) = \frac{r(1-p)}{p^2}$$

Special case: Suppose $Y \sim \text{geo}(y; p)$, the expected value of Y is

$$E(Y) = \frac{1}{p}. \quad \text{[Speech bubble icon]}$$

$$\text{Var}(Y) = \frac{1-p}{p^2}$$

Lack of Memory Property

Because the trials are independent, the count of the number of trials until the next success can be started at any trial without changing the probability distribution of the random variable. Example:

The probability that a wafer contains a large particle of contamination is 0.01. If it is assumed that the wafers are independent, then on average, how many wafers need to be analyzed until 10 large particles are detected?



Solution:

X = number of wafers need to be analyzed until 10 large particles are detected.

$$X \sim nb(r = 10, p = 0.01)$$

$$E(X) = \frac{r}{p} = \frac{10}{0.01} = 1000$$

Intro to Poisson Distribution

Note: The Poisson distribution is commonly used to model counts, such as

1. the number of customers entering a post office in a given hour
2. the number of machine breakdowns per month
3. the number of insurance claims received per day
4. the number of defects on a piece of raw material.

Poisson Distribution

- ▶ Poisson distribution can be used to model the number of events occurring in a continuous time or space.
- ▶ Let λ be the average number of occurrences per base unit and t is the number of base units inspected.
- ▶ Let Y = the number of “occurrences” over in a unit interval of time (or space). Suppose Poisson distribution is adequate to describe Y . Then, the pmf of Y is given by

$$P(Y = y) = \begin{cases} \frac{(\lambda t)^y e^{-\lambda t}}{y!}, & y = 0, 1, 2, \dots \\ 0, & \text{otherwise.} \end{cases}$$

- ▶ The shorthand notation is $Y \sim \text{Poisson}(\lambda t)$.
- ▶ $E(Y) = \lambda t$
- ▶ $\text{Var}(Y) = \lambda t$

Example: Insulated Wire

Consider a process that produces insulated copper wire. Historically the process has averaged 2.6 breaks in the insulation per 1000 meters of wire. We want to find the probability that 1000 meters of wire will have 1 or fewer breaks in insulation?

► Solution:

Let $X = \text{number of breaks 1000 meters of wire will have}$. $\lambda = 2.6$ and $t = 1$, so $X \sim \text{Poisson}(2.6)$.

$$P(Y = 0 \text{ or } Y = 1) = \frac{2.6^0 e^{-2.6}}{0!} + \frac{2.6^1 e^{-2.6}}{1!}.$$

► Using R,

```
> dpois(0,2.6)+dpois(1,2.6)
[1] 0.2673849
> ppois(1,2.6)
[1] 0.2673849
```



Insulated Wire Cont'd

- ▶ If we were inspecting 2000 meters of wire, $\lambda t = (2.6)(2) = 5.2.$

$$Y \sim \text{Poisson}(5.2).$$

- ▶ If we were inspecting 500 meters of wire, $\lambda t = (2.6)(0.5) = 1.3.$

$$Y \sim \text{Poisson}(1.3).$$

Conditions for a Poisson Distribution

- ▶ Areas of inspection are independent of one another.
- ▶ The probability of the event occurring at any particular point in space or time is negligible.
- ▶ The mean remains constant over all areas of inspection.

The pmf of Poisson distribution is derived based on these three assumption!

<http://www.pp.rhul.ac.uk/~cowan/stat/notes/PoissonNote.pdf>

STAT560 - Foundation of Data Science - Lecture 13

Iteration and Simulation

Programming languages free humans from having to perform iterative computations by re-running chunks of code, or worse, copying-and-pasting a chunk of code many times, while changing just one or two things in each chunk. In this lecture, we present a variety of techniques for automating these types of iterative operations.

Writing a function in R

In R, a new function is defined by the syntax shown below, using the keyword “function”. This creates a new object in the workspace that takes a set of arguments specified in the function(...). The body is made up of a series of commands or expressions, typically separated by line breaks and enclosed in curly braces. You may explicitly return() a list of values. If no return statement is provided, the result of the last expression evaluation is returned by default.

```
myf1=function(arg1,arg2,arg3){  
  a=arg1+arg2+arg3  
  b=arg1*arg2*arg3  
  return(list(sum=a,product=b))  
}  
myf1(3,5,2)
```

```
## $sum  
## [1] 10  
##  
## $product  
## [1] 30  
  
myf2=function(arg1,arg2,arg3){  
  arg1+arg2+arg3  
  arg1*arg2*arg3  
}  
myf2(3,5,2)
```



```
## [1] 30
```

Here we write a function to calculate the pmf of a binomial random variable

```
pmf_binom=function(x=1,n=1,p=0.5){  
  n_c_x=factorial(n)/(factorial(x)*factorial(n-x))  
  pmf=n_c_x*p^x*(1-p)^(n-x)  
  return(probability=pmf)  
}  
pmf_binom()
```

```
## [1] 0.5
```

```
pmf_binom(10,30,0.7)
```

```
## [1] 2.959225e-05
```

```
dbinom(10,30,0.7) #compare with dbinom() function from R base package
```

```
## [1] 2.959225e-05
```

Write for() loop

A for-loop is used to iterate over a vector in R programming. The basic syntax of a for-loop in R is for (val in sequence) {statement}.

```
x=c(2,5,3,9,8,11,6)
count=0
for (val in x) {
  if(val %% 2 == 0) count = count+1 # the operation %% gives the remainder
}
print(count) # we counted the number of even numbers in x
```

```
## [1] 3
```

Vectorized operations

The use of a for-loop in R is generic but may not be ideal in situations when it is possible (and usually preferable) to iterate without explicitly defining a loop. R programmers prefer to solve this type of problem by applying an operation to each element in a vector.

R is highly optimized for vectorized operations (see Appendix B in msdr book for more detailed information about R internals). Loops, by their nature, do not take advantage of this optimization. Thus, R provides several tools for performing loop-like operations without actually writing a loop. This is different from general-purpose programming languages like C++ or Python. Many functions in R are vectorized. This means that they will perform an operation on every element of a vector by default.

```
x=c(2,5,3,9,8,11,6)
exp(x)

## [1]      7.389056   148.413159   20.085537  8103.083928 2980.957987
## [6] 59874.141715   403.428793
x %% 2 == 0

## [1] TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE
count=sum(x %% 2 == 0)
print(count) # we again counted the number of even numbers in x without writing a loop
```

```
## [1] 3
```

Using summarize() and across() with dplyr functions

The across() adverb allows us to specify the set of variables that summarize() includes.

```
library(tidyverse)
myData=read.csv(file="https://www.openintro.org/data/csv/loan50.csv",header=T)
myData %>%
  summarize(across(where(is.numeric), mean))

##   emp_length  term annual_income debt_to_income total_credit_limit
## 1           NA    42.72          86170        0.7226426            208546.6
##   total_credit_utilized num_cc_carrying_balance loan_amount interest_rate
## 1                61546.54                      5.06       17083        11.5672
```

```
##   public_record_bankrupt total_income  
## 1                      0.08      105220.6
```

The apply() and map() family of functions

More generally, to apply a function to each item in a list or vector, or a data matrix, you may use `apply()` or `map()` (or one of their variants). `apply()` is available in R base package while `map()` function is from the `purrr` package. You will need to install the `purrr` package first.

```
library(tidyverse)  
library(purrr)
```

```
myData=read.csv(file="https://www.openintro.org/data/csv/loan50.csv",header=T)  
d1=myData %>%  
  select(annual_income,total_credit_limit,num_cc_carrying_balance,loan_amount,total_income)  
apply(d1,2,mean)
```

```
##           annual_income      total_credit_limit num_cc_carrying_balance  
##            86170.00          208546.64                  5.06  
##           loan_amount      total_income  
##            17083.00          105220.56
```

```
sapply(d1,mean)
```



```
##           annual_income      total_credit_limit num_cc_carrying_balance  
##            86170.00          208546.64                  5.06  
##           loan_amount      total_income  
##            17083.00          105220.56
```

```
lapply(d1,mean)
```



```
## $annual_income  
## [1] 86170  
##  
## $total_credit_limit  
## [1] 208546.6  
##  
## $num_cc_carrying_balance  
## [1] 5.06  
##  
## $loan_amount  
## [1] 17083  
##  
## $total_income  
## [1] 105220.6
```

```
map_dbl(d1,mean)
```

```
##           annual_income      total_credit_limit num_cc_carrying_balance  
##            86170.00          208546.64                  5.06  
##           loan_amount      total_income  
##            17083.00          105220.56
```

`apply()` and `map()` can be used to iterate an arbitrary function over a vector of values.

```
top3 = function(data, cr_grade) {  
  data %>%  
    filter(grade == cr_grade) %>%
```

```
    select(grade,annual_income,loan_amount) %>%
  arrange(desc(annual_income)) %>%
  head(n = 3)
}
```

```
gd = c("A","B","C","D","E")
gd %>% map(top3, data=myData)
```

```
## [[1]]
##   grade annual_income loan_amount
## 1     A      325000      35000
## 2     A      120000      12000
## 3     A      105000      16500
##
## [[2]]
##   grade annual_income loan_amount
## 1     B      254000      25000
## 2     B      245000      35000
## 3     B      185000      15000
##
## [[3]]
##   grade annual_income loan_amount
## 1     C      160000      38500
## 2     C      80000       18500
## 3     C      80000       5000
##
## [[4]]
##   grade annual_income loan_amount
## 1     D      76000       18000
## 2     D      73000       17000
## 3     D      68700       13500
##
## [[5]]
##   grade annual_income loan_amount
## 1     E      98000       29400
## 2     E      75000       25000
```

```
lapply(gd,top3,data=myData)
```

```
## [[1]]
##   grade annual_income loan_amount
## 1     A      325000      35000
## 2     A      120000      12000
## 3     A      105000      16500
##
## [[2]]
##   grade annual_income loan_amount
## 1     B      254000      25000
## 2     B      245000      35000
## 3     B      185000      15000
##
## [[3]]
##   grade annual_income loan_amount
## 1     C      160000      38500
```

```

## 2      C      80000     18500
## 3      C      80000      5000
##
## [[4]]
##   grade annual_income loan_amount
## 1    D      76000     18000
## 2    D      73000     17000
## 3    D      68700     13500
##
## [[5]]
##   grade annual_income loan_amount
## 1    E      98000     29400
## 2    E      75000     25000

```

Simulation

It can be useful to repeat an operation many times and collect the results. If this operation involves randomness, then you won't get the same answer every time. So through the simulation we get a distribution of outcome values and understanding the distribution of values can be useful. Below are some examples of simulation.

Calculate probability using simulation:

```

#Morris's Kidney example in Lecture 9
M = 5000
res=data.frame((matrix(ncol = 4, nrow = M)))
colnames(res) = c("stage1", "stage2", "stage3","final")
for (i in 1:M){
  stage1=sample(c("S","D"),1,prob=c(0.9,0.1))
  res[i,1]=stage1
  if (stage1=="D"){res[i,4]=stage1}
  else{
    stage2=sample(c("S","D"),1,prob=c(0.6,0.4))
    res[i,2]=stage2
    if (stage2=="S"){stage3=sample(c("S","D"),1,prob=c(0.7,0.3))}
    if (stage2=="D"){stage3=sample(c("S","D"),1,prob=c(0.5,0.5))}
    res[i,3]=stage3
    res[i,4]=stage3
  }
}
print(res[1:5,])

##   stage1 stage2 stage3 final
## 1      S      S      D      D
## 2      S      S      S      S
## 3      S      S      S      S
## 4      S      S      S      S
## 5      S      D      D      D

table(res$final)

##
##      D      S
## 2219 2781

table(res$final)/M

```

```
##
```

```
##      D      S
```

```
## 0.4438 0.5562
```

```
#simulate the expected value of a binomial random variable
```

```
bn_sim=rbinom(10000,size=10,prob=0.3)
```

```
mean(bn_sim)
```

```
## [1] 2.978
```

```
#get the sampling distribution of a sample mean by resampling. This is a statistical technique known as
```

```
n=5000
```

```
bs=1:n %>%
```

```
map_dbl(
```

```
~myData %>% pull (total_income) %>%
```

```
sample(replace = TRUE) %>%
```

```
mean()
```

```
)
```

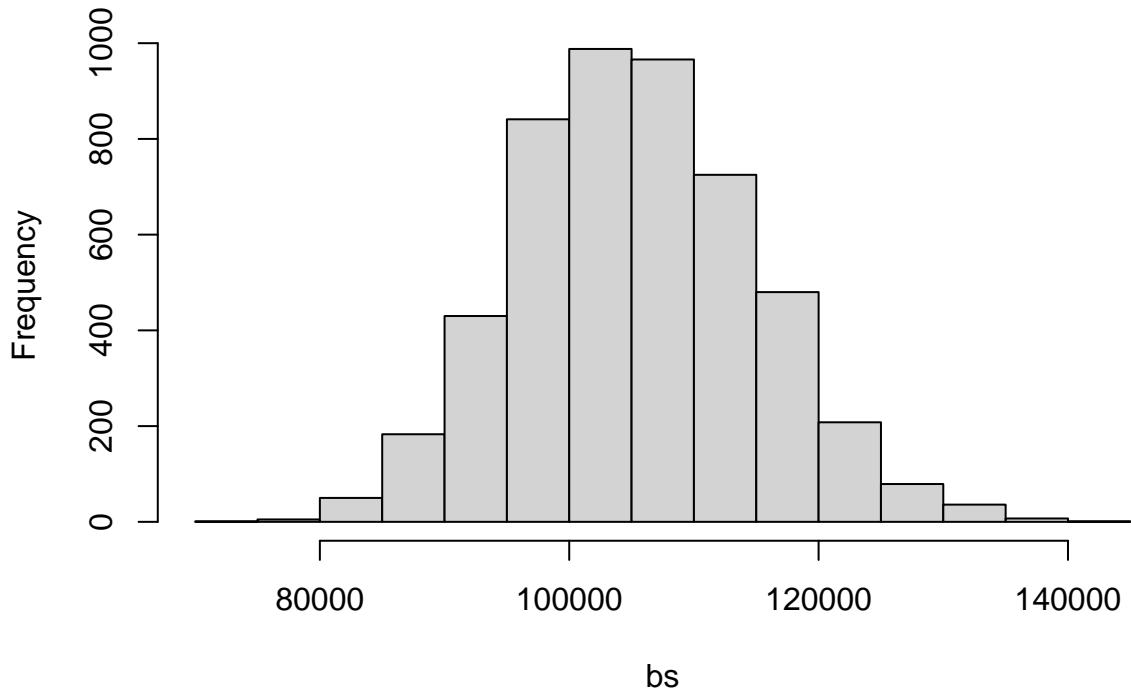
```
summary(bs)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
```

```
##    72730    98519   105015   105339   111633   140442
```

```
hist(bs)
```

Histogram of bs



STAT560 Lecture 14: Sampling Distribution and Central Limit Theorem

Beidi Qiang

SIUE

Intro to sampling variability

Suppose a poll of 500 participants suggested the US President's approval rating is 45%. We would consider 45% to be a point estimate of the approval rating we might see if we collected responses from the entire population.

- ▶ **Sampling variability** Imagine taking another samples of 500 and calculating the sample approval rating. Would the number stay the same each time?
- ▶ **Sampling distribution** What are the possible values the sample approval rating can have? How often would we expect it to take those values?
- ▶ **Error** Unless we collect responses from every individual in the population, the approval rating remains unknown. The difference we observe from the poll versus the true approval rating is called the error in the estimate.

- ▶ A **population** refers to the entire group of "individuals" (e.g., parts, people, batteries, etc.) about which we would like to make a statement (e.g., defective proportion, median weight, mean lifetime, etc.).
 - ▶ Problem: Population can not be measured (generally)
 - ▶ Solution: We observe a **sample** of individuals from the population to draw inference
 - ▶ We denote a random sample of observations by

$$Y_1, Y_2, \dots, Y_n$$

- ▶ **n is the sample size**

- ▶ A **parameter** is a numerical quantity that describes a *population*. In general, population parameters are unknown.
- ▶ Some very common examples are:
 - ▶ μ = population mean
 - ▶ σ^2 = population variance
 - ▶ σ = population standard deviation
 - ▶ p = population proportion
- ▶ Connection: all of the probability distributions that we talked about in previous chapter are indexed by population (model) parameters.

- ▶ A **statistic** is a numerical quantity that can be calculated from a sample of data.
- ▶ Suppose Y_1, Y_2, \dots, Y_n is a random sample from a population, some very common examples are:

- ▶ **sample mean:**

$$\bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i$$

- ▶ **sample variance:**

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (Y_i - \bar{Y})^2$$

- ▶ **sample standard deviation:** $S = \sqrt{S^2}$

- ▶ **sample proportion:** $\hat{p} = \frac{1}{n} \sum_{i=1}^n Y_i$ if Y_i 's are binary.

SUMMARY: The table below succinctly summarizes the salient differences between a population and a sample (a parameter and a statistic):

Comparison between parameters and statistics	
Statistics	Parameters
<ul style="list-style-type: none">• Describes a sample• Always known• Changes upon repeated sampling• Ex: \bar{X}, S^2, S, \hat{p}	<ul style="list-style-type: none">• Describes a population• Usually unknown• Is fixed but unknown• Ex: μ, σ^2, σ, p

We consider the sample statistics as point estimators of the population parameters of interest.

Terminology: sampling distribution

- ▶ The distribution of a sample statistic $\hat{\theta}$ is called its **sampling distribution**.
- ▶ A sampling distribution describes mathematically how $\hat{\theta}$ would vary in repeated sampling.

Simulate a Sampling Distribution

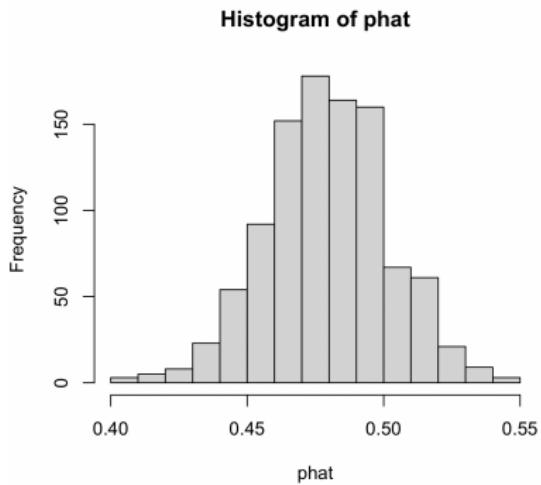
Suppose US President's approval rating is 48% in the population of all American adults. We were to take a poll of 500 American adults. How do we expect the sample approval rating in the poll to be?

- ▶ population proportion: $p =$ approval rating for all American adults= 48%, There were about 260 million American adults.
- ▶ sample proportion: $\hat{p} =$ approval rating in a poll of 500

Simulation in R:

```
pop_size = 2600000000
pop_entries = c(rep("approve", 0.48*pop_size), rep("disapprove", 0.52*pop_size))
phat=vector(length=1000)
for (i in 1:1000) {
  sampled_entries = sample(pop_entries, size = 500)
  phat[i]=sum(sampled_entries == "approve") /500
}
```

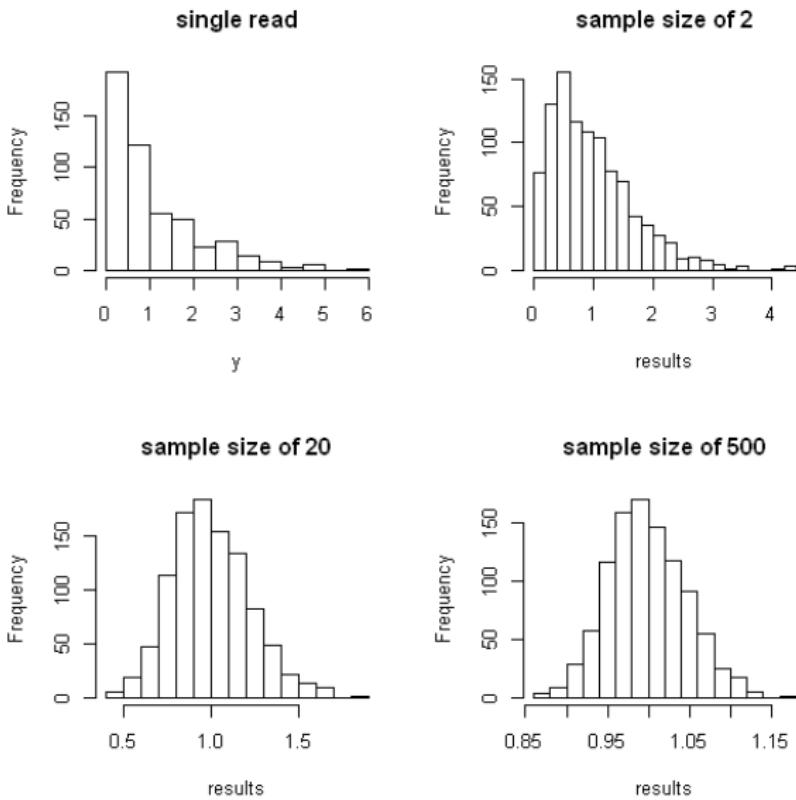
Simulation Result



- ▶ The center of the distribution is about 0.48, which is the same as the parameter.
- ▶ The standard deviation of the distribution is 0.022. When we're talking about a sampling distribution or the variability of a estimate, we typically use the term standard error rather than standard deviation.
- ▶ The distribution is roughly symmetric and bell-shaped, and it resembles a normal distribution.

- ▶ **Central Limit Theorem:** Consider independent random sample from a population with mean μ and variance σ^2 . When the sample size n is sufficiently large, the sample mean \bar{X} will tend to follow a normal distribution with mean μ and standard error σ/\sqrt{n} .
Rule of thumb: If $n > 30$, CLT can be used.
- ▶ **CLT for sample proportion:** When observations are independent and the sample size is sufficiently large from a population with proportion p , the sample proportion \hat{p} will tend to follow a normal distribution with mean p and standard error $\sqrt{p(1-p)/n}$.
Rule of thumb: $np > 10$ and $n(1-p) > 10$, CLT can be used.

Simulation Study of CLT



Example:

Suppose the time needed to grade a randomly chosen exam paper is a random variable with a mean of 6 min and a standard deviation of 5 min. Take a random sample of 36 exam papers.

- ▶ What is the approximate distribution of average grading time?
 \bar{X} is normally distributed with mean 6 and standard deviation $5/\sqrt{36} = 0.833$ 
- ▶ What is the probability that the instructor will finish grading within 3.5 hrs (210 min)?
 $P(36 * \bar{X} < 210) = P(\bar{X} < 5.833) = P(Z < -0.2) = 0.4207$

Example:

Suppose US President's approval rating is 48% in the population of all American adults. We were to take a poll of 500 American adults.

- ▶ What is the mean and standard error of sample proportion of approval?
 $\text{mean} = 0.48$, $\text{std.error} = \sqrt{0.48 \times (1 - 0.48) / 500} = 0.022$
- ▶ What is the probability that the sample proportion is within 0.05 (5%) of the population approval rating?
 $P(0.43 < \hat{p} < 0.53) = P(-2.27 < Z < 2.27) = P(Z < 2.27) - P(Z < -2.27) = 0.9768$
- ▶ If, in reality, the proportion of approval in a poll of 500 is 42%. Are you going to question the statement "US President's approval rating is 48% in the population of all American adults"?

Intro to confidence interval

If we report a point estimate, we probably will not hit the exact population parameter. Since a point estimator are not perfect and will have some standard error associated with it, it is better practice to report a range of plausible values, representing a confidence interval, and so we have a good shot at capturing the parameter.

- ▶ Point estimate is the most plausible value of the population parameter so it makes sense to build a confidence interval around this point estimate.
- ▶ The standard error provides a guide for how large we should make the confidence interval.
- ▶ If we want to be more certain (confidence level) that we capture the parameter, we use a wider confidence interval.

Example:

A poll of 500 participants suggested the US President's approval rating is 45%.
point estimate for population approval rating is 45%.
95% confidence interval for population approval rating is (40.6%, 49.4%).

Intro to hypothesis testing

We may want to understand a claim about a population parameter. For example: the population proportion less than 50%. Such a statement is called a hypothesis. Our job as data scientists is to play the role of a skeptic: before we buy into the statement, we need to see strong supporting evidence. We do that through the hypothesis testing framework.

- ▶ **Null and alternative hypotheses:** The set of competing ideas. The null hypothesis (H_0) often represents a skeptical position or a perspective of no difference. The alternative hypothesis (H_1) generally represents a new or stronger perspective.
- ▶ **Evidence:** The p-value is a way of quantifying the strength of the evidence (in the observed data) against the null hypothesis and in favor of the alternative hypothesis.
- ▶ **Conclusion:** p-value to a pre-defined level (level of significance) to see if the evidence is strong enough. If so, we reject null and support the alternative.

STAT560 Lecture 15: Statistical Inference for Categorical Data

Beidi Qiang

SIUE

Inference on Population Proportion

The population proportion p emerges when the characteristic we measure on each individual is categorical and binary. Some examples:

p = US President's approval rating (proportion of approval)

p = proportion of defective water filters in a factory

p = proportion of Covid positive at SIUE

We make the following assumptions for each individual in the sample:

1. Each trial results in only two possible outcomes, labeled as "success" and "failure."
2. The sample's observations are independent.
3. The probability of a success for each individual, denoted as p , remains constant. It follows that the probability of a failure in each trial is $1 - p$.

Notice these assumptions are the same as how we define *Bernoulli trials*.

Point Estimator of Proportion p

Suppose we take a sample of size n from a population with a true proportion p . A natural point estimator for p , the population proportion, is the sample proportion,

$$\hat{p} = \frac{\text{number of success in the sample}}{n}.$$

- ▶ **Accuracy:** \hat{p} is an unbiased estimator of p . That is,

$$E(\hat{p}) = p.$$

- ▶ **Precision:** The standard error of \hat{p} is

$$SE(\hat{p}) = \sqrt{\text{Var}(\hat{p})} = \sqrt{\frac{p(1-p)}{n}}.$$

Note: estimator with smaller standard error or variance is more precise.

Sampling Distribution of \hat{p}

Suppose that X_1, X_2, \dots, X_n is a random sample from a binary population with population proportion p . When the sample size n is large, the sampling distribution for \hat{p} is nearly normal with

$$\text{mean } E(\hat{p}) = p, \text{ standard error } SE(\hat{p}) = \sqrt{p(1-p)/n}.$$

Conditions:

- ▶ The sample's observations are independent, e.g. are from a simple random sample.
- ▶ At least 10 successes and 10 failures in the sample (success-failure condition).

Using a point estimator only ignores important information; namely, how variable the estimator is. We therefore pursue the topic of interval estimation (also known as confidence intervals).

The main difference between a point estimate and an interval estimate is that

- ▶ a **point estimate** is a one shot guess at the value of the parameter. It is like throwing a spear at a fish in a murky lake. We will likely miss it!
- ▶ a **confidence interval** is an interval of values. It is formed by taking the point estimate and then adjusting it downwards and upwards to account for the point estimate's variability. It is like toss a net in that area, we have a good chance of catching the fish!

Confidence Interval for p

- An approximate $100(1 - \alpha)\%$ confidence interval for p is

$$\left(\hat{p} - z^* \times \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}}, \hat{p} + z^* \times \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}} \right).$$

- $100(1 - \alpha)\%$ is called the confidence level.
- The quantity $z^* \times \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}}$ is called the **margin of error**.
- Note of the form of the interval:
point estimate $\pm Z^* \times$ standard error 
- Z^* is the standard normal quantile (Z-score) corresponds to the confidence level selected.

Confidence level and Z^*

The choice of Z^* corresponds to the confidence level of a confidence interval. The value of Z^* increases as confidence level goes higher. Commonly used confidence levels and corresponding z-score:

confidence level	90%	95%	99%
z-score	1.645	1.96	2.58

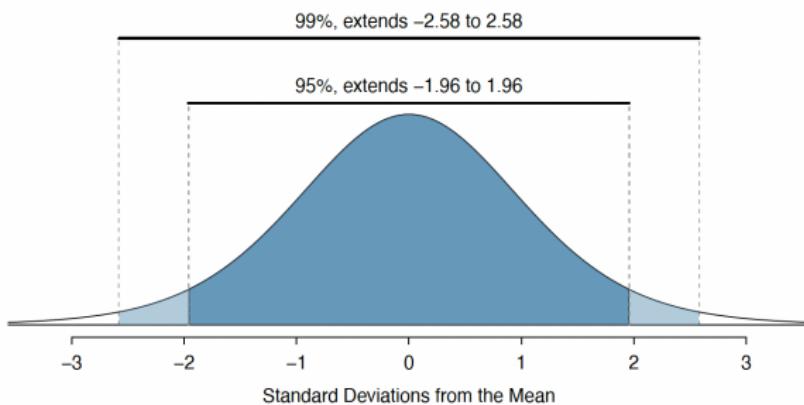


Figure 5.7: The area between $-z^*$ and z^* increases as z^* becomes larger. If the confidence level is 99%, we choose z^* such that 99% of a normal normal distribution is between $-z^*$ and z^* , which corresponds to 0.5% in the lower tail and 0.5% in the upper tail: $z^* = 2.58$.

Interpretation of Confidence Interval

Suppose that we are interested in parameter p for certain population. We take a sample of size n and calculate the sample proportion \hat{p} . A 95% confidence interval is given by $\hat{p} \pm 1.96 \times \sqrt{\hat{p}(1 - \hat{p})/n}$. What does "95%" confident mean here?



Suppose we took many samples and built a 95% confidence interval from each. Then about 95% of those intervals would contain the true parameter, p and only 5% of the time would the interval be in error.

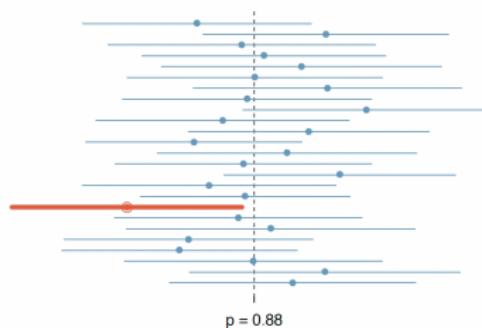


Figure 5.6: Twenty-five point estimates and confidence intervals from the simulations in Section 5.1.2. These intervals are shown relative to the population proportion $p = 0.88$. Only 1 of these 25 intervals did not capture the population proportion, and this interval has been bolded.

Steps to a Hypothesis Test

1. State the **null** and **alternative** hypotheses. These are statements about p .
2. Collect the data and summarize the evidence by calculate **p -value** assuming null is true. 
3. Draw conclusion based on the p -value. We either **reject null or fail to reject null**.

Let's illustrate these steps...

The Null and Alternative Hypothesis

- ▶ *Null hypothesis* is denoted by H_0 , which represents a skeptical position or a perspective of “no difference”.
- ▶ *Alternative hypothesis* is denoted by H_a , which represents the researcher's interest or claim.
- ▶ In most situation, we want to reject null hypothesis in favor of the alternative hypothesis by performing some experiment.
- ▶ Example: Is the coin biased? p=proportion of head



$$H_0 : p = 0.5$$

$$H_a : p \neq 0.5$$

Calculate p-value



The **p-value** is the probability of getting the sample results you got or something more extreme assuming that the null hypothesis is true. We typically use a summary statistic of the data, called test statistics, to help compute the p-value.

- ▶ assume $H_0 : p = p_0$ is true, i.e., the population proportion is p_0
- ▶ Test statistics is the sample proportion standardized by its mean and standard error.



$$z_0 = \frac{\text{point estimate} - \text{null value}}{\text{SE of point estimate}} = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}}$$



- ▶ P-value is probability of observing such an extreme sample proportion.

Alternative hypothesis	Hypothesis type	p-value formula
$H_a : p < p_0$	Left-tail hypothesis	$P(Z < z_0)$
$H_a : p > p_0$	Right-tail hypothesis	$P(Z > z_0)$
$H_a : p \neq p_0$	Two-tail hypothesis	$2P(Z < - z_0)$

Interpreting p-value

P-value for a hypothesis test measures how much evidence we have against H_0 , that is,

the smaller the p-value \implies the more evidence against H_0

- ▶ If the p-value is small, we doubt the null hypothesis since it is not likely to observe such a "extreme" test statistic under H_0 . There is evidence to against null hypothesis.
- ▶ On the other hand, if the p-value is large, we have a pretty good chance to observe the computed test statistic under H_0 in a single experiment, there is no reason to question the H_0 .
- ▶ There is one remaining question, how small should p-value be to be considered as "small"? We need level of significance to answer it.



Level of significance and the conclusion

We use α to denote the level of significance. Level of significance is determined before you see the data. We simply compare the p -value with the α level.

- ▶ If the p -value is less than α , we **reject the null hypothesis** and **conclude we have enough evidence to support the alternative hypothesis**.
- ▶ If the p -value is greater than or equal to α , we **do not reject the null hypothesis** and **conclude we don't have enough evidence to support the alternative hypothesis**.

Example:



A certain type of flu outbreaks in northern part of the USA. The historical records shows that there are 7% of the residences in St. Louis area carrying flu under usual condition. Researchers want to see whether there is an outbreak (more flu cases than usual) in St. Louis and they take a random sample of residences. There are 30 out of 250 randomly chosen people in the sample carrying flu. Can we conclude that there is an outbreak?



Inference for p : Confidence Interval Approach

Recall that a $100(1 - \alpha)\%$ C.I. is

$$\left(\hat{p} - z^* \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}}, \quad \hat{p} + z^* \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}} \right).$$

- ▶ The point estimate $\hat{p} = 30/250 = 0.12$.
- ▶ $z^* = 1.96$ for 95% confidence level.
- ▶ Standard error: $\sqrt{0.12(1 - 0.12)/250} = 0.021$

- ▶ 95% CI is: $(0.079, 0.161)$



- ▶ Conclusion?

We are 95% confident, the proportion of all residences in St. Louis carrying flu is between 7.9% and 16.1%



Inference for p : Hypothesis Test



- ▶ Step 1: State H_0 and H_1

$$H_0 : p = 0.07$$

$$H_a : p > 0.07$$

- ▶ Step 2: Calculate test statistic and p -value assuming H_0 is true



$$Z = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{n}}} = \frac{0.12 - 0.07}{\sqrt{\frac{0.07(1-0.07)}{250}}} = 3.10.$$

$$p\text{-value} = P(Z > 3.10) \approx 0.001.$$

- ▶ Step 3: Draw the conclusion

$\alpha = 0.05$, p -value is smaller than 0.05. We reject H_0 , and conclude that there is an outbreak.

Method of Evaluating a Test: Type I and Type II Errors

There are two mistakes we can make in a hypothesis test.

- ▶ Type I error: H_0 is rejected but in reality H_0 is true
- ▶ Type II error: H_0 is not rejected but in reality H_1 is true

		Do not reject H_0	Reject H_0
H_0 is true	No Error	Type I error	
	Type II error	No Error	

- ▶ If we reduce how often we make one type of error, we generally make more of the other type.
- ▶ We do not reject H_0 unless we have strong evidence. In hypothesis testing, we control the probability of making a Type I error.

$$P(\text{type I error}) = \alpha = \text{level of significance}$$



Inference for difference of two proportions

We would like to extend the methods to apply confidence intervals and hypothesis tests to differences in population proportions: $p_1 - p_2$. For example, we can compare

1. defective rate of water filters for two different suppliers
2. the proportion of on-time payments for two classes of customers

► Assumptions:

1. The data are independent within and between the two groups. This is satisfied if the data come from two independent random samples
 2. The success-failure condition holds for both groups, i.e. at least 10 success and 10 failures in each group.
- Point estimate: $\hat{p}_1 - \hat{p}_2$.
- Sampling distribution of $\hat{p}_1 - \hat{p}_2$: Normal with mean $p_1 - p_2$ and standard error

$$SE = \sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}$$

► Confidence interval:

$$(\hat{p}_1 - \hat{p}_2) \pm Z^* \sqrt{\frac{\hat{p}_1(1 - \hat{p}_1)}{n_1} + \frac{\hat{p}_2(1 - \hat{p}_2)}{n_2}}$$

► Hypothesis test:

Hypothesis:

$H_0 : p_1 = p_2 = 0$, $H_1 : p_1 > p_2$, $<$, or $\neq 0$



The test statistic under H_0 :

$$\text{z}_0 = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}_{pooled}(1 - \hat{p}_{pooled})(\frac{1}{n_1} + \frac{1}{n_2})}}, \text{ where } \hat{p}_{pooled} = \frac{\text{total number of successes}}{n_1 + n_2}$$

P-value:

For test $H_a : p_1 \neq p_2$, the p-value is $2P(Z < -|z_0|)$;

For test $H_a : p_1 < p_2$, the p-value is $P(Z < z_0)$;

For test $H_a : p_1 > p_2$, the p-value is $P(Z > z_0)$.

Example: Proportion of Exceedence

Airplanes approaching the runway for landing are required to stay within the localizer (a certain distance left and right of the runway). When an airplane deviates from the localizer, it is sometimes referred to as an exceedence.

Consider two airlines at a large airport. During a three-week period, airline 1 had 14 exceedences out of 156 flights and airline 2 had 11 exceedences out of 198 flights. We are interested in whether airline 1 has a exceedence rate that is significantly different from airline 2.

- ▶ $\hat{p}_1 = 14/156 = 0.090$, $\hat{p}_2 = 11/198 = \square$, $\hat{p}_{pooled} = (14 + 11)/(156 + 198) = 0.071$
- ▶ 95% CI:

$$(0.090 - 0.056) \pm 1.96^* \sqrt{\frac{0.090(1 - 0.090)}{156} + \frac{0.056(1 - 0.056)}{198}} = (-0.0208, 0.089)$$

- ▶ Hypothesis test:
 $H_0: p_1 = p_2 = 0$; $H_1: p_1 \neq p_2 \neq 0$

Test statistic:

$$z_0 = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}_{pooled}(1 - \hat{p}_{pooled})(\frac{1}{n_1} + \frac{1}{n_2})}} = \frac{0.090 - 0.056}{\sqrt{0.071(1 - 0.071)(\frac{1}{156} + \frac{1}{198})}} = 1.25$$

P-value: $2 \times P(z > 1.25) = 0.2126$



You can also use R:



```
> prop.test(c(14,11),c(156,198),correct=F)
```

2-sample test for equality of proportions without continuity correction

```
data: c(14, 11) out of c(156, 198)
X-squared = 1.5538, df = 1, p-value = 0.2126
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.02085337  0.08922944
sample estimates:
 prop 1      prop 2
 0.08974359 0.05555556
```



Chi-square test for one-way tables

Chi-square test can be used to tests about whether your data is as expected.

Observed	O_1	O_2	\cdots	O_k
Expected	E_1	E_2	\cdots	E_k

We are to evaluate whether there is convincing evidence that a set of observed counts O_1, O_2, \dots, O_k in k categories are unusually different from what might the expected counts E_1, E_2, \dots, E_k .

Conditions for the Chi-square test:

- ▶ Independence. Each case that contributes a count to the table must be independent of all the other cases in the table.
- ▶ Sample size / distribution. Each particular scenario (i.e. cell count) must have at least 5 expected cases.

Chi-square test statistic

- ▶ Test statistics:

$$\chi_0^2 = \frac{(O_1 - E_1)^2}{E_1} + \frac{(O_2 - E_2)^2}{E_2} + \cdots + \frac{(O_k - E_k)^2}{E_k}$$

Note the each term of the test statistics is in the form of

$$\left(\frac{\text{point estimate} - \text{null value}}{\text{SE of point estimate}} \right)^2$$

The standard error for the point estimate of the count in binned data is the square root of the count under the null.

- ▶ P-value:

The test statistic χ_0^2 follows a Chi-square distribution with $k - 1$ degrees of freedom. P-value is found by looking at the **upper tail** of this Chi-square distribution.



Chi-square distribution

The chi-square distribution is always positive and typically right skewed. It has just one parameter called degrees of freedom (df), which influences the shape, center, and spread of the distribution.

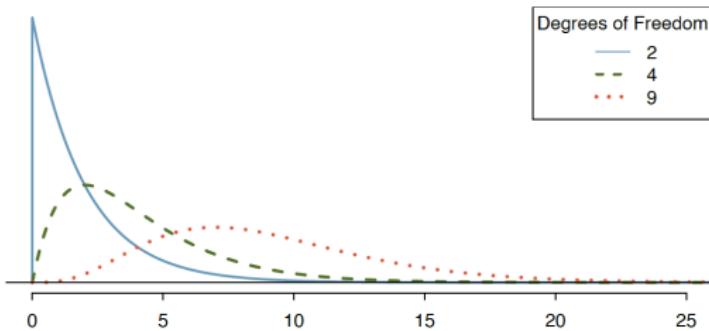


Figure 6.7: Three chi-square distributions with varying degrees of freedom.

In the calculation of p-values, we find the relevant area in the tail of a chi-square distribution. The most common ways to do this are using computer software (R), using a graphing calculator, or using a table (Appendix C.3 in OS).

Chi-square goodness of fit test



We have bags of candy with five flavors in each bag. The bags should contain an equal number of pieces of each flavor. We'd like to test is that the proportions of the five flavors in each bag are the same. We sample 10 bags of candy with 100 pieces in each bag.

Flavors	Apple	Cherry	Grape	Lime	Orange
Number candies observed	180	210	170	220	220
Theoretical probability	0.2	0.2	0.2	0.2	0.2
Expected counts	200	200	200	200	200

The Chi-square goodness of fit test checks whether your sample data is likely to be from a specific theoretical distribution.

H_0 : The proportions of the five flavors in each bag are the same, which is 0.2.

H_1 : The proportions of the five flavors in each bag are not the same.

Test statistic:

$$\chi^2_0 = \frac{(180 - 200)^2}{200} + \frac{(210 - 200)^2}{200} + \frac{(170 - 200)^2}{200} + \frac{(220 - 200)^2}{200} + \frac{(220 - 200)^2}{200} = 11$$

P-value = 0.0266 from chi-square upper tail with df=4.

```
> 1-pchisq(11,df=4)
```

0.02656401



Chi-square test for two-way table

A one-way table describes counts for each outcome in a single variable. A two-way table describes counts for combinations of outcomes for two variables. We can use Chi-square test to see if the two variables are independent.

Here is a sample data presenting the relationship between the type movie watch and whether purchasing snack

	Action	Comedy	Family	Horror	total
snack	50 (65.58)	125 (155)	90 (62)	45 (28.42)	310
no snack	75 (60.42)	175 (145)	30 (58)	10 (26.58)	290
Total	125	300	120	55	600

The expected counts (given in blue) for each Movie-Snack combination assuming movie type and snack purchase are independent is calculated as:

$$\text{expected cell counts} = \frac{\text{row total} \times \text{column total}}{\text{Grand total}}$$

e.g. expected counts of Action movie with snack = $(310 \times 125)/600 = 64.58$

Chi-square test for independence, cont.

	Action	Comedy	Family	Horror	total
snack	50 (65.58)	125 (155)	90 (62)	45 (28.42)	310
no snack	75 (60.42)	175 (145)	30 (58)	10 (26.58)	290
Total	125	300	120	55	600

The chi-square test statistic for a two-way table is found the same way it is found for a one-way table. For two way tables, the degrees of freedom is equal to $(\text{number of rows} - 1) \times (\text{number of columns} - 1)$

H_0 : the type of movie and whether or not people bought snacks are unrelated/independent.

H_1 : the type of movie and whether or not people bought snacks are related/dependent

Test statistic:

$$\chi^2_0 = \frac{(50 - 65.48)^2}{65.48} + \frac{(125 - 155)^2}{155} + \frac{(90 - 62)^2}{62} + \dots + \frac{(10 - 26.58)^2}{26.58} = 65.03$$

P-value = 4.94×10^{-14} from with $df = (4 - 1) \times (2 - 1) = 3$.

```
> 1-pchisq(65.03,df=3)  
4.940492e-14
```



STAT560 Lecture 16: Statistical Inference for Numerical Data

Beidi Qiang

SIUE

Inference on Single Population mean

The population mean μ emerges when the characteristic we measure on each individual is numerical. Some examples:

μ = average number of followers for each individual account on Twitter

μ = average time for all runners who finished a race

We make the following assumptions for each individual in the sample:

1. The sample observations must be independent. The most common way to satisfy this condition is when the sample is a simple random sample from the population.
2. When a sample is small, we also require that the sample observations come from a normally distributed population. We can relax this condition more and more for larger and larger sample sizes ($n > 30$).

Suppose we take a sample of size n from a population with a population mean μ and population standard deviation σ . A natural point estimator for μ , is the **sample mean**, \bar{X}

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n}.$$

- ▶ **Accuracy:** \bar{X} is an unbiased estimator of μ . That is,

$$E(\bar{X}) = \mu.$$

- ▶ **Precision:** The standard error of \bar{X} is

$$SE(\bar{X}) = \sigma / \sqrt{n}.$$

When the assumptions are met, the sampling distribution for \bar{X} is nearly normal. If we standardized \bar{X} , we obtain

$$Z = \frac{\bar{X} - \mu}{\sigma / \sqrt{n}} \sim \mathcal{N}(0, 1)$$

The Issue of Unknown Population Standard Deviation

In practice, we cannot directly calculate the standard error for \bar{X} since we do not know the population standard deviation σ . So we approximate the standard error, using the sample standard deviation s in place of σ .

$$SE(\bar{X}) = \frac{\sigma}{\sqrt{n}} \approx \frac{s}{\sqrt{n}}$$

With the approximation, the sampling distribution changes to

$$t = \frac{\bar{X} - \mu}{S/\sqrt{n}} \sim t(n-1),$$

a **t distribution** with degrees of freedom $df = n - 1$.

t distribution

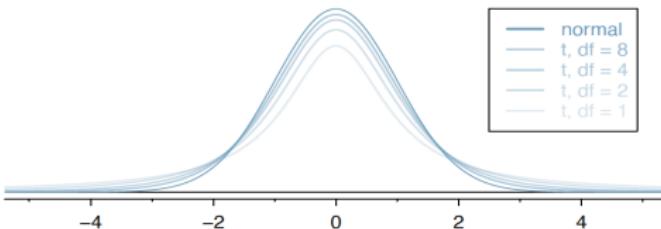


Figure 7.2: The larger the degrees of freedom, the more closely the t -distribution resembles the standard normal distribution.

The t distribution has the following characteristics:

- ▶ It is continuous and symmetric about 0.
- ▶ It is indexed by a single parameter called the degrees of freedom, which describes the shape of the t -distribution.
- ▶ When compared to the standard normal distribution, the t distribution, in general, is less peaked and has more probability (area) in the tails. The larger the degrees of freedom, the more closely the distribution approximates the normal model.
- ▶ In practice, it's common to use statistical software, such as R, for probabilities associated with t distribution. Alternatively, a graphing calculator or a t -table may be used (Appendix C.2 in OS)

Confidence Interval for population mean

- An approximate $100(1 - \alpha)\%$ confidence interval for population mean, μ is

$$\left(\bar{x} - t_{df}^* \times \frac{s}{\sqrt{n}}, \bar{x} + t_{df}^* \times \frac{s}{\sqrt{n}} \right).$$

- Note of the form of the interval is again:
point estimate $\pm t_{df}^* \times$ standard error
- Recall that In the normal model, we used z^* and the standard error to determine the margin of error. Here we revise the confidence interval formula slightly when using t_{df}^* from the t-distribution.
- The choice of t^* corresponds to the confidence level and the degree of freedom $df = n - 1$



When completing a hypothesis test for the one-sample mean, the process is nearly identical to completing a hypothesis test for a single proportion we used previously:

1. State the **null** and **alternative** hypotheses. These are statements about μ . 
2. Collect the data and summarize the evidence by calculate **p-value** assuming null is true. We now calculate p-value base on **t-distribution**. 
3. Draw conclusion based on the **p-value**. We either **reject null or fail to reject null**.

the smaller the p-value \implies the more evidence against H_0

Calculate p-value for a One-sample t-test

Recall, previously we find the Z-score using the observed value, null value, and standard error, then we find the p-value using the standard normal distribution. Now, we calculate a T-score instead and use a t-distribution for calculating the tail area.

- ▶ assume $H_0 : \mu = \mu_0$ is true, i.e., the population mean is μ_0
- ▶ Test statistics is the sample mean standardized by its mean and standard error.

$$t_0 = \frac{\text{point estimate} - \text{null value}}{\text{SE of point estimate}} = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$



- ▶ P-value is probability of observing such an extreme sample mean.

Alternative hypothesis	Hypothesis type	p-value formula
$H_1 : \mu < \mu_0$	Left-tail hypothesis	$P(t < t_0)$
$H_1 : \mu > \mu_0$	Right-tail hypothesis	$P(t > t_0)$
$H_1 : \mu \neq \mu_0$	Two-tail hypothesis	$2P(t < - t_0)$

Example:

The average time for all runners who finished the Cherry Blossom Race in 2006 was 93.29 minutes. We want to determine using data from 100 participants in the 2017 Cherry Blossom Race whether runners in this race are getting slower or faster versus the other possibility that there has been no change. The sample mean and sample standard deviation of the sample of 100 runners from the 2017 Cherry Blossom Race are 97.32 and 16.98 minutes, respectively.



Inference for μ : Confidence Interval Approach

Recall that a $100(1 - \alpha)\%$ C.I. is

$$\left(\bar{x} - t_{n-1}^* \frac{s}{\sqrt{n}}, \bar{x} + t_{n-1}^* \frac{s}{\sqrt{n}} \right).$$

- ▶ The point estimate $\bar{x} = 97.32$.
- ▶ $df = n - 1 = 99$, $t_{99}^* = 1.984$ for 95% confidence level.
- ▶ Standard error: $s/\sqrt{n} = 16.98/\sqrt{100} = 1.698$
- ▶ 95% CI is: $(93.591, 100.689)$
- ▶ We are 95% confident, the average time for runners from the 2017 Cherry Blossom Race to finish is between 93.591 minutes and 100.689 minutes.



Use R to get the t^* value:

```
> qt(0.025, df=99)  
-1.984217
```



Inference for μ : one sample t-test

- ▶ Step 1: State H_0 and H_1



$$H_0 : \mu = 93.29$$

$$H_1 : \mu \neq 93.29$$

- ▶ Step 2: Calculate test statistic and p-value assuming H_0 is true

$$t_0 = \frac{\bar{x} - \mu_0}{s/\sqrt{n}} = \frac{97.32 - 93.29}{16.98/\sqrt{100}} = 2.367$$

$$\text{p-value} = 2 * P(t < -2.367) \approx 0.02.$$



- ▶ Step 3: Draw the conclusion

$\alpha = 0.05$, p-value is smaller than 0.05. We reject H_0 , and conclude that the average time for runners in 2017 Cherry Blossom Race is significantly different from 93.39 minutes, the average time in 2006

Use R to get the p-value from t-distribution:

```
> pt(-2.367,df=99)  
0.00993908
```

Two sets of observations are paired if each observation in one set has a special correspondence or connection with exactly one observation in the other data set.

We would like to extend the methods to apply confidence intervals and hypothesis tests to the differences.

For example, we can compare

1. the selling prices of textbooks in the Bookstore and on Amazon.
 2. pre- (beginning of semester) and post-(end of semester) math placement test scores of students.
-
- Assumptions:
1. The data are independent within group. The data are not independent between the two groups.
 2. Distribution of population data is approximately normal or sample size is sufficiently large.

CI and H.test for paired data

To analyze paired data, it is often useful to look at the difference in outcomes of each pair of observations. We then simply analyze the differences.

- ▶ Confidence interval:

$$\bar{x}_{\text{diff}} \pm t_{df}^* \times \frac{s_{\text{diff}}}{\sqrt{n}}$$

- ▶ Hypothesis test:

Hypothesis:

$$H_0 : \mu_{\text{diff}} = \mu_1 - \mu_2 = 0, H_1 : \mu_{\text{diff}} >, <, \text{ or } \neq 0$$

The test statistic under H_0 :

$$t_0 = \frac{\bar{x}_{\text{diff}} - 0}{s_{\text{diff}} / \sqrt{n}},$$



P-value: same as before.

Example:

We want to see if Amazon prices were, on average, lower than the books from the Bookstore. A sample 68 books are collected. A portion of the data set is shown below:

book	Bookstore Price	Amazon Price	Price difference
1	47.97	47.45	0.52
2	14.26	13.55	0.71
:	:	:	:
68	35.96	32.40	3.56

- We first calculate the price difference for each book (show in the table above).
- The sample mean of the price difference is 3.56 and sample standard deviation of the price difference is 13.42.
- 95% CI:

$$3.56 \pm 1.996 \times \frac{13.42}{\sqrt{68}} = (0.332, 6.828)$$

- Hypothesis test:

$$H_0: \mu_{diff} = \text{price difference} = 0; H_1: \mu_{diff} \neq 0$$

Test statistic:

$$t_0 = \frac{3.56}{13.42/\sqrt{68}} = 2.19, df = 67$$

P-value: $2 \times P(t < -2.19) = 0.032$

We may also wish to compare the mean from two different populations under the condition that the data are not paired. For example, we may wish to compare

- ▶ Average starting salaries of male and female engineers
- ▶ Mean weight of newborns from mothers who smoke v.s. the mean weight of newborns from mothers who don't smoke



Assumptions:

1. Independence. The data are independent within and between the two groups. e.g. the data come from independent random samples.
2. Normality. We check the outliers and rules of thumb for each group separately.

Inference on $\mu_1 - \mu_2$

- ▶ Point estimator for $\mu_1 - \mu_2$, is the difference in sample means, $\bar{X}_1 - \bar{X}_2$.
- ▶ When the assumptions are met, the sampling distribution for $\bar{X}_1 - \bar{X}_2$ is normal with mean $\mu_1 - \mu_2$ and standard error



$$SE(\bar{X}_1 - \bar{X}_2) = \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}.$$

- ▶ Note the construction of standard error depend on population standard deviation σ_1 and σ_2 , which are unknown. We estimate those using the sample standard deviations s_1 and s_2 .

$$T = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \sim t, \text{ with } df = \frac{(s_1^2/n_1 + s_2^2/n_2)^2}{\frac{(s_1^2/n_1)^2}{n_1-1} + \frac{(s_2^2/n_2)^2}{n_2-1}}.$$

This degree of freedom approximation above is called Welch–Satterthwaite equation, which is the commonly used one in Welch two sample t-test. Your textbook (OS) used the the smaller of $n_1 - 1$ and $n_2 - 1$ as the degrees of freedom instead.

Two-sample t-test

- Degree of freedom approximation

$$df = \frac{(\bar{s}_1^2/n_1 + \bar{s}_2^2/n_2)^2}{\frac{(\bar{s}_1^2/n_1)^2}{n_1-1} + \frac{(\bar{s}_2^2/n_2)^2}{n_2-1}}$$

- Confidence Interval

An approximate $(1 - \alpha)100\%$ confidence interval $\mu_1 - \mu_2$ is given by

$$(\bar{x}_1 - \bar{x}_2) \pm t_{df}^* \times \sqrt{\frac{\bar{s}_1^2}{n_1} + \frac{\bar{s}_2^2}{n_2}}.$$

- Hypothesis Test

$H_0 : \mu_1 - \mu_2 = 0$, $H_1 : \mu_1 - \mu_2 >, <, \text{ or } \neq 0$

The test statistic:

$$t_0 = \frac{\bar{x}_1 - \bar{x}_2 - 0}{\sqrt{\frac{\bar{s}_1^2}{n_1} + \frac{\bar{s}_2^2}{n_2}}}$$



P-value: same as before.

Example: Recycling Project

You are part of a recycling project that is examining how much paper is being discarded (not recycled) by employees at two large plants. These data are obtained on the amount of white paper thrown out per year by employees (data are in hundreds of pounds). Samples of employees at each plant were randomly selected. We'd like to see if there is a difference in the average amount of paper wasted.



Plant 1 ($n_1 = 25$):	3.01	2.58	3.04	1.75	2.87	2.75	2.51	2.93	2.85	3.09
	1.43	3.36	3.18	2.74	2.25	1.95	3.68	2.29	1.86	2.63
	2.83	2.04	2.23	1.92	3.02					
Plant 2 ($n_2 = 20$):	3.79	2.08	3.66	1.53	4.07	4.31	2.62	4.52	3.80	5.30
	3.41	0.82	3.03	1.95	6.45	1.86	1.87	3.78	2.74	3.81

Example: Recycling Project Cont'd

- ▶ $\bar{x}_1 = 2.58, \bar{x}_2 = 3.27, s_1 = 0.55, s_2 = 1.37$



$$df = \frac{\left(\frac{0.55^2}{25} + \frac{1.37^2}{20}\right)^2}{\frac{(0.55^2/25)^2}{24} + \frac{(1.37^2/20)^2}{19}} \approx 23$$

- ▶ 90% CI:

$$(2.58 - 3.27) \pm 1.71 \times \sqrt{\frac{0.55^2}{25} + \frac{1.37^2}{20}} = (-1.25, -0.13)$$

> qt(0.05, df=23)
-1.713872



- ▶ Hypothesis test:

$$H_0: \mu_1 - \mu_2 = 0; H_1: \mu_1 - \mu_2 \neq 0$$

Test statistic:



$$t_0 = \frac{2.58 - 3.27}{\sqrt{\frac{0.55^2}{25} + \frac{1.37^2}{20}}} = -2.12$$

P-value: $2 \times P(t < -2.12) = 0.045$

> pt(-2.12, df=23)
0.02250455

Example: Recycling Project using R

You may also use R to do the two sample t-test directly:

```
> plant.1 = c(3.01,2.58,3.04,1.75,2.87,2.57,2.51,2.93,2.85,3.09,  
+ 1.43,3.36,3.18,2.74,2.25,1.95,3.68,2.29,1.86,2.63,  
+ 2.83,2.04,2.23,1.92,3.02)  
> plant.2 = c(3.79,2.08,3.66,1.53,4.07,4.31,2.62,4.52,3.80,5.30,  
+ 3.41,0.82,3.03,1.95,6.45,1.86,1.87,3.78,2.74,3.81)  
> t.test(plant.1,plant.2,conf.level=0.95,var.equal=FALSE)
```

Welch Two Sample t-test



```
data: plant.1 and plant.2  
t = -2.1037, df = 23.972, p-value = 0.04608  
alternative hypothesis: true difference in means is not equal to 0  
95 percent confidence interval:  
 -1.35825799 -0.01294201  
sample estimates:  
mean of x mean of y  
 2.5844    3.2700
```

p-value = 0.04608, reject H_0 at 0.05 level. We have sufficient evidence to conclude the average amount of paper waisted in the two plant are different.

Try the following R code for one-side tests:

```
t.test(plant.1,plant.2,conf.level=0.95,var.equal=FALSE,alternative="less")  
t.test(plant.1,plant.2,conf.level=0.95,var.equal=FALSE,alternative="greater")
```

Two-sample t-test with pooled variance

Occasionally, two populations will have standard deviations that are so similar that they can be treated as identical. The pooled standard deviation of two groups is a way to use data from both samples to better estimate the standard error.

- ▶ Pooled variance:

$$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}.$$

- ▶ Confidence Interval for $\mu_1 - \mu_2$:

$$(\bar{x}_1 - \bar{x}_2) \pm t_{df}^* \times \sqrt{s_p^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}, \text{ where, } df = n_1 + n_2 - 2.$$

- ▶ Hypothesis Test

$$H_0 : \mu_1 - \mu_2 = 0, H_1 : \mu_1 - \mu_2 >, <, \text{ or } \neq 0$$

The test statistic:

$$t_0 = \frac{\bar{x}_1 - \bar{x}_2 - 0}{\sqrt{s_p^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}}$$

P-value: same as before.

Comparing many means with ANOVA

We've saw how to compare two population means. In the next we will learn to compare three or more population means at the same time, which is often of interest in practical applications. We will learn a new method called analysis of variance (ANOVA) and a new test statistic called F. We observe data from k groups:

	Group 1	Group 2	...	Group k	Total
population mean (unknown)	μ_1	μ_2	...	μ_k	μ
sample size (n_i)	n_1	n_2	...	n_k	n
sample mean (\bar{x}_i)	\bar{x}_1	\bar{x}_2	...	\bar{x}_k	\bar{x}
sample std. dev (s_i)	s_1	s_2	...	s_k	s

Conditions for the ANOVA test:

- ▶ the observations are independent within and across groups,
- ▶ the data within each group are nearly normal,
- ▶ the variability across the groups is about equal.

ANOVA F-Test

- Hypothesis:



$H_0 : \mu_1 = \mu_2 = \dots = \mu_k$. The mean is the same across all groups.

H_1 : At least one mean is different.

- Test statistic:



mean square between groups (MSG)



$$MSG = \frac{1}{k-1} \sum_{i=1}^k n_i (\bar{x}_i - \bar{x})^2$$

mean square error (MSE)

$$MSE = \frac{\sum_{i=1}^k (n_i - 1) s_i^2}{n - k}$$

Test statistic:

$$F_0 = \frac{MSG}{MSE}$$

F distribution

F statistic represents a standardized ratio of variability in the sample means relative to the variability within the groups. It follows an F distribution, which has two associated parameters: df_1 and df_2 .

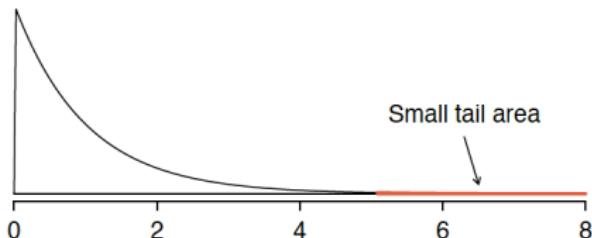


Figure 7.24: An F distribution with $df_1 = 2$ and $df_2 = 426$.

If H_0 is true and the model conditions are satisfied, the statistic F follows an F distribution with parameters $df_1 = k - 1$ and $df_2 = n - k$.



In the calculation of p-values, we find the relevant area in the tail of a F distribution.

The most common ways to do this are using computer software (R).

One-way ANOVA Example:

The average of grade point averages (GPAs) of college courses in a specific major is a measure of difficulty of the major. An educator wishes to conduct a study to find out whether the difficulty levels of different majors are the same. A random sample of major grade point averages (GPA) of 11 graduating seniors at a large university is selected for each of the four majors.



Major	Math	English	Education	biology	Total
population mean	μ_1	μ_2	μ_3	μ_4	μ
sample size	11	11	11	11	44
sample mean	2.90	3.34	3.36	3.02	3.153
sample std. dev	0.434	0.384	0.478	0.396	0.456

$$H_0: \mu_1 = \mu_2 = \mu_3 = \mu_4$$

$H_1:$ not all four populations are equal.



Test statistic:



$$MSG = \frac{11(2.90 - 3.15)^2 + 11(3.34 - 3.15)^2 + 11(3.36 - 3.15)^2 + 11(3.02 - 3.15)^2}{4 - 1} = 0.585$$

$$MSE = \frac{(11 - 1)0.434^2 + (11 - 1)0.384^2 + (11 - 1)0.478^2 + (11 - 1)0.456^2}{44 - 4} = 0.181$$

$$F = MSG / MSE = 3.232, \text{ P-value} = 0.0322 \text{ from F distribution with } df1 = 3, df2 = 40.$$

To get the F tail area in R:

```
> 1-pf(3.232, 3, 40)  
0.0322
```



Multiple Testing



When we reject the null hypothesis in an ANOVA analysis, we might wonder, which of these groups have different means. To answer this question, we compare the means of each possible pair of groups using 2-sample t-tests.

The scenario of testing many pairs of groups is called multiple comparisons. The Bonferroni correction suggests that a more stringent significance level is more appropriate in order to control family-wise error rate (FWER), that is, the probability of making at least one type I error.

$$\alpha^* = \alpha/K,$$

where K is the number of comparisons being considered.

Remark: With respect to FWER control, the Bonferroni correction can be conservative. There are many other method in context of multiple testing, such as Scheffé's method, Tukey's procedure, Hochberg's step-up procedure, etc.

In previous example, we can perform the 6 possible pairwise comparisons using the Bonferroni correction ($\alpha^* = 0.05/6 = 0.0083$)

e.g. $H_0 : \mu_1 = \mu_2$; $H_1 : \mu_1 \neq \mu_2$

t-test statistic (pooled variance): $t_0 = -2.477$ with $df = 20$

p-value = 0.0222 > 0.0083

Fail to reject H_0 .

STAT560 - Foundations of Data Science - Lecture 17

Statistical inference using R

In this lecture we will explore inferential statistics using functions in R. We will start with sampling distribution and central limit theorem, then we discuss confidence interval and hypothesis testing.

Sampling Distribution

For illustration purpose, we will use a flight dataset in the nycflights13 package. The data contains information of 336,776 flights in 2013 from NYC airport. We treat this collection as our population of interest. In this example, we assume have access to the entire population, but this is rarely the case in real life. Gathering information on an entire population is often extremely costly or impossible. Because of this, we often take a sample of the population having the entire population of interest can help us understand the relation between population, samples, and sampling distribution.

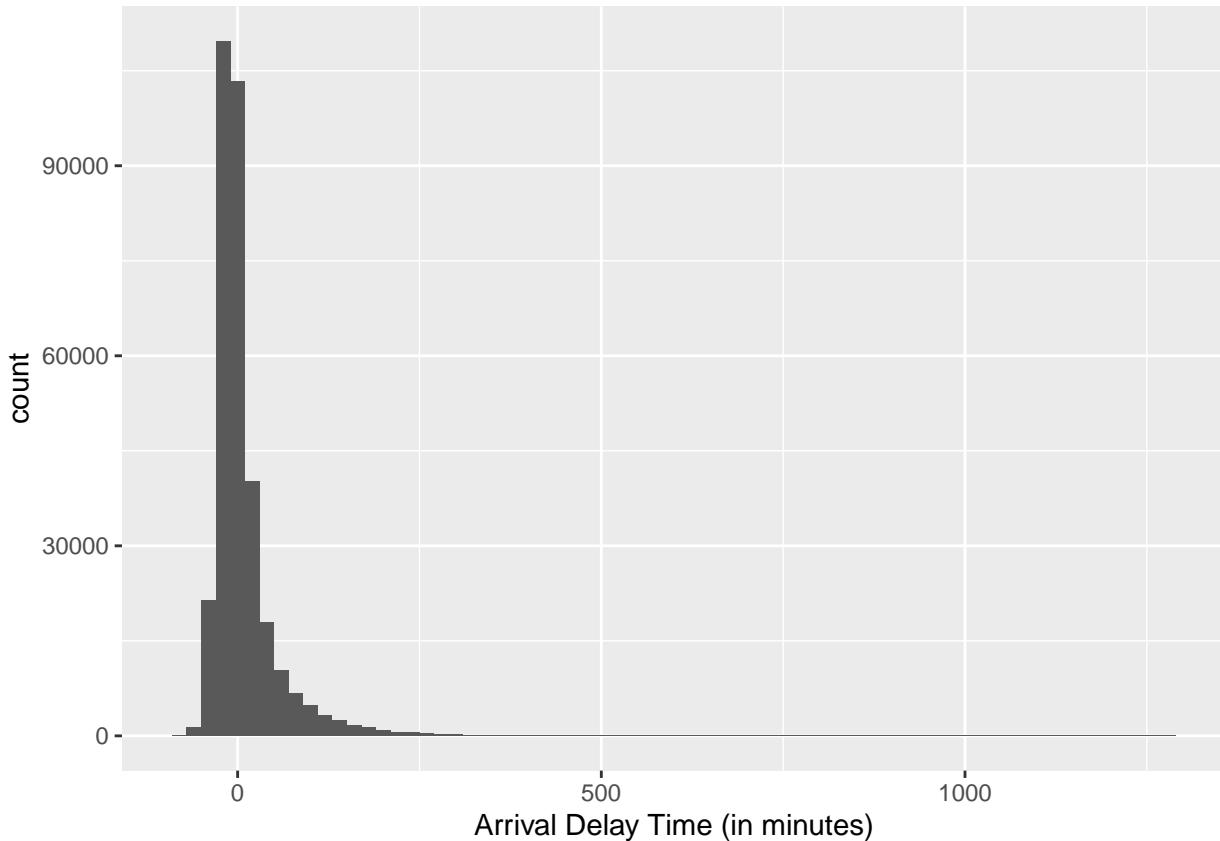
```
library(tidyverse)
library(nycflights13)
```

Let's look at the distribution of arrival delay times (arr_delay) in our population of flights by calculating a few summary statistics and making a histogram. The histogram of the variable shows a positive (right) skew of the population data.

```
delay=flights %>% filter(!is.na(arr_delay)) #remove the missing values (NAs)
summary(delay$arr_delay)

##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
## -86.000  -17.000   -5.000    6.895   14.000 1272.000

ggplot(data = delay, aes(x = arr_delay)) + geom_histogram(binwidth = 20) + labs(x = "Arrival Delay Time")
```



Let's take a sample of size $n=25$ from the population. You can use the `slice_sample()` function in `dplyr` or `sample()` function in R base.

```
delay25 = delay %>% slice_sample(n = 25)
summary(delay25$arr_delay)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    -51.0   -25.0  -15.0     -6.6    6.0     56.0

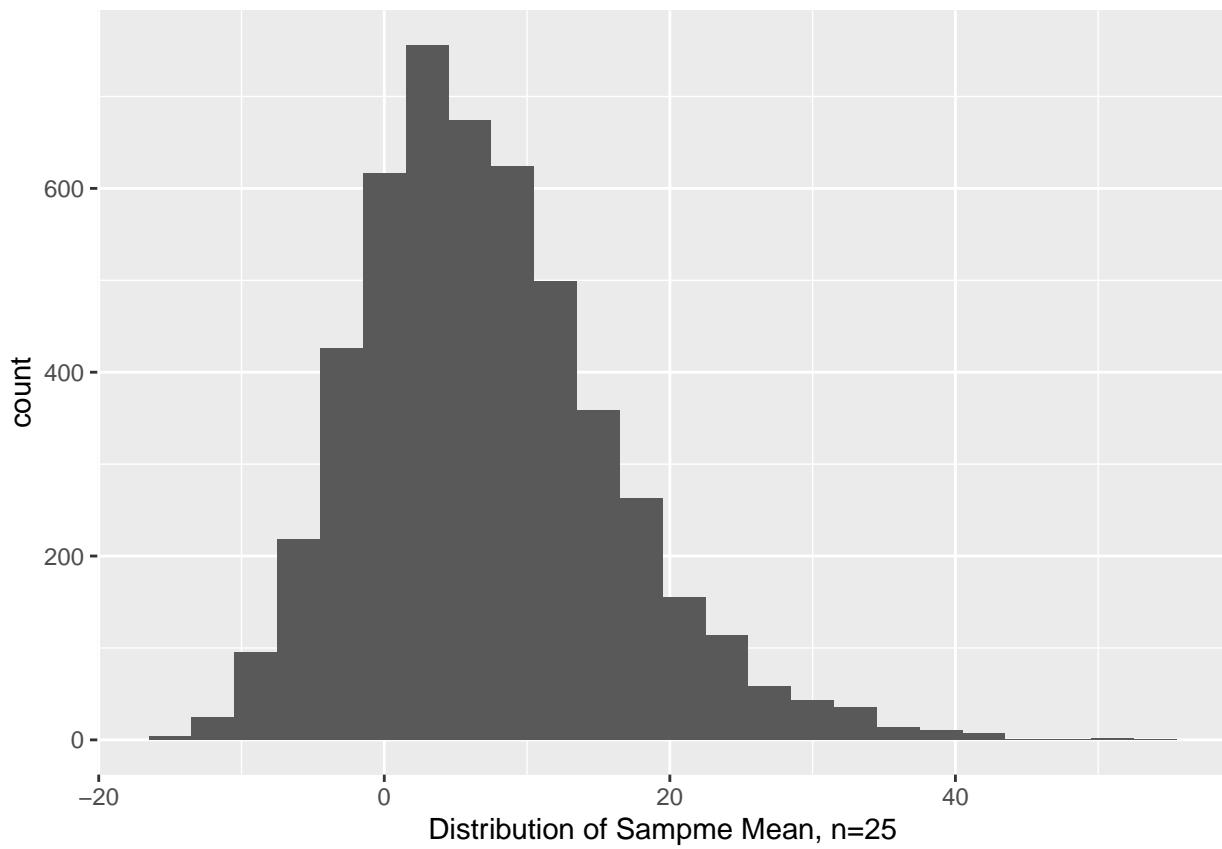
d25=sample(delay$arr_delay,25)
summary(d25)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    -44.0   -16.0   -2.0      1.2    14.0     99.0
```

Notice every time we take another random sample, we get a different sample mean. Here we will generate 5000 samples of size 25 from the population, calculate the mean of each sample, and store each result in a vector called `mean25`. We will then plot the histogram of this sampling distribution. You may do this using a for-loop for the `map()` function

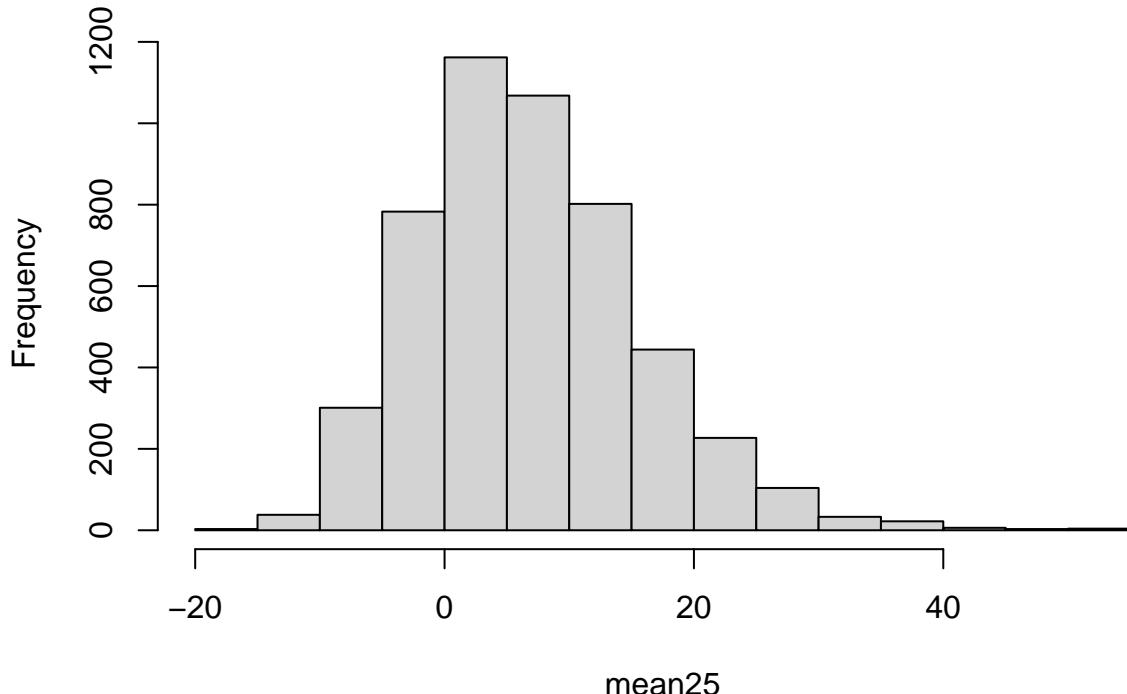
```
library(purrr)
num_trials = 5000
mean25 = 1:num_trials %>% map_dfr(~delay %>%
  slice_sample(n = 25) %>%
  summarize(mean_delay = mean(arr_delay)))
)

ggplot(data=mean25, aes(x = mean_delay)) + geom_histogram(binwidth = 3) + labs(x = "Distribution of Sampl
```



```
mean25=vector(length=5000)
for(i in 1:5000){
  samp = sample(delay$arr_delay, 25)
  mean25[i] = mean(samp)
}
hist(mean25,main="Distribution of Sampme Mean, n=25")
```

Distribution of Sample Mean, n=25



```
summary(mean25)
```

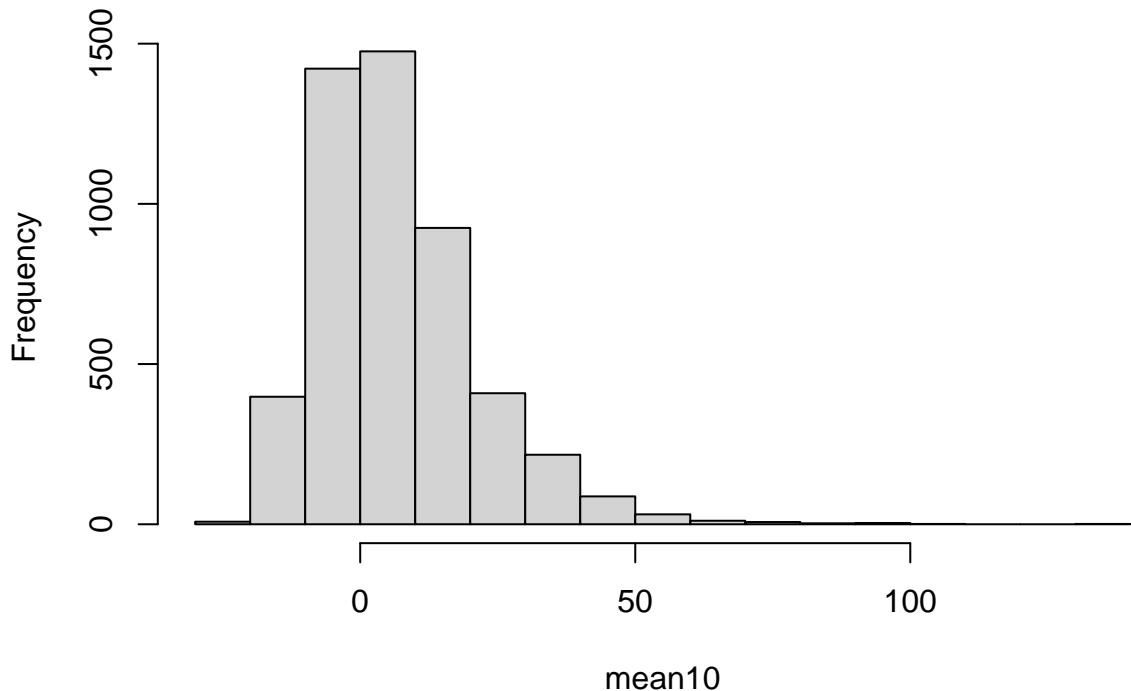
```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## -16.160    0.640   6.000   6.917  12.090  54.360
```

Central Limit Theorem

The sampling distribution that we computed tells us much about estimating the average delay time of flights. Because the sample mean is an unbiased estimator, the sampling distribution is centred at the true average of the population, and the spread of the distribution indicates how much variability is induced by sampling. To get a sense of the effect that sample size has on our distribution, let's build up some more sampling distributions: based on a sample size of 10, 100 and 500.

```
mean10=vector(length=5000)
for(i in 1:5000){
  samp = sample(delay$arr_delay, 10)
  mean10[i] = mean(samp)
}
hist(mean10,main="Distribution of Sample Mean, n=10")
```

Distribution of Sample Mean, n=10

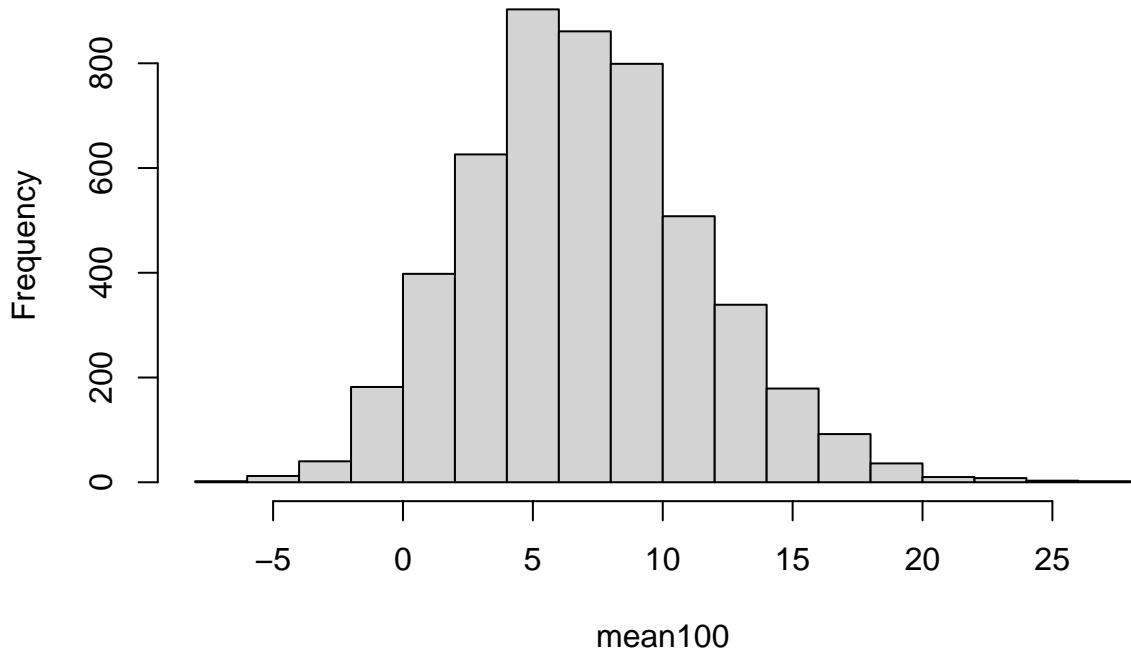


```
summary(mean10)

##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.
## -25.400 -3.400  4.300   6.765 14.200 134.800

mean100=vector(length=5000)
for(i in 1:5000){
  samp = sample(delay$arr_delay, 100)
  mean100[i] = mean(samp)
}
hist(mean100,main="Distribution of Sample Mean, n=100")
```

Distribution of Sample Mean, n=100

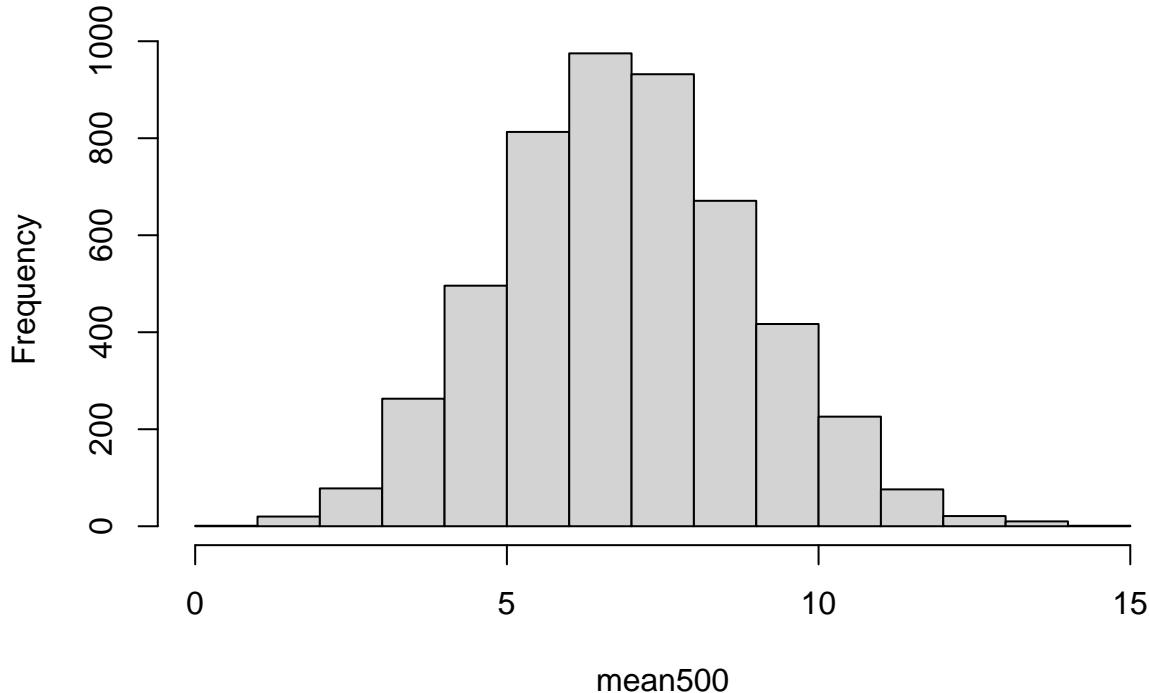


```
summary(mean100)

##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.
## -6.740   3.980   6.780   7.017   9.760  27.890

mean500=vector(length=5000)
for(i in 1:5000){
  samp = sample(delay$arr_delay, 500)
  mean500[i] = mean(samp)
}
hist(mean500,main="Distribution of Sample Mean, n=500")
```

Distribution of Sample Mean, n=500



```
summary(mean500)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##  0.798   5.488   6.835   6.895   8.202  14.242
```

One-sample Inference for population proportion (categorical variable)

In the next a few topics, we are going to see how to perform inference (confidence interval and hypothesis tests) using R. We are going to illustrate use full loan data, which is the complete version of the loan50 data we used previously.

```
myData=read.csv(file="https://www.openintro.org/data/csv/loans_full_schema.csv",header=T)
```

You may perform one-sample test for proportion in R using prop.test() function in base package. The output will also give a confidence interval. You will need to table the counts in the categories before using the function.

```
table(myData$application_type)
```

```
##
## individual      joint
##     8505       1495
x=table(myData$application_type)[ "joint"]
n=sum(table(myData$application_type))
# default two-sided test and two-sided confidence interval
prop.test(x,n,p=0.2,correct=F)

##
## 1-sample proportions test without continuity correction
##
## data: x out of n, null probability 0.2
```

```

## X-squared = 159.39, df = 1, p-value < 2.2e-16
## alternative hypothesis: true p is not equal to 0.2
## 95 percent confidence interval:
## 0.1426458 0.1566234
## sample estimates:
##      p
## 0.1495

#left one-tailed test and one side confidence limit
prop.test(x,n,p=0.2,alternative = "less",correct=F)

##
## 1-sample proportions test without continuity correction
##
## data: x out of n, null probability 0.2
## X-squared = 159.39, df = 1, p-value < 2.2e-16
## alternative hypothesis: true p is less than 0.2
## 95 percent confidence interval:
## 0.00000 0.15546
## sample estimates:
##      p
## 0.1495

# change confidence level
prop.test(x,n,p=0.2,correct=F,conf.level=0.9)

##
## 1-sample proportions test without continuity correction
##
## data: x out of n, null probability 0.2
## X-squared = 159.39, df = 1, p-value < 2.2e-16
## alternative hypothesis: true p is not equal to 0.2
## 90 percent confidence interval:
## 0.1437296 0.1554600
## sample estimates:
##      p
## 0.1495

```

Two-sample Inference for population proportion (Categorical variable)

You may also perform Two-sample test for proportion in R using prop.test() function in base package. The output will also give a confidence interval. You will again need to table the counts in the categories before using the function.

```

tab_counts=myData %>% group_by(application_type,homeownership) %>%
  summarise(N = n()) %>% spread(homeownership, N) %>% mutate(TOTAL=MORTGAGE+OWN+RENT)

## `summarise()` has grouped output by 'application_type'. You can override using
## the `.groups` argument.

tab_counts

## # A tibble: 2 x 5
## # Groups:   application_type [2]
##   application_type MORTGAGE OWN  RENT TOTAL
##   <chr>           <int> <int> <int> <int>
## 1 individual       3839  1170  3496  8505

```

```

## 2 joint          950   183   362  1495
# default two-sided test for difference in proportions
prop.test(tab_counts$MORTGAGE,tab_counts$TOTAL,correct=F)

##
## 2-sample test for equality of proportions without continuity correction
##
## data: tab_counts$MORTGAGE out of tab_counts$TOTAL
## X-squared = 172.63, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
## -0.2106611 -0.1574788
## sample estimates:
## prop 1   prop 2
## 0.4513815 0.6354515

#left one-tailed test for p1<p2
prop.test(tab_counts$MORTGAGE,tab_counts$TOTAL,alternative = "less",correct=F)

##
## 2-sample test for equality of proportions without continuity correction
##
## data: tab_counts$MORTGAGE out of tab_counts$TOTAL
## X-squared = 172.63, df = 1, p-value < 2.2e-16
## alternative hypothesis: less
## 95 percent confidence interval:
## -1.0000000 -0.1617539
## sample estimates:
## prop 1   prop 2
## 0.4513815 0.6354515

```

Chi-squared Tests

You may perform chi-squared test on one-way or two-way table using function chisq.test() in R base package.

Goodness of fit test. If you have more than 2 categories (data is not binary), you may want to perform a goodness of fit test using Chi-squares.

```

table(myData$homeownership)

##
## MORTGAGE      OWN      RENT
##    4789     1353     3858

chisq.test(table(myData$homeownership),p=c(0.4, 0.2, 0.4),correct=F)

```

```

##
## Chi-squared test for given probabilities
##
## data: table(myData$homeownership)
## X-squared = 369.98, df = 2, p-value < 2.2e-16

```

Test for independence:

```

tab_counts

## # A tibble: 2 x 5
## # Groups:   application_type [2]

```

```

##   application_type MORTGAGE    OWN    RENT TOTAL
##   <chr>           <int> <int> <int>
## 1 individual        3839   1170  3496  8505
## 2 joint             950    183   362   1495
chisq.test(myData$application_type,myData$homeownership,correct=F)

##
## Pearson's Chi-squared test
##
## data: myData$application_type and myData$homeownership
## X-squared = 186.15, df = 2, p-value < 2.2e-16

```

T-test for inference on population means (numerical data)

To perform inference with numerical data, we can use `t.test()` function in R base package to perform t-tests on population means.

One-sample t-test:

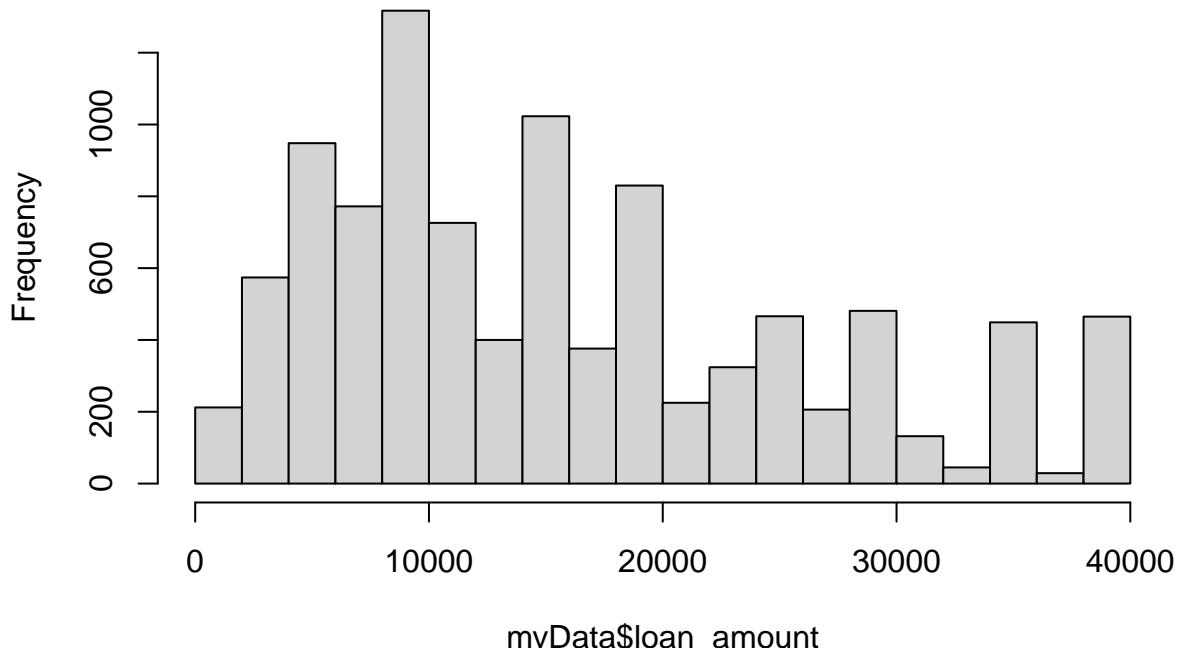
```
summary(myData$loan_amount)
```

```

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##      1000    8000   14500    16362   24000    40000
hist(myData$loan_amount)

```

Histogram of myData\$loan_amount



```
t.test(myData$loan_amount, mu=15000, conf.level=0.90)
```

```

##
## One Sample t-test
##
## data: myData$loan_amount

```

```

## t = 13.22, df = 9999, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 15000
## 90 percent confidence interval:
## 16192.45 16531.39
## sample estimates:
## mean of x
## 16361.92
t.test(myData$loan_amount, mu=15000, conf.level=0.90, alternative="greater")

```

```

##
## One Sample t-test
##
## data: myData$loan_amount
## t = 13.22, df = 9999, p-value < 2.2e-16
## alternative hypothesis: true mean is greater than 15000
## 90 percent confidence interval:
## 16229.89      Inf
## sample estimates:
## mean of x
## 16361.92

```

Two-sample t-test (independent sample): we perform a t-test to see if average loan amount for joint applications are higher than individual applications.

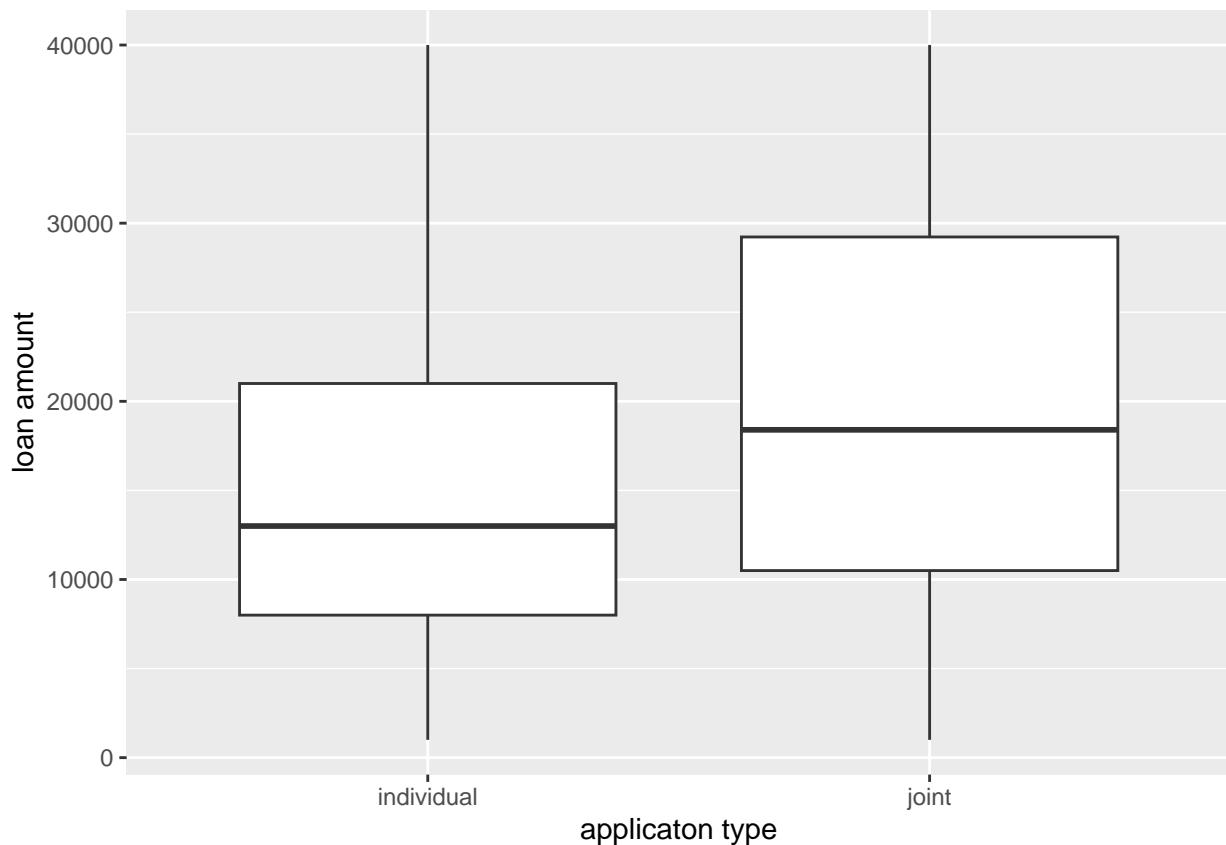
```

myData %>% group_by(application_type) %>% summarise(
  count = n(),
  mean = mean(loan_amount, na.rm = TRUE),
  sd = sd(loan_amount, na.rm = TRUE)
)

## # A tibble: 2 x 4
##   application_type count    mean     sd
##   <chr>          <int>  <dbl>   <dbl>
## 1 individual      8505 15749. 10092.
## 2 joint           1495 19850. 10784.

ggplot(data = myData, aes(x = application_type, y = loan_amount)) +
  geom_boxplot() +
  xlab("applicaton type") +
  ylab("loan amount")

```



```
#Welch two sample t-test
t.test(loan_amount~application_type,alternative="greater",data=myData)

##
## Welch Two Sample t-test
##
## data: loan_amount by application_type
## t = -13.687, df = 1981.1, p-value = 1
## alternative hypothesis: true difference in means between group individual and group joint is greater
## 95 percent confidence interval:
## -4593.91      Inf
## sample estimates:
## mean in group individual      mean in group joint
##           15748.84                  19849.70

#two sample t-test with pooled variance
t.test(loan_amount~application_type,alternative="greater", data=myData,var.equal=T)

##
## Two Sample t-test
##
## data: loan_amount by application_type
## t = -14.339, df = 9998, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group individual and group joint is not equal
## 95 percent confidence interval:
## -4661.469 -3540.240
## sample estimates:
## mean in group individual      mean in group joint
```

```
##                      15748.84                      19849.70
```

We can use a F-test to test for homogeneity in variances and decide if we'd like to use a pooled variance formulation.

```
var.test(loan_amount~application_type, data = myData)
```

```
##  
##  F test to compare two variances  
##  
## data: loan_amount by application_type  
## F = 0.87564, num df = 8504, denom df = 1494, p-value = 0.0006528  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
##  0.8092419 0.9454460  
## sample estimates:  
## ratio of variances  
##                      0.8756374
```

Paired t-test: we wish to test whether average loan amount equals to the average debt amount for applicants applying the loan for debt consolidation purpose.

```
d1=myData %>% filter(loan_purpose=="debt_consolidation") %>% mutate(debt_amount=annual_income*debt_to_i  
t.test(d1$loan_amount, d1$debt_amount, paired=T)
```

```
##  
##  Paired t-test  
##  
## data: d1$loan_amount and d1$debt_amount  
## t = 21.629, df = 5131, p-value < 2.2e-16  
## alternative hypothesis: true mean difference is not equal to 0  
## 95 percent confidence interval:  
##  3082.523 3697.010  
## sample estimates:  
## mean difference  
##                      3389.767
```

ANOVA F-test

We use ANOVA F-test to test difference in three or more group means. You may apply the test using aov() function in R base package. Here we perform ANOVA test to see if the average loan amount for loans in different grades are the same.

```
myData %>% group_by(grade) %>% summarise(  
  count = n(),  
  mean = mean(loan_amount, na.rm = TRUE),  
  sd = sd(loan_amount, na.rm = TRUE)  
)
```

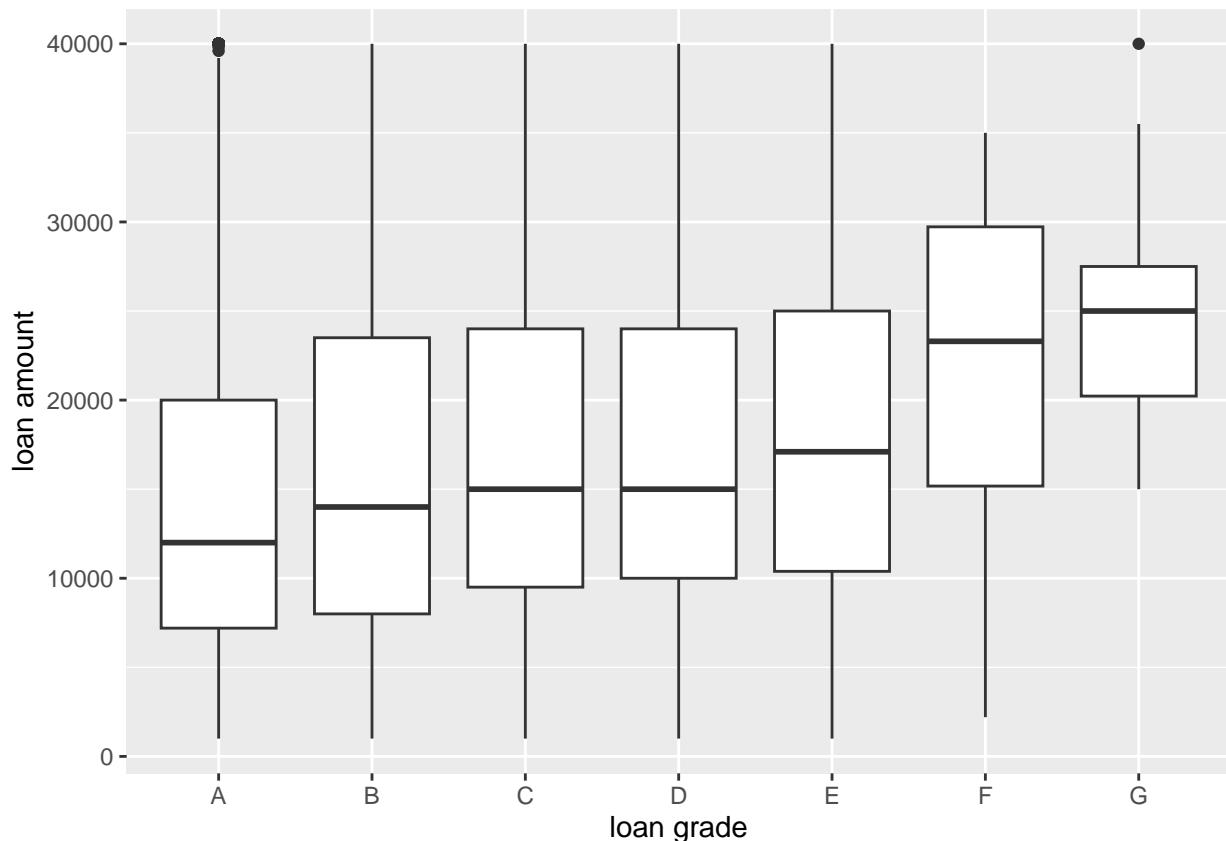
```
## # A tibble: 7 x 4  
##   grade count    mean     sd  
##   <chr> <int>   <dbl>   <dbl>  
## 1 A      2459 15400. 10548.  
## 2 B      3037 16251. 10587.  
## 3 C      2653 16841. 10192.  
## 4 D      1446 16614.  9531.  
## 5 E      335  18261.  9345.
```

```

## 6 F      58 21923.  8236.
## 7 G     12 25429.  7619.

ggplot(data = myData, aes(x = grade, y = loan_amount)) +
  geom_boxplot() +
  xlab("loan grade") +
  ylab("loan amount")

```



```

summary(aov(loan_amount~grade,data=myData))

##          Df  Sum Sq  Mean Sq F value    Pr(>F)
## grade       6 7.003e+09 1.167e+09   11.06 2.48e-12 ***
## Residuals  9993 1.054e+12 1.055e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Aside from the base functions, you may also use a function called `inference()`, in “`statsr`” package to perform inference. This function will allow us to conduct hypothesis tests and construct confidence intervals of various types.

```

library(statsr)
inference(y = loan_amount, x = grade, data = myData,
          statistic = c("mean"),
          type = c("ht"),
          alternative = c("greater"),
          method = c("theoretical"))
)

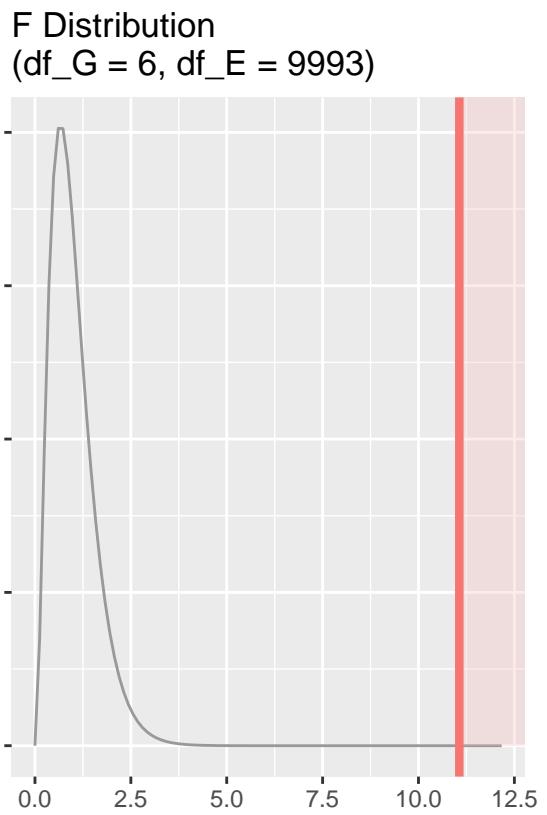
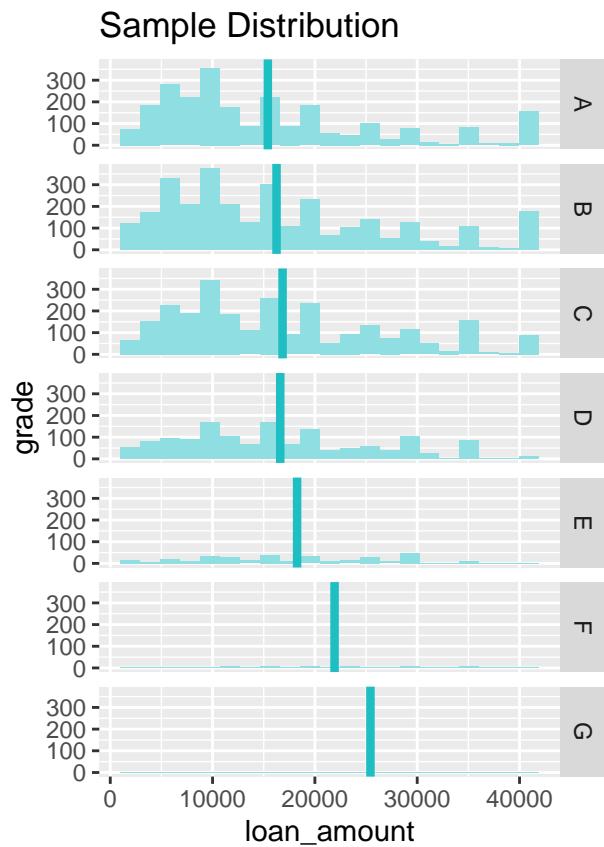
## Response variable: numerical

```

```

## Explanatory variable: categorical (7 levels)
## n_A = 2459, y_bar_A = 15399.5323, s_A = 10548.0711
## n_B = 3037, y_bar_B = 16251.3006, s_B = 10586.9141
## n_C = 2653, y_bar_C = 16840.6615, s_C = 10192.2345
## n_D = 1446, y_bar_D = 16614.2289, s_D = 9530.9932
## n_E = 335, y_bar_E = 18261.0448, s_E = 9344.7348
## n_F = 58, y_bar_F = 21922.8448, s_F = 8236.2302
## n_G = 12, y_bar_G = 25429.1667, s_G = 7619.0687
##
## ANOVA:
##           df      Sum_Sq   Mean_Sq      F p_value
## grade       6 7003167179.3584 1167194529.8931 11.0642 < 0.0001
## Residuals 9993 1054193833235.59 105493228.5836
## Total     9999 1061197000414.95
##
## Pairwise tests - t tests with pooled SD:
## # A tibble: 21 x 3
##   group1 group2     p.value
##   <chr>  <chr>     <dbl>
## 1 B      A      0.00224
## 2 C      A      0.000000547
## 3 C      B      0.0309
## 4 D      A      0.000360
## 5 D      B      0.269
## 6 D      C      0.500
## 7 E      A      0.00000174
## 8 E      B      0.000679
## 9 E      C      0.0171
## 10 E     D      0.00820
## # i 11 more rows

```



Linear Regression: A case study with R

STAT560 - Foundations of Data Science - Lecture 20

Linear Regression Analysis in R - A case study

In this lecture we use a case study to provide a general walk-through about setup, diagnostic test, evaluation of a linear regression model in R.

The data set that we will use for this study is the mtcars data, which is a built-in dataset in R that contains measurements for 32 different cars models. The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance.

The variables are:

mpg - Miles per Gallon cyl - # of cylinders disp - displacement, in cubic inches hp - horsepower drat - Rear axle ratio wt - weight qsec - 1/4 mile time; a measure of acceleration vs - 'V' or straight - engine shape (0 = V-shaped, 1 = straight) am - transmission; auto or manual (0 = automatic, 1 = manual) gear - # of forward gears carb - # of carburetors.

The goal of our case study is to fit a regression model to predict the variable mpg (= miles per gallon) using other variables. And we also want to answer questions such as "Is an automatic or manual transmission better for MPG?"

Load the mtcars Dataset

Since the mtcars dataset is a built-in dataset in R, we can load it by using the following data() command. We check the dimension of the data and take a look at the first several rows using head() function.

```
data(mtcars)
dim(mtcars)

## [1] 32 11

head(mtcars)

##          mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4   21.0   6 160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag 21.0   6 160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710  22.8   4 108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive 21.4   6 258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8 360 175 3.15 3.440 17.02  0  0    3    2
## Valiant     18.1   6 225 105 2.76 3.460 20.22  1  0    3    1
```

Converting variables with discrete values to factors (so that R will treat those as categorical).

```
mtcars$vs <- as.factor(mtcars$vs)
mtcars$am <- as.factor(mtcars$am)
mtcars$gear <- as.factor(mtcars$gear)
mtcars$cyl <- as.factor(mtcars$cyl)
```

Exploratory Data Analysis

We are not using a high-dimensional dataset, one of the common practice is to plot the summary statistics and either scatter plots or boxplots to help understand the possible relationship between each of the independent variables and the dependent variable mpg.

```
library(tidyverse)

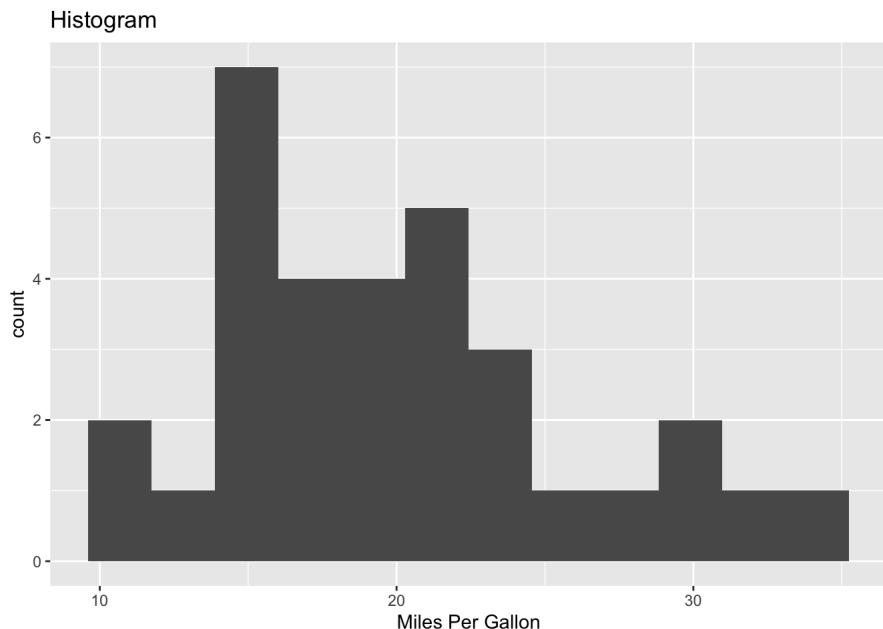
summary(mtcars)

##      mpg      cyl      disp       hp      drat
## Min.   :10.40   4:11   Min.   :71.1   Min.   :52.0   Min.   :2.760
## 1st Qu.:15.43   6: 7   1st Qu.:120.8  1st Qu.:96.5   1st Qu.:3.080
## Median :19.20   8:14   Median :196.3   Median :123.0   Median :3.695
## Mean   :20.09                    Mean   :230.7   Mean   :146.7   Mean   :3.597
## 3rd Qu.:22.80                    3rd Qu.:326.0   3rd Qu.:180.0   3rd Qu.:3.920
## Max.   :33.90                    Max.   :472.0   Max.   :335.0   Max.   :4.930
##      wt      qsec      vs      am      gear      carb
## Min.   :1.513   Min.   :14.50   0:18   0:19   3:15   Min.   :1.000
## 1st Qu.:2.581   1st Qu.:16.89   1:14   1:13   4:12   1st Qu.:2.000
## Median :3.325   Median :17.71                5: 5   Median :2.000
## Mean   :3.217   Mean   :17.85                Mean   :2.812
## 3rd Qu.:3.610   3rd Qu.:18.90                3rd Qu.:4.000
## Max.   :5.424   Max.   :22.90                Max.   :8.000
```

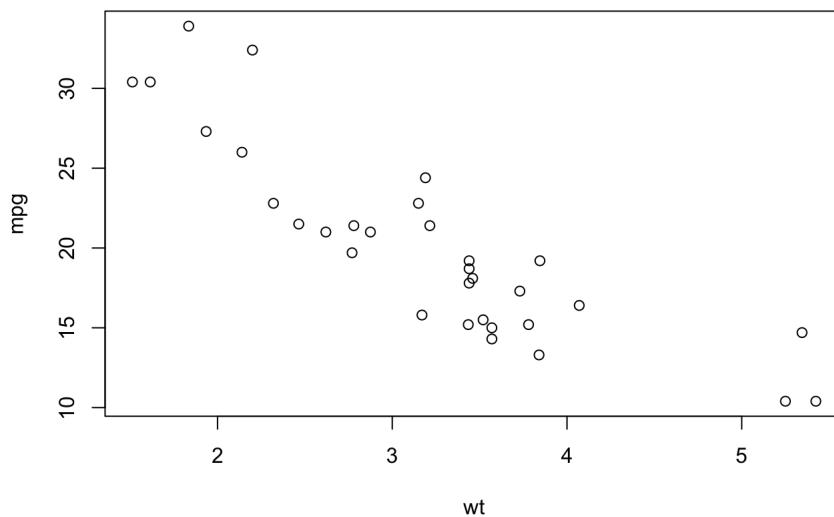
```
g=ggplot(data = mtcars, aes(x=mpg))
```



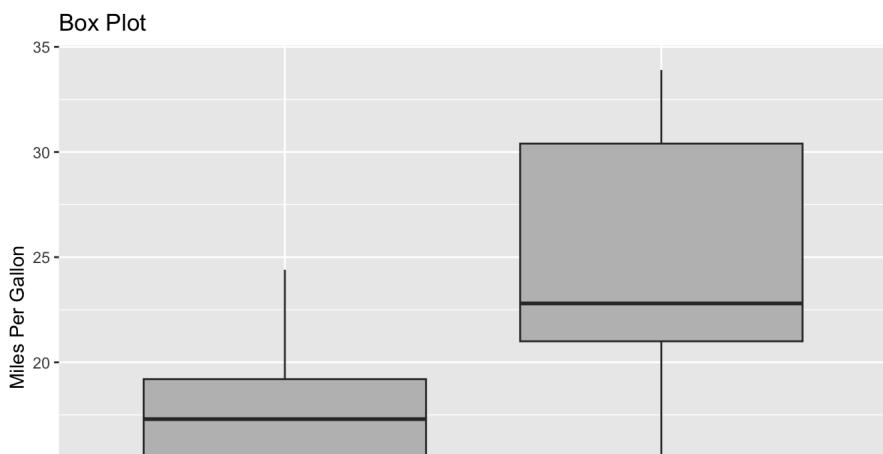
```
g+ geom_histogram(binwidth=1) + labs(title= "Histogram",x= "Miles Per Gallon")
```

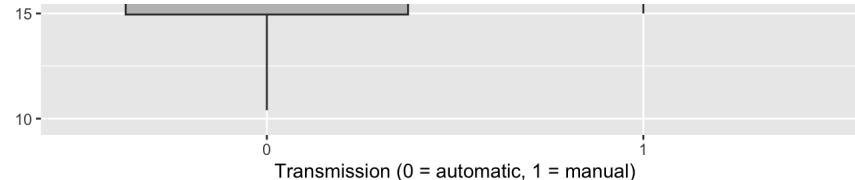


```
plot(mpg~wt,data=mtcars)
```



```
g=ggplot(data = mtcars, aes(x=am, y=mpg))  
g+ geom_boxplot(fill = "grey") + labs(title="Box Plot", x="Transmission (0 = automatic, 1 = manual)",y="Miles Per Gallon")
```

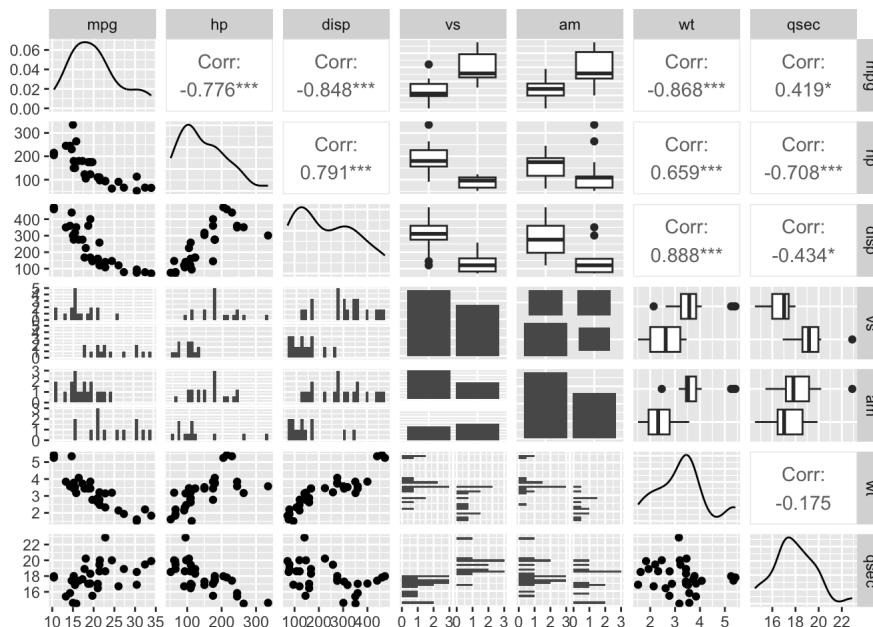




We can check the pairwise correlation and plots between desired variables using `ggpairs()` function in GGally library. We see hp, disp, wt are highly correlated with mpg. The values of mpg for different transmission/engine shape are also noticeably different. The relationship of mpg v.s. hp and disp seems to be non-linear.

We also see issue of multicollinearity (which is the correlation among predictors) among hp, disp, wt and qsec. Multicollinearity does not affect model prediction, but it will cause the inference on slopes to be inaccurate.

```
library(GGally)
ggpairs(data = mtcars %>% select(mpg, hp, disp, vs, am, wt, qsec))
```



Inference on mpg: Hypothesis Test and Confidence Interval

We can perform hypothesis test (2 sample t-test or ANOVA F-test) or confidence interval to check individual effects of categorical predictors. For the test results, there is a significant difference between the mpg for different transmission types, engine shape and number of cylinders.

```
t.test(mpg ~ am, data=mtcars)
```

```
## 
## Welch Two Sample t-test
##
## data: mpg by am
## t = -3.7671, df = 18.332, p-value = 0.001374
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
## -11.280194 -3.209684
## sample estimates:
## mean in group 0 mean in group 1
## 17.14737 24.39231
```

```
t.test(mpg ~ vs, data=mtcars)
```

```
## 
## Welch Two Sample t-test
##
## data: mpg by vs
## t = -4.6671, df = 22.716, p-value = 0.0001098
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
## -11.462508 -4.418445
## sample estimates:
## mean in group 0 mean in group 1
## 16.61667 21.55714
```

```
##          10.0100/        24.55/14

summary(aov(mpg ~ cyl, data=mtcars))

##             Df Sum Sq Mean Sq F value    Pr(>F)
## cyl          2  824.8   412.4   39.7 4.98e-09 ***
## Residuals   29  301.3    10.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Multiple Regression Model Estimation

We first fit a full model for mpg with all other 10 variables as predictors.

```
# "~." means model with all other variables
model=lm(mpg ~ ., data = mtcars)
summary(model)
```



```
## 
## Call:
## lm(formula = mpg ~ ., data = mtcars)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -3.2015 -1.2319  0.1033  1.1953  4.3085 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 15.09262   17.13627   0.881   0.3895    
## cyl6       -1.19940   2.38736  -0.502   0.6212    
## cyl8       3.05492   4.82987   0.633   0.5346    
## disp        0.01257   0.01774   0.708   0.4873    
## hp        -0.05712   0.03175  -1.799   0.0879 .  
## drat        0.73577   1.98461   0.371   0.7149    
## wt        -3.54512   1.90895  -1.857   0.0789 .  
## qsec        0.76801   0.75222   1.021   0.3201    
## vs1         2.48849   2.54015   0.980   0.3396    
## am1         3.34736   2.28948   1.462   0.1601    
## gear4      -0.99922   2.94658  -0.339   0.7382    
## gear5        1.06455   3.02730   0.352   0.7290    
## carb        0.78703   1.03599   0.760   0.4568    
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.616 on 19 degrees of freedom
## Multiple R-squared:  0.8845, Adjusted R-squared:  0.8116 
## F-statistic: 12.13 on 12 and 19 DF, p-value: 1.764e-06
```

Since none of the coefficients have a p-value less than 0.05, This is likely due to the multicollinearity issue among predictors. We'd like to get rid of some of the non-significant variables first. Backward Elimination method is used here. We can perform step-wise model selection using `step()` function in R base package.

```
#the metric used for variable selection at each step is AIC
re_model=step(model, direction = "backward")
```



```
## Start:  AIC=70.87
## mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + gear + carb
## 
##             Df Sum of Sq   RSS   AIC
## - gear     2   5.1061 135.16 68.103
## - drat     1   0.9408 130.99 69.101
## - disp     1   3.4354 133.49 69.705
## - carb     1   3.9503 134.00 69.828
## - vs       1   6.5693 136.62 70.447
## - qsec     1   7.1353 137.19 70.579
## - cyl      2  16.4500 146.50 70.682
## <none>           130.05 70.870
## - am       1  14.6316 144.68 72.282
## - hp       1  22.1573 152.21 73.905
## - wt       1  23.6065 153.66 74.208
## 
## Step:  AIC=68.1
## mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + carb
## 
##             Df Sum of Sq   RSS   AIC
## - drat     1   0.025125 135.66 68.103
```

```

## - drat 1 0.025 135.18 66.108
## - carb 1 3.866 139.02 67.005
## - vs 1 4.035 139.19 67.044
## - disp 1 4.732 139.89 67.204
## - qsec 1 4.941 140.10 67.251
## - cyl 2 14.238 149.40 67.308
## <none> 135.16 68.103
## - am 1 15.929 151.09 69.668
## - hp 1 18.284 153.44 70.163
## - wt 1 31.992 167.15 72.901
##
## Step: AIC=66.11
## mpg ~ cyl + disp + hp + wt + qsec + vs + am + carb
##
##          Df Sum of Sq   RSS   AIC
## - vs     1    4.250 139.43 65.099
## - carb  1    4.808 139.99 65.227
## - disp  1    4.895 140.08 65.247
## - qsec  1    4.918 140.10 65.252
## - cyl   2   17.095 152.28 65.919
## <none> 135.18 66.108
## - am     1   16.829 152.01 67.863
## - hp     1   19.891 155.07 68.501
## - wt     1   33.543 168.73 71.201
##
## Step: AIC=65.1
## mpg ~ cyl + disp + hp + wt + qsec + am + carb
##
##          Df Sum of Sq   RSS   AIC
## - carb  1    2.898 142.33 63.757
## - disp  1    4.214 143.65 64.052
## - cyl   2   13.993 153.43 64.160
## <none> 139.43 65.099
## - qsec  1   10.717 150.15 65.469
## - am    1   14.361 153.79 66.236
## - hp    1   15.649 155.08 66.503
## - wt    1   36.334 175.77 70.510
##
## Step: AIC=63.76
## mpg ~ cyl + disp + hp + wt + qsec + am
##
##          Df Sum of Sq   RSS   AIC
## - disp  1    1.651 143.98 62.126
## - cyl   2   11.107 153.44 62.162
## - qsec  1    8.078 150.41 63.524
## <none> 142.33 63.757
## - hp    1   15.403 157.73 65.046
## - am    1   17.424 159.75 65.453
## - wt    1   40.707 183.04 69.807
##
## Step: AIC=62.13
## mpg ~ cyl + hp + wt + qsec + am
##
##          Df Sum of Sq   RSS   AIC
## - cyl   2   16.085 160.07 61.515
## - qsec  1    7.044 151.03 61.655
## <none> 143.98 62.126
## - hp    1   15.443 159.42 63.387
## - am    1   16.566 160.55 63.611
## - wt    1   52.932 196.91 70.145
##
## Step: AIC=61.52
## mpg ~ hp + wt + qsec + am
##
##          Df Sum of Sq   RSS   AIC
## - hp    1    9.219 169.29 61.307
## <none> 160.07 61.515
## - qsec  1   20.225 180.29 63.323
## - am    1   25.993 186.06 64.331
## - wt    1   78.494 238.56 72.284
##
## Step: AIC=61.31
## mpg ~ wt + qsec + am
##
##          Df Sum of Sq   RSS   AIC
## <none> 169.29 61.307
## - am    1   26.178 195.46 63.908
## - qsec  1   109.034 278.32 75.217
## - wt    1   183.347 352.63 82.790

```

```
summary(re_model)

##
## Call:
## lm(formula = mpg ~ wt + qsec + am, data = mtcars)
##
## Residuals:
##   Min     1Q Median     3Q    Max 
## -3.4811 -1.5555 -0.7257  1.4110  4.6610 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 9.6178    6.9596   1.382  0.177915  
## wt         -3.9165    0.7112  -5.507 6.95e-06 *** 
## qsec        1.2259    0.2887   4.247 0.000216 *** 
## am1         2.9358    1.4109   2.081 0.046716 *  
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.459 on 28 degrees of freedom
## Multiple R-squared:  0.8497, Adjusted R-squared:  0.8336 
## F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

Previously we noticed curved relationship between mpg and weight. We may want to add a higher order term of wt variables to the model, i.e. add the squared weight.

```
# poly() function is used for the higher order polynomial terms
re_model2=lm(mpg ~ poly(wt, 2)+qsec+am, data = mtcars)
# This re_model2 is preferred since it has a Lower AIC and higher Adjusted R-squared compared with re_model
summary(re_model2)
```

```
##
## Call:
## lm(formula = mpg ~ poly(wt, 2) + qsec + am, data = mtcars)
##
## Residuals:
##   Min     1Q Median     3Q    Max 
## -3.7508 -1.3125 -0.4954  1.0722  4.8553 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.3532    5.2560   0.448  0.65792  
## poly(wt, 2)1 -25.4634   3.7659  -6.762 2.92e-07 *** 
## poly(wt, 2)2  7.0659    2.5093   2.816  0.00898 ** 
## qsec         0.9705    0.2739   3.543  0.00146 ** 
## am1          1.0215    1.4346   0.712  0.48252  
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.201 on 27 degrees of freedom
## Multiple R-squared:  0.8838, Adjusted R-squared:  0.8666 
## F-statistic: 51.33 on 4 and 27 DF,  p-value: 3.107e-12
```

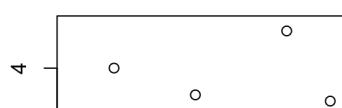
The final model here is $\text{mpg} = 2.35 - 25.46\text{wt} + 7.07\text{squared_wt} + 0.97\text{qsec} + 1.02\text{am}$. The R-squared value of 0.8838 confirms that this model explains about 88% of the variance in MPG. The coefficients conclude that 1 unit improvement in acceleration (qsec) increase the MPG by 0.97. Increasing the weight decreases MPG in general, but the relationship is non-linear. A Manual transmission improves the MPG by 1.02.

Residuals & Diagnostics

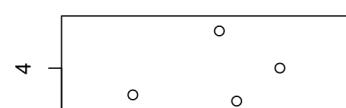
Residual plots are used for model diagnostics. The following plots shows: The randomness of the Residuals vs. index supports the assumption of independence. The Residuals vs.fitted values plot did not present any pattern, which confirms the constant variance assumption.

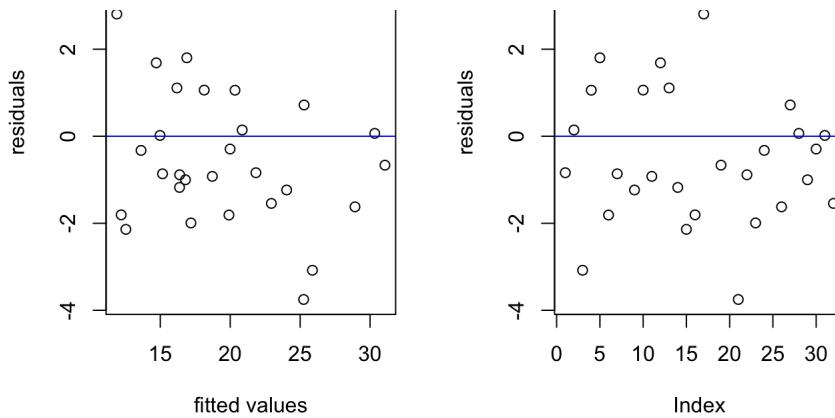
```
fit = lm(mpg ~ poly(wt, 2)+qsec+am, data = mtcars) #final model
par(mfrow=c(1,2))
plot(fit$res~fit$fitted,main="Residual v.s. fitted values",xlab="fitted values",ylab="residuals");abline(h=0,col="blue")
plot(fit$res, main="Residual v.s. index",ylab="residuals");abline(h=0,col="blue")
```

Residual v.s. fitted values



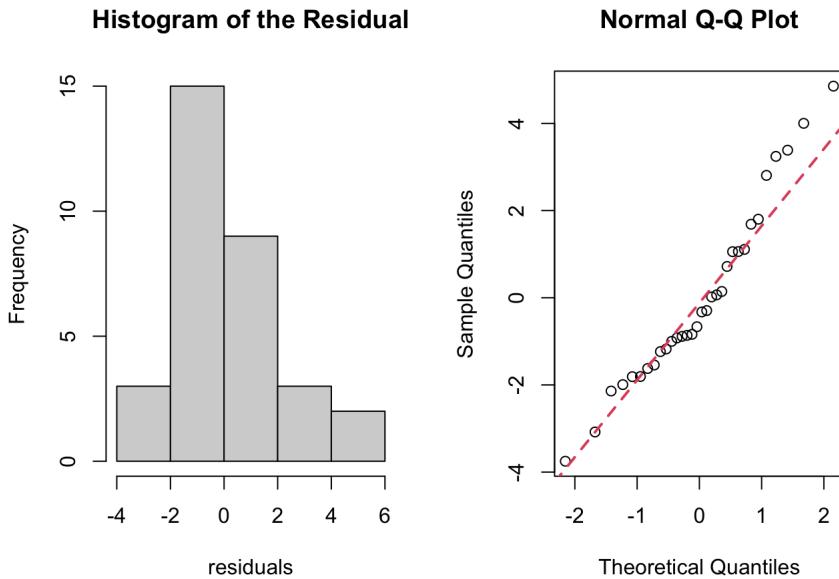
Residual v.s. index





The histogram and the Normal Q-Q plot following closely to the line conclude that the distribution of residuals is approximately normal.

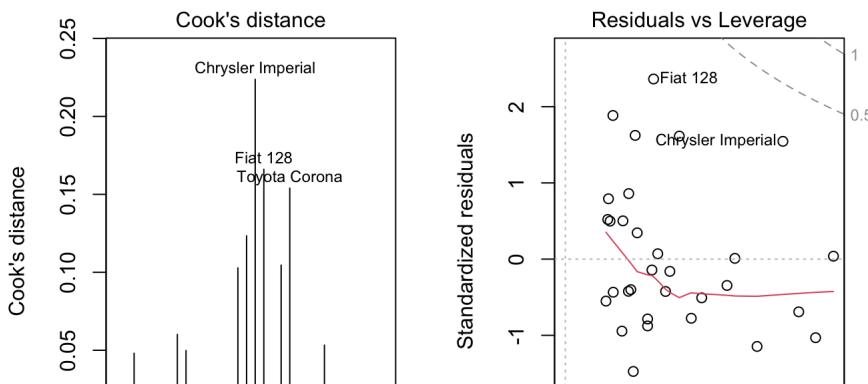
```
par(mfrow=c(1,2))
hist(fit$res,main="Histogram of the Residual",xlab="residuals")
qqnorm(fit$res); qqline(fit$res, col = 2,lwd=2, lty=2)
```

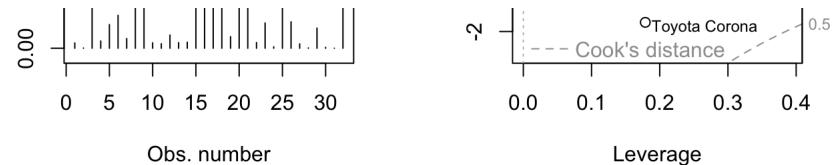


```
par(mfrow=c(1,2))
```

Cook's distance and residual v.s. leverage plots are used to identify influential points or points with high leverage. Since all points are within the 0.05 lines, the Residuals vs. Leverage concludes that there are no outliers.

```
par(mfrow=c(1,2))
plot(fit,4)
plot(fit,5)
```





STAT560 Lecture 18: Intro to Linear Regression

Beidi Qiang

SIUE

- ▶ Many problems involve a study or analysis of the relationship between two or more variables.
- ▶ For example, we want to study the displacement of an object y (in meter) and time x (in second). We have a **deterministic (perfect) linear relationship**, such as $y = 5 + 10x$, when the velocity is 10 m/s. We say it is **deterministic** since we know the exact value of y just by knowing the value of x .
- ▶ However, deterministic relationship is unrealistic in almost any natural process.
- ▶ For example, the electrical energy consumption of a house (y) is related to the size of the house (x , in square feet), but it is unlikely to be a deterministic relationship.

Linear Regression

Linear regression is the statistical method for fitting a line to data where the relationship between two variables, x and y , can be modeled by a straight line with some error, e.g.

$$Y = \beta_0 + \beta_1 x + \epsilon$$

- ▶ **Regression coefficients:** β_0 (intercept), β_1 (slope)
- ▶ ϵ is the **random error term**.
- ▶ We will call this model the **simple linear regression** model, because it has only one independent variable 
- ▶ No simple curve passed exactly through all the data. Commonly data appear as a cloud of points.

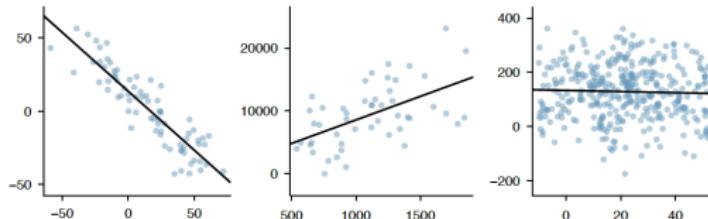


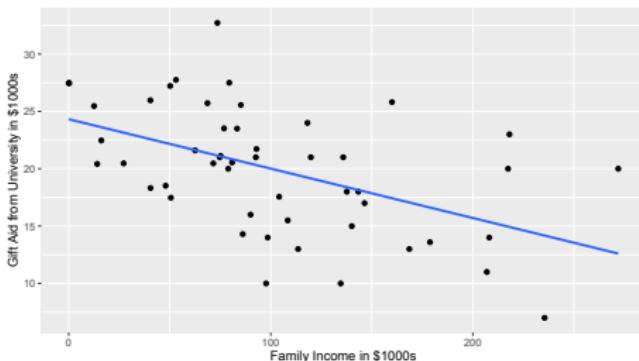
Figure 8.2: Three data sets where a linear model may be useful even though the data do not all fall exactly on the line.

- ▶ The slope, β_1 , quantifies the average change in y brought about by a one-unit change in x
- ▶ We can calculate an estimate (**predicted value**) for y , denoted by \hat{y} , by plugging x into the model.
- ▶ **Residuals** are the leftover variation in the data after accounting for the model fit. The residual of the i th observation (x_i, y_i) is the difference of the observed response (y_i) and the response we would predict based on the model (\hat{y}_i) :

$$e_i = y_i - \hat{y}_i$$

Example:

Consider family income (in \$1000s) and financial gift aid (in \$1000s) data from a random sample of 50 students in the freshman class of Elmhurst College in Illinois.



```
> aid=read.csv(file="https://www.openintro.org/data/csv/elmhurst.csv", header=T)
> lm(gift_aid ~ family_income, data = aid)
```



Coefficients:

(Intercept)	family_income
24.31933	-0.04307

The equation for the regression line: $\hat{y} = 24.319 - 0.043x$

For a family with income \$160k predicted gift aid:

$$\hat{y} = 24.319 - 0.043 \times 160 = 17.428$$

If the family actually receives a gift aid of \$25.8k, the residual is

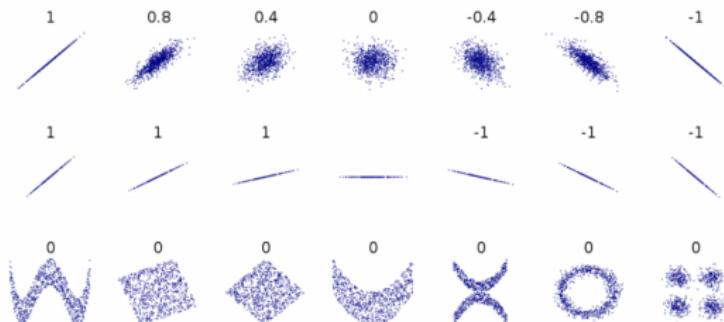
$$e = 25.8 - 17.428 = 8.372$$

Coefficient of Correlation

- ▶ **Correlation** measures the linear relationship between two quantitative variables.
- ▶ To calculate sample correlation:

$$R = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{SS_{xy}}{\sqrt{SS_{xx} SS_{yy}}}.$$

- ▶ Correlation always takes values between -1 and 1. it describes the direction and strength of the linear relationship between two variables.



Least squares estimation

We want to fit a regression model, i.e., we would like to estimate the regression coefficients β_0 and β_1 using **least squares estimation**.

- ▶ Least squares says to choose the values β_0 and β_1 that minimize

$$Q = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n [y_i - (\beta_0 + \beta_1 x_i)]^2.$$



- ▶ In calculus, we minimize or maximize a multivariable function by taking partial derivatives with respect to each arguments and set them to 0. So, taking partial derivative of Q , we obtain

$$\frac{\partial Q(\beta_0, \beta_1)}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) \stackrel{\text{set}}{=} 0$$

$$\frac{\partial Q(\beta_0, \beta_1)}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) x_i \stackrel{\text{set}}{=} 0$$

- ▶ Solve above system of equations yields the **least squares estimators**.

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{s_y}{s_x} R.$$

Conditions for the least squares line

- ▶ When fitting a least squares line, $Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$, We generally assume the following
 - ▶ Linearity. The data should show a linear trend.
 - ▶ Nearly normal residuals, i.e. ϵ_i are approximately normally distributed.
 - ▶ No outliers or concerns about influential points.
 - ▶ Constant variability. The variability of points around the least squares line remains roughly constant. That is, $\text{Var}(\epsilon_i) = \sigma^2$, for all $i = 1, 2, \dots, n$,
 - ▶ Independent observations.
- ▶ Those assumptions of the error terms can be summarized as

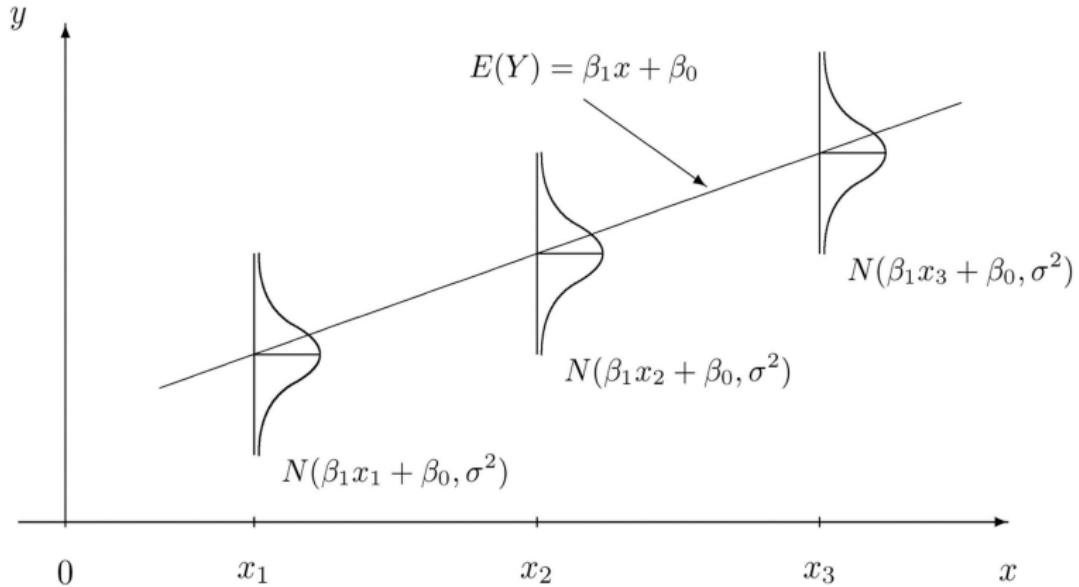
$$\epsilon_1, \epsilon_2, \dots, \epsilon_n \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2),$$

where *i.i.d.* stands for independent and identically distributed.

- ▶ Under the assumptions, it follows that

$$Y_i \sim \mathcal{N}(\beta_0 + \beta_1 x_i, \sigma^2)$$

Pictorial Illustration of Model Assumptions



Sampling Distribution of Least Squares Estimators

- ▶ Recall that

$$\begin{aligned}\hat{\beta}_0 &= \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 &= \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}.\end{aligned}$$

- ▶ $\hat{\beta}_0$ and $\hat{\beta}_1$ are functions of y_i , so they are random variables and have their **sampling distributions**.

- ▶ It can be shown that

$$\hat{\beta}_0 \sim \mathcal{N} \left(\beta_0, \left(\frac{1}{n} + \frac{\bar{x}^2}{SS_{xx}} \right) \sigma^2 \right) \text{ and } \hat{\beta}_1 \sim \mathcal{N} \left(\beta_1, \frac{\sigma^2}{SS_{xx}} \right)$$

- ▶ Note that both $\hat{\beta}_0$ and $\hat{\beta}_1$ are unbiased.

Sampling Distribution of Least Squares Estimators, cont.

- ▶ Since σ^2 is unknown, we need to estimate it.

The residuals $e_i = y_i - \hat{y}_i$ are used to obtain an estimator of σ^2 . The sum of squares of the residuals, often called the **error sum of squares or residual sum of squares**, is

$$SSE = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2.$$

It can be shown that the expected value of the error sum of squares is $E(SSE) = (n - 2)\sigma^2$. Therefore an unbiased estimator of σ^2 is

$$MSE = \hat{\sigma}^2 = \frac{SSE}{n - 2}$$

- ▶ The **estimated standard error** of $\hat{\beta}_0$ and $\hat{\beta}_1$ are then

$$se(\hat{\beta}_0) = \sqrt{\left(\frac{1}{n} + \frac{\bar{x}^2}{SS_{xx}}\right) \hat{\sigma}^2} \text{ and } se(\hat{\beta}_1) = \sqrt{\frac{\hat{\sigma}^2}{SS_{xx}}}$$

where

$$\hat{\sigma}^2 = \frac{SSE}{n - 2}$$

- ▶ An important part of assessing the adequacy of a linear regression model is testing statistical hypotheses about the model parameters and constructing certain confidence intervals.
- ▶ In practice, inference for the slope parameter β_1 is of primary interest because of its connection to the independent variable x in the model.
- ▶ Inference for β_0 is less meaningful, unless one is explicitly interested in the mean of y when $x = 0$. We will focus on inference on β_1 .
- ▶ Under our model assumptions, the following sampling distribution arises:

$$t = \frac{\hat{\beta}_1 - \beta_1}{\text{se}(\hat{\beta}_1)} = \frac{\hat{\beta}_1 - \beta_1}{\sqrt{\hat{\sigma}^2 / SS_{xx}}} \sim t(n - 2)$$

Confidence Interval of β_1

- The sampling distribution of $\hat{\beta}_1$ leads to the following $(1 - \alpha)100\%$ confidence interval of β_1 :

$$\hat{\beta}_1 \pm t_{\alpha/2} \sqrt{\hat{\sigma}^2 / SS_{xx}}$$

Point Estimate Quantile standard error

- Note that this is two-sided confidence interval, which corresponds to the test $H_0 : \beta_1 = 0$ against $H_a : \beta_1 \neq 0$.
 - If '0' is covered by this interval, we fail to reject H_0 at significance level of α . This suggests that y and x are not linearly related.
 - If '0' is not covered by this interval, we reject H_0 at significance level of α . This suggests that y and x are linearly related.

Hypothesis Test for β_1

- ▶ Suppose we want to test β_1 equals to a certain value, say $\beta_{1,0}$, that is our interest is to test

$$H_0 : \beta_1 = \beta_{1,0} \text{ versus } H_a : \beta_1 \neq \beta_{1,0}$$

where $\beta_{1,0}$ is often set to 0.

- ▶ The test statistic under the null is

$$t_0 = \frac{\hat{\beta}_1 - \beta_{1,0}}{\sqrt{\hat{\sigma}^2 / SS_{xx}}} \sim t(n-2).$$

- ▶ The p -value of the test is $2P(T_{n-2} < -|t_0|)$, you can use R to find this probability. Remember that smaller p -value provide stronger evidence against H_0
- ▶ Let us look at an example...

Gift Aid Example

We usually rely on statistical software to identify point estimates, standard errors, test statistics, and p-values in practice.

```
> fit = lm(gift_aid~family_income, data=aid)
> summary(fit)

Residuals:
    Min      1Q  Median      3Q     Max 
-10.1128 -3.6234 -0.2161  3.1587 11.5707 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 24.31933   1.29145 18.831 < 2e-16 ***
family_income -0.04307   0.01081 -3.985 0.000229 ***  
---
Residual standard error: 4.783 on 48 degrees of freedom
```

What is your conclusion based the R output? Note that the residual standard error is $\sqrt{\hat{\sigma}^2} = \sqrt{MSE} = 4.783$.

Analysis of Variance Approach to Test Significance of Regression

- (Analysis of Variance Identity) We decompose the total variability into

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n (y_i - \hat{y}_i)^2.$$

- We usually call $SSE = \sum_{i=1}^n (y_i - \hat{y}_i)^2$ the error sum of squares and $SSR = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$ the regression sum of squares.
- Symbolically, we have

$$SSTO = SSR + SSE.$$

- We can use an F-test to test overall model significance. Under the null (all slopes are 0),

$$F_0 = \frac{SSR/1}{SSE/(n-2)} = \frac{MSR}{MSE} \sim F(1, n-2).$$

- If p -value is small, we will reject null and conclude the slopes are not all 0.

ANOVA Table for Simple Linear Regression



We can summarize these results in the ANOVA table.

Source of Variation	SS	df	MS	F
Regression	$\sum_{i=1}^n (\hat{y}_i - \bar{y})^2$	1	$\frac{SSR}{1}$	$F = \frac{MSR}{MSE}$
Error	$\sum_{i=1}^n (y_i - \hat{y}_i)^2$	$n - 2$	$\frac{SSE}{n-2}$	
Total	$\sum_{i=1}^n (y_i - \bar{y})^2$	$n - 1$		

Gift Aid Example: F Test

We wish to test $H_0 : \beta_1 = 0$ against $H_a : \beta_1 \neq 0$ using the ANOVA approach. You can use `anova` command in R.

```
> fit = lm(gift_aid ~ family_income, data = aid)
> anova(fit)
Analysis of Variance Table

Response: gift_aid
            Df  Sum Sq Mean Sq F value    Pr(>F)
family_income  1  363.16  363.16  15.877 0.0002289 ***
Residuals     48 1097.92   22.87
```

Again, we reject $H_0 : \beta_1 = 0$ at any reasonable α level and conclude that there is a strong evidence support $\beta_1 \neq 0$.

Coefficient of Determination

- ▶ **Coefficient of Determination**, denoted by R^2 , measures the contribution of x in the predicting of y .
- ▶ Recall that

$$SSTO = \sum_{i=1}^n (y_i - \bar{y})^2, SSE = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

- ▶ If the model contribute to prediction of y_i , then we expect $SSE \ll SSTO$. In other words, the independent variable x “explain” significant amount of variability among data.
- ▶ Coefficient of determination is the proportion of total sample variation explained by linear relationship:

$$R^2 = \frac{\text{Explained Variability}}{\text{Total Variability}} = \frac{SSR}{SSTO} = \frac{SSTO - SSE}{SSTO}.$$

- ▶ It can be shown that the coefficient of determination of a simple linear regression equals to the squared sample coefficient of correlation between x and y .

Example: Removal Project Example

We can use command `cor` to calculate sample coefficient of correlation. The coefficient of determination r^2 is called Multiple R-squared in the summary of simple linear regression.

```
> cor(aid$gift_aid,aid$family_income)
[1] -0.4985561
> fit<-lm(gift_aid~family_income, data=aid)
> summary(fit)

Residuals:
    Min      1Q  Median      3Q     Max 
-10.1128 -3.6234 -0.2161  3.1587 11.5707 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 24.31933   1.29145 18.831 < 2e-16 ***
family_income -0.04307   0.01081 -3.985 0.000229 ***  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.783 on 48 degrees of freedom
Multiple R-squared:  0.2486, Adjusted R-squared:  0.2329 
F-statistic: 15.88 on 1 and 48 DF,  p-value: 0.0002289
```

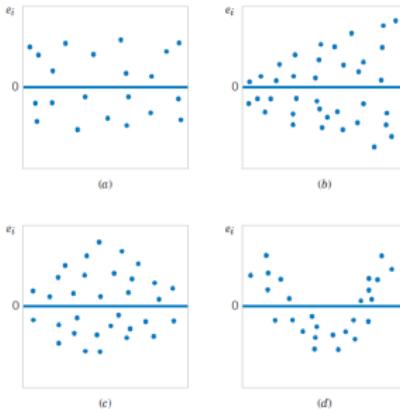
Model Adequacy Checking: Residual Analysis

Analysis of the residuals is frequently helpful in checking the assumption that the errors are approximately normally distributed with constant variance, and in determining whether additional terms in the model would be useful.

- ▶ We plot the residuals (1) against predicted values \hat{y} , and (2) against the independent variable x to check for linearity, constant variance and outliers.
- ▶ As an approximate check of normality, we can use the histogram or normal probability plot of residuals.
- ▶ We use ACF plot to check independence.
- ▶ Model checking is an important exercise because if the model assumptions are violated, then our analysis (and all subsequent interpretations) could be compromised.

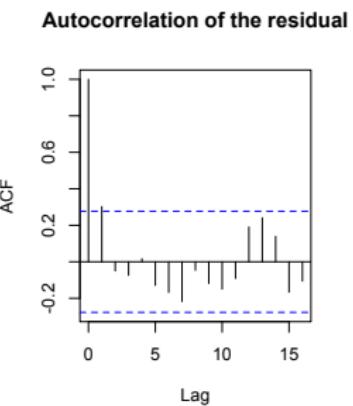
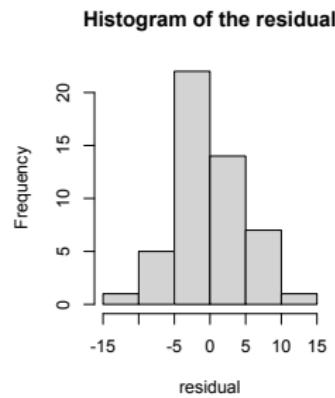
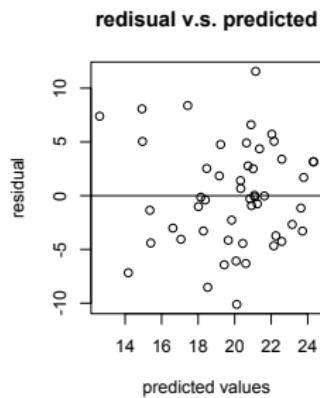
Residual Plots

Following are some scatterplot of residuals e_i 's v.s. predicted values.



- ▶ Pattern (a) represents the ideal situation.
- ▶ Pattern (b) represents the cases where the variance of the observations may be increasing with the magnitude of y_i or x_i . Pattern (b) and (c) represents the unequal variance cases.
- ▶ Pattern (d) indicates the linear relationship is not proper. We need to add higher order term, which requires multiple linear regression.

Residual Plots for Gift Aid Regression

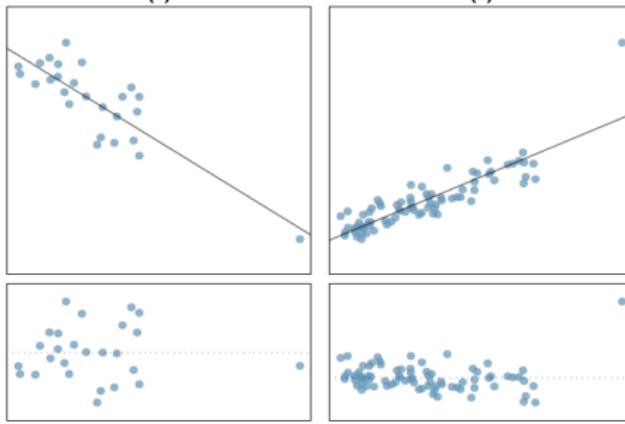


Types of outliers in linear regression

Recall that an outlier is any point that doesn't appear to belong with the vast majority of the other points.

- ▶ In regression, some outlier are far from the other points, but it only slightly influence the line.
- ▶ While some points that fall horizontally away from the center of the cloud tend to pull harder on the line. We call them points with high leverage, or influential points.

Figure: Least squares line and residual plot for data with outliers



Categorical predictors with two levels

Categorical variables can also be used in predicting outcomes. Here we consider a categorical predictor with two levels (recall that a level is the same as a category).

- ▶ To incorporate a categorical predictors into a regression equation, we must convert the categories into a numerical form. We will do so using an indicator variable x , which takes value 1 and 0.
- ▶ The fitted model is still $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$, for x being either 0 or 1.
- ▶ The estimated intercept, $\hat{\beta}_0$ is the average value of the response variable for the first category (i.e. the category corresponding to an indicator value of 0). The value of $\hat{\beta}_0 + \hat{\beta}_1$ is the average value of response variable in the 2nd category (corresponding to an indicator value of 1).

STAT560 Lecture 19: Multiple Regression

Beidi Qiang

SIUE

Multiple Regression

Multiple regression extends simple two-variable regression to the case that still has one response but many predictors (denoted $x_1, x_2, x_3, \dots, x_k$).

- ▶ Multiple regression model (when there are k predictors):

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \epsilon$$

- ▶ Model parameters: β_0 (intercept), $\beta_1, \beta_2, \dots, \beta_k$ (slopes).
- ▶ We estimate the parameters $\beta_0, \beta_1, \beta_2, \dots, \beta_k$ in the same way as we did in the simple linear regression. We select $\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_k$ that minimize the sum of the squared residuals:

$$SSE = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$



- ▶ We typically use a computer software (e.g. R) to minimize the sum of squares and compute point estimates.

Example: Loan Data

Consider the Loan data from OS textbook. The data is the complete version of the loan50 data we introduced before. The dataset includes results from 10,000 loans with information on terms of the loan as well as information about the borrower. We'd like to fit a multiple regression model for predicting interest rate with some variables from the data.

```
> loan=read.csv(file="https://www.openintro.org/data/csv/loans_full_schema.csv",header=T)
> lm(interest_rate~homeownership+debt_to_income, data=loan)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	11.046539	0.098077	112.631	<2e-16 ***
homeownershipOWN	0.320679	0.151948	2.110	0.0348 *
homeownershipRENT	0.981873	0.107021	9.175	<2e-16 ***
debt_to_income	0.049398	0.003297	14.982	<2e-16 ***

Residual standard error: 4.927 on 9972 degrees of freedom

(24 observations deleted due to missingness)

Multiple R-squared: 0.02833, Adjusted R-squared: 0.02804

F-statistic: 96.92 on 3 and 9972 DF, p-value: < 2.2e-16

The equation for the regression line:

$$\widehat{\text{interest_rate}} = 11.047 + 0.321 \times \text{own} + 0.982 \times \text{rent} + 0.049 \times \text{debt_to_income}$$

Example: Loan Data, cont.

- ▶ Independent variables used: homeownership, debt_to_income.
- ▶ Effective number of predictors (number of slopes): $k = 3$. 
- ▶ Note categorical predictor with p levels will be represented by $p - 1$ terms in a multiple regression model.
- ▶ For an observation with
homeownership = own, debt_to_income = 15,

$$\widehat{\text{interest_rate}} = 11.047 + 0.321 \times 1 + 0.982 \times 0 + 0.049 \times 15 = 12.102$$

Categorical variables as predictors: Dummy Encoding

Most machine learning models accept only numerical variables. So categorical variables are converted to number so the model can understand better. This is done using **Dummy Encoding**.

Figure: Dummy Encoding for a categorical predictor with 3 levels. Level "Green" is skipped from the encoding, since a (0,0) signifies green

The diagram illustrates the process of dummy encoding. On the left, a vertical stack of three boxes represents the categorical levels: Red, Blue, and Green. An arrow points from this stack to a 4x3 matrix on the right. The matrix has columns labeled Red, Blue, and Green. The rows represent individual samples. The values in the matrix are as follows:

	Red	Blue	Green
1	1	0	0
0	0	1	0
0	0	0	1

A diagonal line through the matrix indicates that the first two rows (Red and Blue) correspond to the first two columns (1 and 0), while the third row (Green) corresponds to the third column (0). A note states that (0,0) signifies green.

- ▶ Dummy Encoding creates a new binary predictor for each of the $p - 1$ levels and assigns a value of 1 to the new predictor of each sample that corresponds to its original category.
- ▶ The level skipped is often called the reference level, or the baseline.
- ▶ We then use the created dummy variables same way as numerical predictors in fitting a regression model.



Interpreting categorical variables in regression

In loan example, the variable homeownership has 3 levels (OWN, RENT, MORTGAGE). Regression output provides 2 rows, one for OWN and one for RENT. The corresponding coefficients measures relative difference for each level against the reference level, which is MORTGAGE.

$$\widehat{\text{interest_rate}} = 11.047 + 0.321 \times \text{own} + 0.982 \times \text{rent} + 0.049 \times \text{debt_to_income}$$

- When homeownership is *MORTGAGE*, the model of is

$$\widehat{\text{interest_rate}} = 11.047 + 0.049 \times \text{debt_to_income}$$

- When homeownership is *OWN*, the model of is

$$\widehat{\text{interest_rate}} = (11.047 + 0.321) + 0.049 \times \text{debt_to_income}$$

Interpretation: the average interest rate for borrowers with owned homes is 0.321 points higher than borrowers with mortgage.

- When homeownership is *RENT*, the model of is

$$\widehat{\text{interest_rate}} = (11.047 + 0.982) + 0.049 \times \text{debt_to_income}$$

Interpretation: the average interest rate for borrowers with rent homes is 0.982 points higher than borrowers with mortgage.

You may include many predictors in a multiple regression model. However, the best model is not always the most complicated. Sometimes including variables that are not evidently important can actually reduce the accuracy of predictions. We can use several model selection strategies to find models that are preferable.

- ▶ Step-wise model selection: backward elimination and forward selection. These common strategies for adding or removing variables one at a time.
- ▶ LASSO regression: a regularization technique that shrinks the coefficients toward 0 and thus select predictors (We'll cover this topic in a later course).

Backward elimination

Backward elimination starts with the model that includes all potential predictor variables.

- ▶ Variables are eliminated one-at-a-time 
- ▶ At each step, we select the predictor to be eliminated according to criteria such as p-value, adjusted R^2 or AIC. Then refit the model.
- ▶ Stop when we are satisfied that all remaining variables.

Example: (Predicting birth weight of babies) Output for the full model:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-80.41	14.35	-5.60	0.0000
gestation	0.44	0.03	15.26	0.0000
parity	-3.33	1.13	-2.95	0.0033
age	-0.01	0.09	-0.10	0.9170
height	1.15	0.21	5.63	0.0000
weight	0.05	0.03	1.99	0.0471
smoke	-8.40	0.95	-8.81	0.0000

In Backward elimination with p-value , we will first eliminate age since it has the largest p-value that is above the significance level (0.05).

Forward selection

The forward selection strategy is the reverse of the backward elimination technique. We begin with a model that has no predictors (with only intercept).



- ▶ We add variables one-at-a-time
- ▶ At each step, we fit models by adding each possible predictor to the model from the previous step, and select the predictor according their importance using p-value, adjusted R^2 or AIC.
- ▶ Stop when no other important variables are found.

Example: In the 1st step of forward selection with p-value , we will first add gestation since it has the small p-value that is below the significance level (0.05).

```
lm(formula = bwt ~ gestation, data = baby)
  Estimate Std. Error t value Pr(>|t|).
(Intercept) -10.06418   8.32220 -1.209   0.227
gestation    0.46426   0.02974 15.609  <2e-16 ***
---
lm(formula = bwt ~ age, data = baby)
  Estimate Std. Error t value Pr(>|t|)
(Intercept) 116.68346   2.50438 46.592  <2e-16 ***
age         0.10622   0.08989  1.182   0.238.
---
lm(formula = bwt ~ weight, data = baby)
  Estimate Std. Error t value Pr(>|t|)
(Intercept) 102.15029   3.25693 31.364  < 2e-16 ***
weight      0.13485   0.02499  5.396 8.21e-08 ***
lm(formula = bwt ~ parity, data = baby)
  Estimate Std. Error t value Pr(>|t|)
(Intercept) 120.0684   0.6005 199.942  <2e-16 ***
parity     -1.9287   1.1895 -1.621   0.105
---
lm(formula = bwt ~ height, data = baby)
  Estimate Std. Error t value Pr(>|t|)
(Intercept) 27.6810   13.0298  2.124   0.0338 *
height      1.4334   0.2033  7.052 2.97e-12 ***
---
lm(formula = bwt ~ smoke, data = baby)
  Estimate Std. Error t value Pr(>|t|)
(Intercept) 123.047   0.649 189.597  <2e-16 ***
smoke       -8.938    1.033  -8.653  <2e-16 ***
```

Recall:

$$R^2 = 1 - \frac{SSE}{SSTO} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

The adjusted R² is computed as

$$R_{adj}^2 = 1 - \frac{\frac{SSE/(n-k-1)}{SSTO/(n-1)}}{1 - \frac{SSE}{SSTO}} = 1 - \frac{\frac{n-1}{SSTO}}{\frac{n-k-1}{SSTO}}$$

- ▶ n is the number of observations and k is the number of effective predictors in the model.
- ▶ Adjusted R^2 will be smaller than the unadjusted R^2 .
- ▶ R^2 always increasing when extra predictors are added to the model.
Adjusted R^2 attempts to correct for this overestimation by penalize on the number of predictors. Adjusted R^2 might decrease if a specific effect does not improve the model.
- ▶ One may use Adjusted R^2 as criteria in the step-wise selections, where you eliminate (backward) or add (forward) the variable that leads to the largest improvement in adjusted R^2 .

Other model accuracy metrics

There are other model selection metrics similar to R^2_{adj} that can be used for model selection purpose.

- ▶ **Akaike information criterion (AIC)** is a method for evaluating how well a model fits the data and the complexity of the model. AIC is calculated from the number of predictors used to build the model (k) and the maximum likelihood estimate of the model (\hat{L}). Models with smaller AIC are preferred.

$$AIC = 2k - 2\ln(\hat{L})$$

- ▶ **Bayesian information criterion (BIC)** is another criteria for model selection that measures the trade-off between model fit and complexity of the model. Compared to AIC, BIC penalizes the model more for its complexity. Models with smaller BIC are preferred.

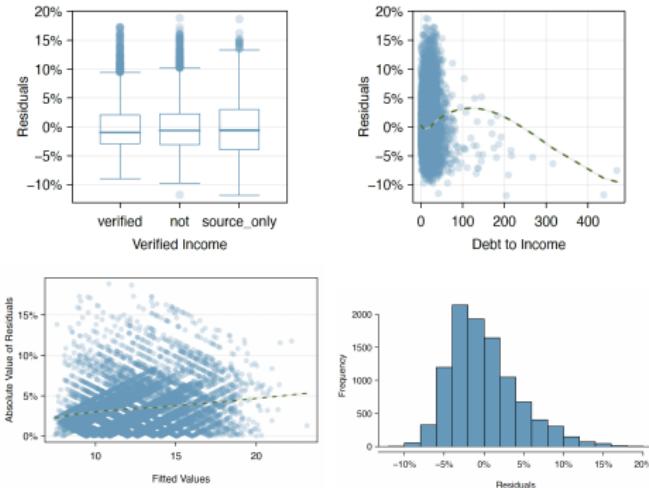
$$BIC = k\ln(n) - 2\ln(\hat{L})$$

- ▶ There are other metrics such as: MAE (Mean Absolute Error), AICc (AIC corrected for small sample), Mallows Cp (a variation of AIC), etc.

Similar to simple linear regression, we can perform residual analysis to check model conditions: errors are approximately normally distributed with constant variance.

- ▶ We plot the residuals (1) against predicted values \hat{y} , and (2) against the independent variable x to check for linearity, constant variance and outliers.
- ▶ As an approximate check of normality, we can use the histogram or normal probability plot of residuals.

Example: Some Residual Plots for Loan Data



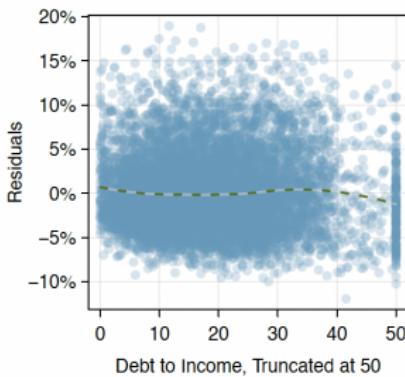
- ▶ no significant outliers observed given the size of the data.
- ▶ There is some minor differences in variability between the verified income groups.
- ▶ There is a very clear pattern for the debt-to-income variable. This variable is very strongly right skewed.

Options for improving the model fit

There are several options for improvement of a model, including transforming variables, seeking out additional variables, adding higher order or interaction terms. For example, you may

- ▶ Transform predictors: $\log(x)$, \sqrt{x} , $1/x$, truncation, or adding terms x^2 , x^3 , etc. Useful when observe nonlinear relationships.
- ▶ Add an interaction term: x_1x_2 when the effect of two predictors interacts.
- ▶ Transform the response: $\log(y)$, often useful when residuals show non-constant variance.

Example:



Intro to Logistic Regression

Logistic regression is a generalized linear model that is used to model binary response. The response, Y_i , takes the value 1 with probability p_i and the value 0 with probability $1 - p_i$.

- ▶ Instead of modeling Y_i 's directly, logistic regression models the probability p_i 's.
- ▶ The problem here is that the p_i 's only take values between 0 and 1. We use a logit transformation on the p_i 's to transform the values to real numbers.

$$\text{logit}(p_i) = \ln\left(\frac{p_i}{1 - p_i}\right)$$

- ▶ Then we fit a linear regression model to $\text{logit}(p_i)$.

$$\text{logit}(p_i) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$$

Logistic Regression, cont.

- ▶ Model outputs: We used statistical software to fit the logistic regression model just like multiple regression. The result is often presented in a summary table, from which you can find point estimates and test results for coefficients.

```
> fit=glm(parity~age+gestation,data=baby,family=binomial)
> summary(fit)
```



```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 1.385370  1.345130  1.030   0.3030
age         -0.186438  0.016461 -11.326  <2e-16 ***
gestation    0.008447  0.004553   1.855   0.0635 .
---
AIC: 1213.7
```

- ▶ Prediction: To convert from values on the logistic regression scale to probability, we have the following formula:

$$\hat{p}_i = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_k x_k}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_k x_k}}$$

e.g. For a mother of age 30 and gestation of 280 days, the predicted probability of that is not her first pregnancy is

$$\hat{p}_i = \frac{e^{1.385 - 0.186*30 + 0.008*280}}{1 + e^{1.385 - 0.186*30 + 0.008*280}} = 0.124$$

- ▶ Model selection: Just like multiple regression, we could perform forward or backward selection. We commonly use either p-value or AIC as model selection criteria.