BIOSTATS 640 – Introduction to R
Fall 2023
https://people.umass.edu/biep640w/webpages/demonstrations.html

Lesson 11
Introduction to Logistic Regression 1
November 17, 2023

Dataset used
titanic.xlsx

To commemorate those lost, a wreath floats in Southampton, England, in the same spot where the Titanic set sail on its ill-fated maiden voyage, April 10, 1912. "R.M.S." stands for "Royal Mail Ship," a designation given to the Titanic because it carried mail for the crown.

Source: https://www.nationalgeographic.co.uk/history-and-civilisation/2022/04/despite-the-warning-iceberg-right-ahead-the-titanic-was-doomed

		Page
1	The Titanic Datasets: **titanic.xlsx**	2
2	Highlights of Lesson 10 – Beautiful Tables	3
3	Prepare Data	5
4	Describe Data - Numerical	7
5	Fit Model	12
6	Working with a Fitted Model: Predicted Probabilities and Odds Ratios	15
7	How to Perform a Likelihood Ratio Test Comparison of Hierarchical Models	18
8	How to Show Models Side-by-Side Using `{stargazer}`	19

Packages used: `tidyverse, gtsummary, gt, Hmisc, compareGroups, lmtest, stargazer`

1. The Titanic Dataset

titanic.xlsx

Source: *https://en.wikipedia.org/wiki/Titanic*

The story of the RMS Titanic is famous. This ship was a passenger ship that sank in the Atlantic in 1912, after hitting an iceberg. There were over 2, 200 passengers and crew, of whom approximately 1500 died. The sinking of the RMS Titanic is known for being the deadliest sinking of a ship in peaceful times. You can download a dataset from the RMS Titanic here: (https://osf.io/aupb4/)

I have downloaded the dataset for you: ***titanic.xlsx***

Right click to download from the course website or Canvas. This dataset has n=1313 observations and p=7 variables. In this illustration, we will be using four variables:

Data Dictionary

Variable	Variable Label	Type	Codes	Missing Data
passenger_class	Passenger class	character	"1st" "2nd" "3rd"	none
survive	Survived	character	"yes" "no"	none
age	Age, years	numeric	Range: 0.17, 71	# missing = 680
sex	Sex	character	"female" "male"	none

2. Highlights of Lesson 10
Beautiful Tables

Using {gtsummary} and {gt} and {tidyverse}

Descriptives -Every Variable	**Basic**
	`tbl_summary(dataframe)`
	
	Aesthetics added
	`tbl_summary(mydata,`
	`type = all_dichotomous() ~ "categorical",`
	`statistic=list(`
	`all_continuous() ~ " {mean} ({sd})")`
	`) %>%`
	`bold_labels( ) %>%`
	`as_gt() %>%`
	`tab_header(title="Table 1. Baseline Characteristics")`

Descriptives - by Group	**Basic**
	`tbl_summary(dataframe, by=groupvariable)`
	
	Aesthetics added
	`tbl_summary(mydata, by=HT,`
	`type = all_dichotomous() ~ "categorical",`
	`missing_text = "(Missing)",`
	`statistic=list(`
	`all_continuous() ~ " {mean} ({sd})")`
	`) %>%`
	`add_p(test = all_continuous() ~ "t.test") %>%`
	`bold_labels() %>%`
	`as_gt() %>%`
	`tab_header(title="Baseline Characteristics, by Group")`

Good to Know	**Code these as options of tbl_summary( ):**
	`( dataframe,`
	`by = survivef,`
	`type = list(age ~ 'continuous2'),`
	`statistic = list(`
	`all_continuous() ~ c("{N_nonmiss",`
	`"{mean} ({sd})",`
	`"({min}, {max})")),`
	`percent = c("row"),`
	`digits = all_continuous() ~ 2,`
	`label = c(age ~ "Age, years",`
	`BMI ~ "Body Mass Index"),`
	`missing_text = "(Missing)" ) %>%`
	
	Code these as their own lines of code:
	`add_p(test = all_continuous() ~ "t.test") %>%`
	`bold_labels() %>%`
	
	These lines of code come AFTER `as_gt( ) %>%`
	`as_gt() %>%`
	`tab_header(title="Baseline Characteristics, by Group")`

Using {compareGroups} and {Hmisc} and {tidyverse}

Preliminary: Label variables	library(Hmisc) **Example** label(mydata$HT) <- "Hormone Therapy" label(mydata$age) <- "Age,years"
Descrptives - every variable	library(compareGroups) **Basic** descrTable(dataframe) **Aesthetics added (e.g., changing the column heading from "all")** mytable <- descrTable(dataframe) print(mytable, header.labels = c("all" = "My preferred title"))
Descriptives - by Group Note. Requires 2 steps.	library(compareGroups) **Step 1: Compute statistics** mytable <- compareGroups(groupvar ~ var1 + var2 + etc, data = mydata, byrow=TRUE) # default is column % **Step 2: Produced basic display and show** createTable(mytable) **Aesthetics added (e.g., changing p.overall** print(createTable(mytable), header.labels = c("p.overall" = "p-value"))
Export	export2pdf(tab, file = "example.pdf") export2xls(tab, file = "example.xlsx") export2word(tab, file = "example.docx") export2latex(tab, file = "example.tex")

A nice example	**Produce crude OR's for a Y=0/1 (survival)in relationship to several predictors**
	mystats <- compareGroups(data=keepdata2,
	survivef ~ age + age_quintilef + femalef + classf,
	byrow = TRUE, # Show row %
	fact.ratio = c(age=10)) # Show OR for +10 years
	
	mytable2 <- createTable(mystats, show.ratio = TRUE,
	show.p.overall = FALSE,
	type=2) # show freqs and %
	
	print(mytable2, header.labels = c(p.ratio = "p-value")) # change labeling

3. Prepare Data

initialize

```
setwd("/cloud/project")
options(scipen=999)                                                    # Turn off scientific notation
rm(list = ls())                                                        # Clear the Decks
```

import excel

```
library(readxl)
library(tidyverse)                                                     # glimpse() requires {tidyverse}
titanic <- read_excel("titanic.xlsx")
titanic <- as.data.frame(titanic)                                      # ggplot2 requires data frame
glimpse(titanic)                                                       # inspect structure of data
## Rows: 1,313
## Columns: 7
## $ passenger_class  <chr> "1st", "1st", "1st", "1st", "1st", "1st", "1st", "1st…
## $ name             <chr> "Allen,MissElisabethWalton", "Allison,MissHelenLorain…
## $ age              <dbl> 29.0000, 2.0000, 30.0000, 25.0000, 0.9167, 47.0000, 6…
## $ embarked         <chr> "Southampton", "Southampton", "Southampton", "Southam…
## $ home_destination <chr> "StLouis,MO", "Montreal,PQ/Chesterville,ON", "Montrea…
## $ sex              <chr> "female", "female", "male", "female", "male", "male",…
## $ survive          <chr> "yes", "no", "no", "no", "yes", "yes", "yes", "no", "…
```

quick look at distributions on every variable

```
summary(titanic)
##  passenger_class        name                age            embarked
##  Length:1313        Length:1313        Min.   : 0.1667   Length:1313
##  Class :character   Class :character   1st Qu.:21.0000   Class :character
##  Mode  :character   Mode  :character   Median :30.0000   Mode  :character
##                                        Mean   :31.1942
##                                        3rd Qu.:41.0000
##                                        Max.   :71.0000
##                                        NA's   :680
##  home_destination       sex              survive
##  Length:1313        Length:1313        Length:1313
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
```

prepare data

Tip!Factor variables use storage 1, 2, etc (Not 0/1)
RECOMMENDED use of 0/1 coding: Use 1=exposure and 0=NOT indicator variables for performing analyses (fitting models),
RECOMMENDED use of 1/2 coding: Use 1=exposure and 2=NOT for producing pretty tables.

```
library(tidyverse)

ready <- titanic                                                    # initalize

# survive
ready <- ready %>%
        mutate(survivef = recode_factor(survive,                    # factor var survivef
                        "no" = "Died",                                    # "Died" is stored as 1
                        "yes" = "Survived")) %>%                          # "Survived" is stored as 2
        mutate(y_died = ifelse(survivef=="Died",1,0))               # 0/1 numeric for modeling
ready$y_died[is.na(ready$survive)] <- NA                            # handle missings explicitly
ready$survivef[is.na(ready$survive)] <- NA

# passenger_class
ready <- ready %>%
        mutate(classf = recode_factor(passenger_class,               # factor for descriptives
                        "3rd" = "3rd Class",
                        "2nd" = "2nd Class",
                        "1st" = "1st Class")) %>%
        mutate(class_2nd= ifelse(classf=="2nd Class", 1, 0)) %>%     # 0/1 for modeling
        mutate(class_3rd= ifelse(classf=="3rd Class", 1, 0))
ready$classf[is.na(ready$passenger_class)] <- NA
ready$class_2nd[is.na(ready$passenger_class)] <- NA
ready$class_3rd[is.na(ready$passenger_class)] <- NA


# sex
ready <- ready %>%
        mutate(femalef = recode_factor(sex,                          # factor var femalef
                        "Female" = "female",
                        "Male" = "male")) %>%
        mutate(female = ifelse(femalef=="female",1,0))               # 0/1 female, numeric
ready$femalef[is.na(ready$sex)] <- NA
ready$female[is.na(ready$sex)] <- NA

# age - create quintiles of age and 0/1 indicators
quantile(ready$age, probs = c(0, .20, .40, .60, .80, 1.0), na.rm = TRUE)     # get quintiles for use in labeling
##      0%     20%     40%     60%     80%    100%
##  0.1667 19.0000 26.0000 33.0000 45.0000 71.0000

ready <- ready %>%
      mutate(age_quintile = ntile(age,5), na.rm=T)                   # ntile( ) creates var with values 1, 2,...5
table(ready$age_quintile)                                            # check.
##
##   1   2   3   4   5
## 127 127 127 126 126

ready <- ready %>%
  mutate(age_quintilef = recode_factor(age_quintile,                 # age_quintilef, factor
      "1" = "Q1:0-19",                                               # "old" = "new"
      "2" = "Q2:19-26",
      "3" = "Q3:26-33",
      "4" = "Q4:33-45",
      "5" = "Q5:45-71"))
```

```
table(ready$age_quintilef)
##  Q1:0-19 Q2:19-26 Q3:26-33 Q4:33-45 Q5:45-71
##      127      127      127      126      126

ready <- ready %>%
 mutate(age_Q2 = ifelse(age_quintile==2,1,0)) %>%                          # 0/1 DESIGN variables
 mutate(age_Q3 = ifelse(age_quintile==3,1,0)) %>%
 mutate(age_Q4 = ifelse(age_quintile==4,1,0)) %>%
 mutate(age_Q5 = ifelse(age_quintile==5,1,0))

# Retain vars of interest
ready <- ready %>%
         select(y_died,survivef,classf,class_2nd,class_3rd,
                femalef, female,
                age,age_quintilef, age_Q2, age_Q3, age_Q4, age_Q5)
glimpse(ready)

## Rows: 1,313
## Columns: 13
## $ y_died        <dbl> 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0,…
## $ survivef      <fct> Survived, Died, Died, Died, Survived, Survived, Survived…
## $ classf        <fct> 1st Class, 1st Class, 1st Class, 1st Class, 1st Class, 1…
## $ class_2nd     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ class_3rd     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ femalef       <fct> female, female, male, female, male, male, female, male, …
## $ female        <dbl> 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0,…
## $ age           <dbl> 29.0000, 2.0000, 30.0000, 25.0000, 0.9167, 47.0000, 63.0…
## $ age_quintilef <fct> Q3:26-33, Q1:0-19, Q3:26-33, Q2:19-26, Q1:0-19, Q5:45-71…
## $ age_Q2        <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, NA, NA, NA, 0, 1, 0,…
## $ age_Q3        <dbl> 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, NA, NA, NA, 0, 0, 0,…
## $ age_Q4        <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, NA, NA, NA, 0, 0, 1,…
## $ age_Q5        <dbl> 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, NA, NA, NA, 1, 0, 0,…
```

4. Describe Data - Numerical

```
describe data - quick check
summary(ready,digits=2)
##      y_died          survivef         classf      class_2nd      class_3rd
##  Min.   :0.00   Died    :864   3rd Class:711   Min.   :0.00   Min.   :0.00
##  1st Qu.:0.00   Survived:449   2nd Class:280   1st Qu.:0.00   1st Qu.:0.00
##  Median :1.00                  1st Class:322   Median :0.00   Median :1.00
##  Mean   :0.66                                  Mean   :0.21   Mean   :0.54
##  3rd Qu.:1.00                                  3rd Qu.:0.00   3rd Qu.:1.00
##  Max.   :1.00                                  Max.   :1.00   Max.   :1.00
##
##    femalef        female          age         age_quintilef     age_Q2
##  female:463   Min.   :0.00   Min.   : 0.17   Q1:0-19 :127   Min.   :0.0
##  male  :850   1st Qu.:0.00   1st Qu.:21.00   Q2:19-26:127   1st Qu.:0.0
##               Median :0.00   Median :30.00   Q3:26-33:127   Median :0.0
##               Mean   :0.35   Mean   :31.19   Q4:33-45:126   Mean   :0.2
##               3rd Qu.:1.00   3rd Qu.:41.00   Q5:45-71:126   3rd Qu.:0.0
##               Max.   :1.00   Max.   :71.00   NA's    :680   Max.   :1.0
##                              NA's   :680                    NA's   :680
##      age_Q3         age_Q4         age_Q5
##  Min.   :0.0   Min.   :0.0   Min.   :0.0
##  1st Qu.:0.0   1st Qu.:0.0   1st Qu.:0.0
##  Median :0.0   Median :0.0   Median :0.0
##  Mean   :0.2   Mean   :0.2   Mean   :0.2
##  3rd Qu.:0.0   3rd Qu.:0.0   3rd Qu.:0.0
##  Max.   :1.0   Max.   :1.0   Max.   :1.0
##  NA's   :680   NA's   :680   NA's   :680
```

describe data, grouped by survival - {summarytools}

```
library(summarytools)
# age
with(ready,                                                    # with(dataframe,
 stby(data = age,                                                    # data = continuous var
 INDICES =survivef,                                                  # INDICES = group var
 FUN = descr,
 stats=c("n.valid", "pct.valid", "mean", "sd", "min", "max"),
 transpose=TRUE))

## Descriptive Statistics
## age by survivef
## Data Frame: ready
## N: 864
##
##                  N.Valid   Pct.Valid    Mean   Std.Dev    Min     Max
## -------------- --------- ----------- ------- --------- ------ -------
##           Died    352.00       40.74   32.25     14.04   0.33   71.00
##       Survived    281.00       62.58   29.87     15.52   0.17   69.00

# classf, femalef,age_quintilef
ctable(x = ready$classf, y = ready$survivef, prop = "r")

## Cross-Tabulation, Row Proportions
## classf * survivef
## Data Frame: ready
##
## ----------- ---------- ------------- ------------- ---------------
##               survivef          Died      Survived           Total
##      classf
##   3rd Class              574 (80.7%)   137 (19.3%)    711 (100.0%)
##   2nd Class              161 (57.5%)   119 (42.5%)    280 (100.0%)
##   1st Class              129 (40.1%)   193 (59.9%)    322 (100.0%)
##       Total              864 (65.8%)   449 (34.2%)   1313 (100.0%)
## ----------- ---------- ------------- ------------- ---------------

ctable(x = ready$femalef, y = ready$survivef, prop = "r")

## Cross-Tabulation, Row Proportions
## femalef * survivef
## Data Frame: ready
##
## --------- ---------- ------------- ------------- ---------------
##             survivef          Died      Survived           Total
##   femalef
##    female              156 (33.7%)   307 (66.3%)    463 (100.0%)
##      male              708 (83.3%)   142 (16.7%)    850 (100.0%)
##     Total              864 (65.8%)   449 (34.2%)   1313 (100.0%)
## --------- ---------- ------------- ------------- ---------------

ctable(x = ready$age_quintilef, y = ready$survivef, prop = "r")

## Cross-Tabulation, Row Proportions
## age_quintilef * survivef
## Data Frame: ready
##
## --------------- ---------- ------------- ------------- ---------------
##                   survivef          Died      Survived           Total
##   age_quintilef
##         Q1:0-19               53 (41.7%)    74 (58.3%)    127 (100.0%)
##        Q2:19-26               78 (61.4%)    49 (38.6%)    127 (100.0%)
##        Q3:26-33               81 (63.8%)    46 (36.2%)    127 (100.0%)
##        Q4:33-45               71 (56.3%)    55 (43.7%)    126 (100.0%)
##        Q5:45-71               69 (54.8%)    57 (45.2%)    126 (100.0%)
##            <NA>              512 (75.3%)   168 (24.7%)    680 (100.0%)
##           Total              864 (65.8%)   449 (34.2%)   1313 (100.0%)
## --------------- ---------- ------------- ------------- ---------------
```

describe data, grouped by survival - {gtsummary} and {gt}

```
library(tidyverse)
library(Hmisc)
library(gtsummary)
library(gt)

# vars to be displayed ONLY
keepdata <- ready %>%
            select(survivef,age,age_quintilef,femalef,classf)

# basic - no frills
keepdata %>% tbl_summary(by=survivef)
```

Characteristic	Died, N = 864[1]	Survived, N = 449[1]
age	30 (22, 41)	29 (19, 40)
Unknown	512	168
age_quintilef		
Q1:0-19	53 (15%)	74 (26%)
Q2:19-26	78 (22%)	49 (17%)
Q3:26-33	81 (23%)	46 (16%)
Q4:33-45	71 (20%)	55 (20%)
Q5:45-71	69 (20%)	57 (20%)
Unknown	512	168
femalef		
female	156 (18%)	307 (68%)
male	708 (82%)	142 (32%)
classf		
3rd Class	574 (66%)	137 (31%)
2nd Class	161 (19%)	119 (27%)
1st Class	129 (15%)	193 (43%)

[1]Median (IQR); n (%)

```
# with frills (parentheses are so tricky!)
keepdata %>% tbl_summary(                          # tbl_summary( ) BEGIN
    by = survivef,
    type = list(age ~ 'continuous2'),
    statistic = list(                                        # list( ) BEGIN
                all_continuous() ~ c("{N_nonmiss}",
                                     "{mean} ({sd})",
                                     "({min}, {max})")
                ),                                            # list( ) END
    percent = c("row"),
    digits = all_continuous() ~ 2,
    label = c(age ~ "Age, years",
              age_quintilef ~ "Age Quintile, years",
              femalef ~ "Sex at birth",
              classf ~ "Passenger Class"),
    missing_text = "(Missing)"
  ) %>%                                           # tbl_summary( ) END
add_p(test = all_continuous() ~ "t.test") %>%
bold_labels() %>%
as_gt() %>%
tab_header(title="Titanic Data (n=1313)")
```

Dear reader,
The table on the next page did not knit, whereas the one on this page did knit. I think the 2nd table would have knit properly if I had put it in its own chunk. Sorry about that! - cb.

Titanic Data (n=1313)			
Characteristic	**Died**, N = 864[1]	**Survived**, N = 449[1]	**p-value**[2]
Age, years			0.047
N	352.00	281.00	
Mean (SD)	32.25 (14.04)	29.87 (15.52)	
(Range)	(0.33, 71.00)	(0.17, 69.00)	
(Missing)	512	168	
Age Quintile, years			0.004
Q1:0-19	53 (42%)	74 (58%)	
Q2:19-26	78 (61%)	49 (39%)	
Q3:26-33	81 (64%)	46 (36%)	
Q4:33-45	71 (56%)	55 (44%)	
Q5:45-71	69 (55%)	57 (45%)	
(Missing)	512	168	
Sex at birth			<0.001
female	156 (34%)	307 (66%)	
male	708 (83%)	142 (17%)	
Passenger Class			<0.001
3rd Class	574 (81%)	137 (19%)	
2nd Class	161 (57%)	119 (42%)	
1st Class	129 (40%)	193 (60%)	

[1] n (%)
[2] Welch Two Sample t-test; Pearson's Chi-squared test

describe data, grouped by survival with display of ODDS RATIO - {compareGroups}

```
library(tidyverse)
library(compareGroups)
library(Hmisc)

# vars to be displayed ONLY
keepdata2 <- ready %>%
            select(survivef,age,age_quintilef,femalef,classf) %>%
            na.omit()                                              # complete data only

# Tip for {compareGroups}:  Set referent category = 1 for EVERY factor variable
keepdata2$survivef <- factor(keepdata2$survivef,
                             levels=c("Survived", "Died"),            # referent = Survived
                             labels=c("Survived", "Died"))
keepdata2$femalef <- factor(keepdata2$femalef,
                             levels=c("male", "female"),              # referent = male
                             labels=c("male", "female"))
keepdata2$classf <- factor(keepdata2$classf,
                             levels=c("1st Class", "2nd Class", "3rd Class"),
                             labels=c("1st Class", "2nd Class", "3rd Class"))

# label variables (Tip - best to do labeling as last step in variable creation)
label(keepdata2$survivef) <- "Survived"
label(keepdata2$age) <- "Age, per 10 years"
label(keepdata2$age_quintilef) <- "Age Quintile, years"
label(keepdata2$femalef) <- "Sex at Birth"
label(keepdata2$classf) <- "Passenger Class"


mystats <- compareGroups(data=keepdata2,
                       survivef ~ age + age_quintilef + femalef + classf,
                       byrow = TRUE,                              # Show row % (default is column)
                       fact.ratio = c(age=10))                    # Show OR for +10 years (nice!)

mytable2 <- createTable(mystats, show.ratio = TRUE,
                        show.p.overall = FALSE,
                        type=2)                                   # show frequencies and relative frequencies

print(mytable2, header.labels = c(p.ratio = "p-value"))        # change labeling of p-value
```

```
##
## --------Summary descriptives table by 'Survived'---------
##
## ____________________________________________________________________
##                        Survived      Died            OR        p-value
##                          N=281       N=352
## --------------------------------------------------------------------
## Age, per 10 years    29.9 (15.5) 32.2 (14.0) 1.12 [1.00;1.24]  0.045
## Age Quintile, years:
##     Q1:0-19          74 (58.3%)  53 (41.7%)        Ref.        Ref.
##     Q2:19-26         49 (38.6%)  78 (61.4%)  2.21 [1.34;3.68]  0.002
##     Q3:26-33         46 (36.2%)  81 (63.8%)  2.45 [1.48;4.08] <0.001
##     Q4:33-45         55 (43.7%)  71 (56.3%)  1.80 [1.09;2.97]  0.021
##     Q5:45-71         57 (45.2%)  69 (54.8%)  1.69 [1.03;2.79]  0.039
## Sex at Birth:
##     male             82 (21.0%)  308 (79.0%)       Ref.        Ref.
##     female           199 (81.9%) 44 (18.1%)  0.06 [0.04;0.09]  0.000
## Passenger Class:
##     1st Class        139 (61.5%) 87 (38.5%)        Ref.        Ref.
##     2nd Class        96 (45.3%)  116 (54.7%) 1.93 [1.32;2.83]  0.001
##     3rd Class        46 (23.6%)  149 (76.4%) 5.14 [3.38;7.94] <0.001
## --------------------------------------------------------------------
```

5. Fit Model

Recall. In normal theory regression, we *directly* model $E\left[Y_{\text{subpopulation with X=x}} \right] = \beta_0 + \beta_1 \cdot x$

We are doing a regression analysis of the means of sub-populations of the outcome Y, where the subpopulations are defined according to the value of the predictor X.

In logistic regression, we model *a function of* $\left\{ E\left[Y_{\text{subpopulation with X=x}} \right] \right\} = \beta_0 + \beta_1 \cdot x$

We are doing a regression analysis of the means of sub-populations of the outcome Y, where the subpopulations are defined according to the value of the predictor X.

The particular function that links E [Y] to the linear model on the right hand side is the logit function

(1) **Start with $E[Y_{\text{subpopulation with X=x}}]$ for a binary outcome**

In this settting, where there are just two outcomes (Y=1 or Y=0):
For each subpopulation defined by X=x,

$Y_{X=x}$ = is a ***Bernoulli*** random variable with probability of event = $p_{X=x}$,
Recall from Unit 2, this means that

$E[Y_{\text{subpopulation with X=x}}] = p_{X=x} = \Pr[Y = 1 \text{ in subpopulation with X=x}]$

(2) **How we get to the "logit" function**

$$function\left\{ E\left[Y_{subpopulation\ with\ X=x} \right] \right\}$$

$$= function\left\{ Pr\left[Y=1_{\ subpopulation\ with\ X=x} \right] \right\}$$

$$= function\left\{ p_{subpopulation\ with\ X=x} \right\}$$

$$= \ln\left\{ \frac{\left(p_{subpopulation\ with\ X=x} \right)}{1 - \left(p_{subpopulation\ with\ X=x} \right)} \right\}$$

$$= \log odds\left\{ \left(p_{subpopulation\ with\ X=x} \right) \right\}$$

$$= logit\left\{ \left(p_{subpopulation\ with\ X=x} \right) \right\}$$

Logistic Regression Model "logit" link from E [Y] to the linear model

$$\left\{ logit\left[Pr\left(Y_{\text{subpopulation with X=x}} \right) = 1 \right] \right\} = \beta_0 + \beta_1 \cdot x$$

fit model

```
library(tidyverse)

# IMPORTANT!!!  For comparing models, make sure you are working with complete data only
complete <- ready %>% na.omit()                                  # complete data only

fit <- glm(y_died ~ class_2nd + class_3rd + female +             # glm() to fit a logistic regression
             age_Q2 + age_Q3 + age_Q4 + age_Q5,
             family=binomial,                                    # must have family=binomial for logistic regression
             data=complete)

## fit()
fit                                                 # display - basic

##
## Call:  glm(formula = y_died ~ class_2nd + class_3rd + female + age_Q2 +
##     age_Q3 + age_Q4 + age_Q5, family = binomial, data = complete)
##
## Coefficients:
## (Intercept)    class_2nd    class_3rd       female       age_Q2       age_Q3
##     -0.8553       1.2210       2.6148      -2.9984       0.9996       1.3401
##      age_Q4       age_Q5
##      1.3051       1.7914
##
## Degrees of Freedom: 632 Total (i.e. Null);  625 Residual
## Null Deviance:       869.5
## Residual Deviance: 548.7     AIC: 564.7

## summary(fit)
summary(fit)                                        # display - more detailed

##
## Call:
## glm(formula = y_died ~ class_2nd + class_3rd + female + age_Q2 +
##     age_Q3 + age_Q4 + age_Q5, family = binomial, data = complete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6755  -0.5974   0.3020   0.6744   2.7837
##
## Coefficients:
##             Estimate Std. Error z value             Pr(>|z|)
## (Intercept)  -0.8553     0.3311  -2.583             0.009782 **
## class_2nd     1.2210     0.2742   4.452  0.00000849866342489 ***
## class_3rd     2.6148     0.3322   7.871  0.00000000000000353 ***
## female       -2.9984     0.2370 -12.653 < 0.0000000000000002 ***
## age_Q2        0.9996     0.3512   2.846             0.004426 **
## age_Q3        1.3401     0.3541   3.784             0.000154 ***
## age_Q4        1.3051     0.3557   3.669             0.000243 ***
## age_Q5        1.7914     0.3795   4.720  0.00000235401724048 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 869.54  on 632  degrees of freedom
## Residual deviance: 548.66  on 625  degrees of freedom
## AIC: 564.66
##
## Number of Fisher Scoring iterations: 5
```

```
names(fit)                                                    # names( ) to list objects stored with fit
```

```
##  [1] "coefficients"      "residuals"         "fitted.values"
##  [4] "effects"           "R"                 "rank"
##  [7] "qr"                "family"            "linear.predictors"
## [10] "deviance"          "aic"               "null.deviance"
## [13] "iter"              "weights"           "prior.weights"
## [16] "df.residual"       "df.null"           "y"
## [19] "converged"         "boundary"          "model"
## [22] "call"              "formula"           "terms"
## [25] "data"              "offset"            "control"
## [28] "method"            "contrasts"         "xlevels"
```

```
names(summary(fit))                                          # names() to list objects stored with summary(fit)
```

```
##  [1] "call"           "terms"          "family"         "deviance"
##  [5] "aic"            "contrasts"      "df.residual"    "null.deviance"
##  [9] "df.null"        "iter"           "deviance.resid" "coefficients"
## [13] "aliased"        "dispersion"     "df"             "cov.unscaled"
## [17] "cov.scaled"
```

6. Working with a Fitted Model: Predicted Probabilities and Odds Ratios

Predicted Pr [Y = 1]$_{\text{at X=x}}$ = p$_{\text{X=x}}$

$$\widehat{p_{X=x}} = \frac{exp\left\{ \widehat{\beta_0} + \widehat{\beta_1} \cdot x \right\}}{1 + exp\left\{ \widehat{\beta_0} + \widehat{\beta_1} \cdot x \right\}}$$

$$= \frac{exp\left\{ \widehat{\text{logit}}_{X=x} \right\}}{1 + exp\left\{ \widehat{\text{logit}}_{X=x} \right\}}$$

working with a fitted model - predicted probabilities

```
# EXAMPLE: Obtain predicted probabilities by passenger class, holding all other vars at their means

# Step 1:  mydata = 3 "new" observations (on 8 variables) for which predicted probabilities are desired
mydata <- data.frame(classf=c("1st Class", "2nd Class", "3rd Class"),      # we want predictions for each class
                     class_2nd=c(0,1,0),                                          # 2nd class indicator values
                     class_3rd=c(0,0,1),                                          # 3rd class indicator values
                     age_Q2=rep(mean(complete$age_Q2),3),                 # all other vars at means
                     age_Q3=rep(mean(complete$age_Q3),3),
                     age_Q4=rep(mean(complete$age_Q4),3),
                     age_Q5=rep(mean(complete$age_Q5),3),
                     female=rep(mean(complete$female),3))


## New data for which predicted probabilities are desired
mydata
##      classf class_2nd class_3rd    age_Q2    age_Q3    age_Q4    age_Q5
## 1 1st Class         0         0 0.2006319 0.2006319 0.1990521 0.1990521
## 2 2nd Class         1         0 0.2006319 0.2006319 0.1990521 0.1990521
## 3 3rd Class         0         1 0.2006319 0.2006319 0.1990521 0.1990521
##      female
## 1 0.3838863
## 2 0.3838863
## 3 0.3838863

# Step 2:  mypredicted = mydata + cbind( ) of predicted probabilities and their standard errors appended
mypredicted <- cbind(mydata,predict(fit,newdata=mydata,type="response", se.fit=TRUE))

# Step 3:  Show
myvars <- c("classf","fit","se.fit")
mypredicted[myvars]

##      classf       fit     se.fit
## 1 1st Class 0.2848481 0.03975811
## 2 2nd Class 0.5745451 0.04427310
## 3 3rd Class 0.8447831 0.03096507
```

Estimated Adjusted Odds Ratio, per 1 unit increase in predictor
Comparison (X=x+1) versus Referent (X=x)

$$\widehat{OR}_{X=x+1 \; versus \; X=x} = exp\{\hat{\beta}\}$$

Estimated Adjusted Odds Ratio - General
Comparison "1": $X_1 = x_{11}$ and $X_2 = x_{21}$
Referent "0": $X_1 = x_{10}$ and $X_2 = x_{20}$

$$\widehat{OR}_{"1"=comparison \; versus \; "0"=referent} = exp\left\{\widehat{\text{logit}}_{"1"=comparison} - \widehat{\text{logit}}_{"0"=referent}\right\}$$

```
working with a fitted model - predicted odds ratios
##
## Estimated Relative Odds (OR) Event per 1 unit increase in each predictor with associated 95% CI
exp(cbind(OR=coefficients(fit),confint(fit)))

##                      OR      2.5 %      97.5 %
## (Intercept)  0.42516341 0.22015270  0.80827260
## class_2nd    3.39043535 1.99591418  5.85856994
## class_3rd   13.66441563 7.25568734 26.75745001
## female       0.04986772 0.03085836  0.07827703
## age_Q2       2.71728554 1.37405268  5.46006282
## age_Q3       3.81950111 1.92466509  7.73409703
## age_Q4       3.68794234 1.85275271  7.49073602
## age_Q5       5.99761828 2.88342936 12.79846620

# User specified comparison (person 1) versus reference profile (person 0)
# METHOD I - by hand
# Step 1:  Create "comparison" and "referent" profiles as data.frame objects- person1 and person0
person1 <- data.frame(class_2nd=0,class_3rd=1,                          # 3rd class
                      female=1,                                         # female
                      age_Q2=0, age_Q3=0, age_Q4=0, age_Q5=1)           # age 45-71

person0 <- data.frame(class_2nd=0,class_3rd=0,                          # 1st class
                      female=0,                                         # male
                      age_Q2=1, age_Q3=0, age_Q4=0, age_Q5=0)           # age 19-26
```

```
# Step 2:  Get prediced probabilities. Show
phat1 <- predict(fit,newdata=person1,type="response")
phat0 <- predict(fit,newdata=person0,type="response")
phat1
## 0.6347141
phat0
## 0.5360254

# Step 2:  Get prediced odds and odds ratio.  Show odds ratio
odds1 <- phat1/(1-phat1)
odds0 <- phat0/(1-phat0)
oddsratio_method1 <- odds1/odds0
oddsratio_method1
## 1.504022

# User specified comparison (person 1) versus reference profile (person 0)
# METHOD II - using predict( ) with option type="link"
logit1 <- predict(fit,newdata=person1,type="link")
logit0 <- predict(fit,newdata=person0,type="link")


## Estimated OR comparison v referent =
oddsratio_method2 <- exp(logit1 - logit0)
oddsratio_method2
## 1.504022
```

Tip - Good to know> Obtain predicted probabilities, odds and logits for new data frame with multiple observations

```
# Step 1 - create new data
# This new data frame has n=5 observations on the predictors used in the model.  Show
# These individuals are hypothetical FUTURE passengers
mynew_data <- data.frame(class_2nd=c(1,0,1,1,1),
                         class_3rd=c(0,0,0,0,0),
                         female=c(0,0,0,1,1),
                         age_Q2=c(0,0,0,0,1),
                         age_Q3=c(1,1,1,1,0),
                         age_Q4=c(0,0,0,0,0),
                         age_Q5=c(0,0,0,0,0))
mynew_data

##   class_2nd class_3rd female age_Q2 age_Q3 age_Q4 age_Q5
## 1         1         0      0      0      1      0      0
## 2         0         0      0      0      1      0      0
## 3         1         0      0      0      1      0      0
## 4         1         0      1      0      1      0      0
## 5         1         0      1      1      0      0      0

# Step 2 - obtain predicted probability, odds, log-odds/logit
mynew_data$phat <- predict(fit,newdata=mynew_data,type="response")
mynew_data$odds <- mynew_data$phat/(1 - mynew_data$phat)
mynew_data$logit <- log(mynew_data$odds)                                    # log() for natural logarithm
mynew_data

##   class_2nd class_3rd female age_Q2 age_Q3 age_Q4 age_Q5      phat      odds
## 1         1         0      0      0      1      0      0 0.8462903 5.5057690
## 2         0         0      0      0      1      0      0 0.6188897 1.6239121
## 3         1         0      0      0      1      0      0 0.8462903 5.5057690
## 4         1         0      1      0      1      0      0 0.2154156 0.2745602
## 5         1         0      1      1      0      0      0 0.1634101 0.1953287
##        logit
## 1  1.7057964
## 2  0.4848381
## 3  1.7057964
## 4 -1.2925849
## 5 -1.6330713
```

7. How to Perform a Likelihood Ratio Test Comparison of Hierarchical Models

```
likelihood ratio test for hierarchial models
library(lmtest)

# Controlling for passenger class and sex at birth, is age statistically significant?

reduced <- glm(y_died ~ class_2nd + class_3rd + female,                # reduced
          data=complete,
          family=binomial)

full <- glm(y_died ~ class_2nd + class_3rd + female +                   # full
                     age_Q2 + age_Q3 + age_Q4 + age_Q5,
          data=complete,
          family=binomial)

lmtest::lrtest(reduced, full)

## Likelihood ratio test
##
## Model 1: y_died ~ class_2nd + class_3rd + female
## Model 2: y_died ~ class_2nd + class_3rd + female + age_Q2 + age_Q3 + age_Q4 + age_Q5
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   4 -287.63
## 2   8 -274.33  4 26.588 0.00002408 ***                                  The null is rejected (p-value << .0001)
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

8. How to Show Models Side-by-Side Using {stargazer}

```
stargazer for side-by-side comparison of models
library(stargazer)

# fit to passenger class only
m_class <- glm(y_died ~ class_2nd + class_3rd,
          data=complete,
          family=binomial)

# fit to quintile of age only
m_age <- glm(y_died ~ age_Q2 + age_Q3 + age_Q4 + age_Q5,
          data=complete,
          family=binomial)

# TABLE 1:  Side by side BETA (SE):  Basic
stargazer(m_class, m_age, type="text",
          title="TABLE 1:  Betas")

##
## TABLE 1: Betas
## ================================================
##                        Dependent variable:
##                    ----------------------------
##                               y_died
##                          (1)            (2)
## ------------------------------------------------
## class_2nd              0.658***
##                        (0.194)
##
## class_3rd              1.644***
##                        (0.217)
##
## age_Q2                                0.799***
##                                       (0.256)
##
## age_Q3                                0.900***
##                                       (0.258)
##
## age_Q4                                 0.589**
##                                       (0.254)
##
## age_Q5                                 0.525**
##                                       (0.254)
##
## Constant              -0.469***        -0.334*
##                        (0.137)         (0.180)
##
## ------------------------------------------------
## Observations             633              633
## Log Likelihood        -403.147         -427.201
## Akaike Inf. Crit.     812.294          864.402
## ================================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

```
# TABLE 2: Side by side OR and SE(OR) with display of odds ratios (OR)

# Step 1:  get OR's for model objects m_class and m_age
or_class <- exp(coef(m_class))
or_age <- exp(coef(m_age))


# Step 2:  get SE of betas (stargazer default is not correct)
sebeta_class <- sqrt(diag(vcov(m_class)))
sebeta_age <- sqrt(diag(vcov(m_age)))
# get SE of odds ratios
seOR_class <-  or_class * sebeta_class
seOR_age <-  or_age * sebeta_age

stargazer(m_class, m_age,
          coef=list(or_class, or_age),
          se = list(seOR_class,seOR_age),
          t.auto=F, p.auto=F,
          type="text",
          title="TABLE 2:  Odds Ratios (SE)")

##
## TABLE 2: Odds Ratios (SE)
## ================================================
##                        Dependent variable:
##                    ----------------------------
##                              y_died
##                         (1)            (2)
## ------------------------------------------------
## class_2nd             1.931***
##                       (0.375)
##
## class_3rd             5.175***
##                       (1.124)
##
## age_Q2                                2.223***
##                                        (0.569)
##
## age_Q3                                2.459***
##                                        (0.634)
##
## age_Q4                                 1.802**
##                                        (0.458)
##
## age_Q5                                 1.690**
##                                        (0.429)
##
## Constant              0.626***         0.716*
##                       (0.086)          (0.129)
##
## ------------------------------------------------
## Observations            633              633
## Log Likelihood        -403.147        -427.201
## Akaike Inf. Crit.     812.294          864.402
## ================================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

```
# TABLE 3: Side by side odds ratio (OR) and 95% CI

# Step 1:  get 95% CI
CIOR_class <- as.matrix(exp(confint.default(m_class)))
CIOR_age <- as.matrix(exp(confint.default(m_age)))
stargazer(m_class, m_age,
          coef=list(or_class, or_age),
          ci.custom = list(CIOR_class, CIOR_age),
          ci=TRUE,
          t.auto=F, p.auto=F,
          type="text",
          title="TABLE 3:  Odds Ratios (95% CI)")

##
## TABLE 3: Odds Ratios (95% CI)
## ================================================
##                         Dependent variable:
##                     ---------------------------
##                                y_died
##                           (1)            (2)
## ------------------------------------------------
## class_2nd              1.931***
##                     (1.319, 2.825)
##
## class_3rd              5.175***
##                     (3.382, 7.920)
##
## age_Q2                                 2.223***
##                                     (1.345, 3.672)
##
## age_Q3                                 2.459***
##                                     (1.483, 4.075)
##
## age_Q4                                 1.802**
##                                     (1.095, 2.967)
##
## age_Q5                                 1.690**
##                                     (1.028, 2.780)
##
## Constant               0.626***        0.716*
##                     (0.479, 0.818) (0.503, 1.019)
##
## ------------------------------------------------
## Observations             633             633
## Log Likelihood         -403.147        -427.201
## Akaike Inf. Crit.      812.294         864.402
## ================================================
## Note:                *p<0.1; **p<0.05; ***p<0.01
```