Lab 2 BIS 505b

Maria Ciarleglio

2/15/2021

Goal of Lab 2

In **Lab 2**, we will review **(1)** statistical functions in **R** used with common probability distributions, **(2)** one-sample and two-sample confidence intervals (CI) and hypothesis tests for means, and **(3)** one-sample and two-sample confidence intervals and hypothesis tests for proportions.

The Framingham Heart Study data contained in `fhs_exam1.csv` will be used in this Lab. We will analyze a subset of the full `fhs` data frame that consists of the first 100 rows called `fhs100`. Let's begin by creating `fhs100` and defining the factor variables that we will use in this Lab: the indicator of being overweight or obese (`OVERWEIGHTOBESE`), `SEX`, and presence of cardiovascular disease (`CVD`).

Note: When we create summary tables, the levels of a factor will be displayed in the order they are specified in the `levels=` argument of the `factor()` function. Since we like to observe "Yes" before "No" in our tables, we will specify the levels of `OVERWEIGHTOBESE_factor` and `CVD_factor` as `levels = c(1, 0)` so that `1` (Yes) will be displayed before `0` (No).

```
# Select first 100 rows from full fhs data frame for analysis
fhs100 <- fhs[1:100,]
dim(fhs100)
```

```
## [1] 100  36
```

```
# Coding Overweight/Obese indicator
fhs100$OVERWEIGHTOBESE <- ifelse(fhs100$BMI >= 25, 1, 0)  # if BMI>=25 is true, OVERWEIGHTOBESE = 1
                                                          # if BMI>=25 is false, OVERWEIGHTOBESE =
 0

# Adding a factor version of our Overweight/Obese indicator and Sex to fhs100 data frame
fhs100 <- dplyr::mutate(fhs100,
                        OVERWEIGHTOBESE_factor = factor(OVERWEIGHTOBESE,
                                                 levels = c(1, 0),
                                                 labels = c("Yes", "No")),
                        SEX_factor = factor(SEX,
                                            levels = c(1, 2),
                                            labels = c("Male", "Female")),
                        CVD_factor = factor(CVD,
                                            levels = c(1, 0),
                                            labels = c("Yes", "No")))
```

General Structure of Statistical Distribution Functions in **R**

Continuous probability distributions are described by a **probability density function** (pdf), $f(x)$. The **cumulative distribution function** (cdf) of a random variable X is defined as $P(X \leq x)$ and is denoted by $F(x)$.

There are **four** types of functions available in **R** associated with probability distributions:

- `dname` - Calculates the probability density function (pdf) of continuous random variable X at input x, $f(x)$; [`d` = density]
- `pname` - Calculates the cdf of random variable X at input x (i.e., $F(x) = P(X \leq x)$); [`p` = probability]; **used to find p-values**
- `qname` - Calculates the inverse cdf (i.e., the quantile/percentile q_{pr} of random variable X where $P(X \leq q_{pr}) = pr$); [`q` = quantile]; **used to find critical values**
- `rname` - Generates a random value from the indicated distribution; [`r` = random]

To work with different probability distributions, replace `name` with the name of the distribution of interest. For example, the Normal distribution uses `name = norm`. The `dnorm()` function returns the probability density function; the `pnorm()` function returns the cdf; the `qnorm()` function returns the quantile; the `rnorm()` function generates random values from a Normal distribution.

For a **continuous probability distribution** such as the Normal, the most useful functions are the `p` and `q` functions. The `d` function returns the pdf, which must be integrated to find Normal probabilities.

Each distribution has **parameters** that must be specified that define the distribution. For example, for a normally distributed random variable $X \sim N(\mu, \sigma)$, we must specify the mean μ and standard deviation σ. For a t random variable $X \sim t_{df}$, we must specify the degrees of freedom, df. The table below lists the probability distributions that we will use in this course, their **R** probability functions, and the **R** function arguments used to specify the parameters that define each distribution:

Distribution	name	R Functions	R Function Arguments
Normal	`*norm`	`dnorm` `pnorm` `qnorm` `rnorm`	`mean = , sd =`
t	`*t`	`dt` `pt` `qt` `rt`	`df =`

Distribution	name	R Functions	R Function Arguments
Chi-square	`*chisq`	`dchisq` `pchisq` `qchisq` `rchisq`	`df =`
F	`*f`	`df` `pf` `qf` `rf`	`df1 =`, `df2 =`

where `*` equals `d`, `p`, `q`, or `r`.

The Normal Distribution

The **Normal distribution** is a continuous probability distribution. The **parameters** that define the Normal distribution are its mean μ and variance σ^2, giving $X \sim N(\mu, \sigma^2)$. When $\mu = 0$ and $\sigma^2 = \sigma = 1$, the Normal distribution is called a **standard Normal**. The letter Z is traditionally used when referring to a standard Normal random variable (i.e., $Z \sim N(0, 1)$). **Note**: Rather than input the variance, σ^2, we must input the **standard deviation** σ when using `dnorm()`, `pnorm()`, `qnorm()`, and `rnorm()`. If the mean and standard deviation are not specified, **R** will assume `mean=0` and `sd=1` (i.e., a **standard Normal** distribution, $N(0, 1)$).

Example: Suppose the distribution of diastolic blood pressure in the population follows a Normal distribution with $\mu = 80$ mm Hg and $\sigma = 12$ mm Hg.

pnorm()

The **cumulative distribution function** (cdf) is defined as $P(X \leq x)$. In continuous distributions, $P(X \leq x) = P(X < x)$. There is not a closed-form algebraic expression to compute the area under the Normal distribution. Numerical methods must be used to calculate the areas (hence the use of "normal tables" such as the one in the back of our textbook). We commonly use the `p` functions (i.e., `pnorm()`, here) to compute **p-values** when performing our hypothesis tests.

The `pnorm(q=, mean=mu, sd=sigma)` function is used to compute the Normal cdf at a given value `q=` x, returning $P(X \leq x)$. For example, to find the probability that an individual from our population has a diastolic blood pressure less than 68, $P(X < 68)$:

```
mu <- 80
sigma <- 12    # standard deviation

# P(X < 68) where X~N(80, 12)
pnorm(68, mean = mu, sd = sigma)
```

```
## [1] 0.1586553
```

To find $P(X > x)$, use the fact that the total area under the Normal density equals 1, giving $P(X > x) = 1 - P(X \leq x)$, or `1-pnorm(q=x, mean=mu, sd=sigma)` for a given mean and standard deviation. We can apply the `lower.tail=FALSE` option in `pnorm()` to directly return the upper tail probability, $P(X > x)$. The probability that an individual from our population has a diastolic blood pressure greater than 68, $P(X > 68)$:

```
1 - pnorm(68, mean = mu, sd = sigma)                          # P(X > 68) = 1 - P(X <= 68)
```

```
## [1] 0.8413447
```

```
# Equivalently,
pnorm(68, mean = mu, sd = sigma, lower.tail = FALSE)  # P(X > 68) using lower.tail=FALSE option
```

```
## [1] 0.8413447
```

To find $P(a < X < b)$, the probability X falls in the interval (a, b), compute $P(a < X < b) = P(X < b) - P(X < a)$ using `pnorm(q=b, mean=mu, sd=sigma) - pnorm(q=a, mean=mu, sd=sigma)`. The probability that an individual from our population has a diastolic blood pressure between 68 and 92, $P(68 < X < 92)$:

```
pnorm(92, mean = mu, sd = sigma) - pnorm(68, mean = mu, sd = sigma)  # P(68 < X < 92)
```

```
## [1] 0.6826895
```

The methods presented in the table below can be used to find probabilities associated with any **continuous probability distribution**:

Probability of Interest	R Function
$P(X \leq x) = P(X < x)$	`pnorm(q=x, mean=mu, sd=sigma)`

$P(X \geq x) = P(X > x) = 1 - P(X \leq x)$	`1-pnorm(q=x, mean=mu, sd=sigma)`
	`pnorm(q=x, mean=mu, sd=sigma, lower.tail=FALSE)`

$P(a < X < b) = P(X < b) - P(X < a)$	`pnorm(q=b, mean=mu, sd=sigma) - pnorm(q=a, mean=mu, sd=sigma)`

We can convert a $N(\mu, \sigma)$ random variable to a standard Normal $N(0, 1)$ random variable by subtracting its mean μ and dividing by its standard deviation σ. That is, if $X \sim N(\mu, \sigma)$, then $Z = \dfrac{X - \mu}{\sigma} \sim N(0, 1)$. Again, if `mean=` and `sd=` are omitted in any of the Normal distribution functions, then **R** assumes the **standard Normal distribution**. For example $P(Z \leq 1.96)$ = `pnorm(1.96)`.

Exercise: Suppose we are performing a Z-test (i.e., $Z \sim N(0, 1)$). (i) Suppose we are conducting a 2-sided test and observe the test statistic $z = 2.01$. Find the p-value of this test; (ii) Suppose we are instead performing a 1-sided upper tailed test and observe the test statistic $z = 2.01$. Find the p-value in this case.

▶ Answer:

qnorm()

Use `qnorm(p=pr, mean=mu, sd=sigma)` to find the inverse cdf, or the quantile q_{pr} corresponding to a given probability $P(X \leq q_{pr}) = pr$. For example, the value of diastolic blood pressure at the 80th percentile is equal to:

```
qnorm(0.80, mean = mu, sd = sigma)
```

```
## [1] 90.09945
```

The `qnorm()` function is used to find the hypothesis test critical value that maintains a desired type I error rate, α.

Exercise: (i) Find the critical value of a two-sided Z-test at the α = 0.05-level; (ii) Find the critical value of a one-sided upper tailed Z-test at the α = 0.05-level; (iii) Find the critical value of a one-sided lower tailed Z-test at the α = 0.05-level.

▶ Answer:

Means

Quantitative variables are summarized using means. We would like to draw conclusions about the population mean μ.

Confidence Interval and Hypothesis Test

In a **one-sample setting**, the goal is to estimate and test a specific hypothesis about the mean of a quantitative variable μ in a population of interest. A **confidence interval** is an interval that is likely to contain the true population mean μ with high probability, $1 - \alpha$. The confidence level is traditionally set to 95%.

The **one-sample t-test** compares the mean μ in the current population to a hypothesized value μ_0. Under the null hypothesis, the mean in the current population is equal to the hypothesized mean, $H_0 : \mu = \mu_0$. Depending on the research question, either a one-sided or two-sided alternative hypothesis H_1 is used. One-sided tests are either upper-tailed ($\mu > \mu_0$) or lower-tailed ($\mu < \mu_0$).

One-Sided (Upper-Tailed)	One-Sided (Lower-Tailed)	Two-Sided or Two-Tailed
$H_0 : \mu = \mu_0$	$H_0 : \mu = \mu_0$	$H_0 : \mu = \mu_0$
$H_1 : \mu > \mu_0$	$H_1 : \mu < \mu_0$	$H_1 : \mu \neq \mu_0$

The one-sample **t-test statistic** is compared to a t-distribution with $n - 1$ degrees of freedom.

In a **two-sample setting**, the goal is to compare the mean of a quantitative variable in two populations, μ_1 and μ_2. Rather than look at each mean separately, our inferences involve the difference in the population means $\mu_1 - \mu_2$. Thus, in the two-sample setting, the **confidence interval** is reported for $\mu_1 - \mu_2$.

The **two-sample t-test** compares μ_1 and μ_2. Under the null hypothesis, there is no difference in the two population means, $H_0 : \mu_1 = \mu_2$, or, equivalently, their difference is equal zero $H_0 : \mu_1 - \mu_2 = 0$. Depending on the research question, either a one-sided or two-sided alternative hypothesis H_1 is used. One-sided tests are either upper-tailed ($\mu_1 > \mu_2$ or $\mu_1 - \mu_2 > 0$) or lower-tailed ($\mu_1 < \mu_2$ or $\mu_1 - \mu_2 < 0$).

One-Sided (Upper-Tailed)	One-Sided (Lower-Tailed)	Two-Sided or Two-Tailed
$H_0 : \mu_1 - \mu_2 = 0$	$H_0 : \mu_1 - \mu_2 = 0$	$H_0 : \mu_1 - \mu_2 = 0$
$H_1 : \mu_1 - \mu_2 > 0$	$H_1 : \mu_1 - \mu_2 < 0$	$H_1 : \mu_1 - \mu_2 \neq 0$

The two-sample **t-test statistic** is compared to a t-distribution. Degrees of freedom $n_1 + n_2 - 2$ are used when conducting the pooled test and Welch-Satterthwaite degrees of freedom are used in the unpooled test.

The `t.test()` function in **R** can be used to carry out a one-sample and two-sample t-test. The output will also report a confidence interval for μ (one-sample case) and $\mu_1 - \mu_2$ (two-sample case). By default, a two-sided (`alternative="two.sided"`) test is performed and a 95% confidence interval (`conf.level=0.95`) is produced. In the two-sample case, Welch's unpooled (`var.equal=FALSE`) t-test is reported testing $H_0 : \mu_1 - \mu_2 = 0$ (`mu=0`). Arguments of the `t.test()` function are listed in the table below:

`t.test()` **Function Arguments**	**Option Definition**
`x=`	(one-sample) Quantitative variable analyzed
`formula=`	`analysis_variable ~ group_variable` factor variable
`alternative=`	`"two.sided"` (default), `"less"` , `"greater"`
`mu=`	Hypothesized value of μ (i.e., μ_0) (one-sample) or $\mu_1 - \mu_2$ (two-sample) under H_0 (default `=0`)
`paired=`	(one-sample) Indicator for paired t-test (`=TRUE`)
`var.equal=`	(two-sample) Logical variable indicating if pooled t-test (`=TRUE`) or unpooled t-test (default `=FALSE`) is performed same variance different variances
`conf.level=`	Confidence level, C (default `=0.95`)

One-Sample CI for μ

We would like to construct a CI for the mean systolic blood pressure the Framingham population, μ. The CI for μ is centered on the point estimate $\bar{x}$. Average systolic blood pressure `SYSBP` in our Framingham sample is equal to 134.46.

```
mean(fhs100$SYSBP, na.rm = TRUE)
```

```
## [1] 134.465
```

Since the `t.test()` function reports both the test results and the confidence interval, we can pick out the confidence interval portion of the output by appending `$conf.int[1:2]` to the `t.test()` function. The 95% CI for mean systolic blood pressure is reported below:

```
t.test(fhs100$SYSBP)$conf.int[1:2]
```

t test object, have many attributes, access attribute by $

```
## [1] 129.982 138.948
```

We can pick out the individual lower and upper bound by requesting the index `[1]` or `[2]` alone in the bracket, where `[1]` returns the lower limit and `[2]` returns the upper limit.

```
t.test(fhs100$SYSBP)$conf.int[1]   # lower limit of 95% CI
```

```
## [1] 129.982
```

```
t.test(fhs100$SYSBP)$conf.int[2]   # upper limit of 95% CI
```

```
## [1] 138.948
```

This is useful when writing inline **R** code (e.g., The 95% confidence interval for mean systolic BP in the Framingham population is (129.98, 138.95).)

We can report the confidence interval in a subgroup of interest (e.g., overweight/obese vs. non-overweight/obese) by creating a new data frame (e.g., `fhs100_ov`) that is a subset of the original data frame and applying the `t.test()` function to the `SYSBP` variable in the subset (`fhs100_ov$SYSBP`). Equivalently, we can directly subset the `SYSBP` variable using the bracket operator, `fhs100$SYSBP[fhs100$OVERWEIGHTOBESE_factor == "Yes"]` in the `t.test()` function:

```
table(fhs100$OVERWEIGHTOBESE_factor, useNA = "ifany")
```

```
## 
## Yes  No 
##  56  44
```

```
# Creating the subset of overweight/obese
fhs100_ov <- subset(fhs100, OVERWEIGHTOBESE_factor == "Yes")

# and requesting the confidence interval for SYSBP in overweight/obese
t.test(fhs100_ov$SYSBP)$conf.int[1:2]
```

```
## [1] 134.1630 146.5156
```

```
# Equivalently,
t.test(fhs100$SYSBP[fhs100$OVERWEIGHTOBESE_factor == "Yes"])$conf.int[1:2]
```

```
## [1] 134.1630 146.5156
```

Exercise: Report the 95% confidence interval for mean `SYSBP` in those who are not overweight or obese. Does one group tend to have higher systolic blood pressure values, on average? Do the confidence intervals in overweight/obese and non-overweight/obese overlap?

▶ Answer:

One-Sample t-Test for μ

When performing a one-sample **hypothesis test** for μ, we must specify the value of μ that is hypothesized under the null hypothesis, $H_0 : \mu = \mu_0$. Remember that in a one-sample problem, hypotheses are specified about a single distribution or a single population. We would like to test the hypothesis that average systolic blood pressure level in our population of Framingham residents is significantly different from a reference value of 120. That is, $H_0 : \mu = 120$ vs. $H_1 : \mu \neq 120$. This is a two-sided alternative hypothesis. We will use our sample data `fhs100` to answer this question.

To test $H_0 : \mu = 120$ vs. $H_1 : \mu \neq 120$, specify `mu = 120`. Note that `alternative = "two.sided"` is the default option, so it does not need to be specified. Very small p-values are often displayed in scientific notation. To "turn off" scientific notation, `options(scipen=999)` is applied in the first code chunk of this Markdown file.

```
t.test(fhs100$SYSBP,
       mu = 120,                    # mu0
       alternative = "two.sided")  # direction of H1 (can omit if 2-sided)
```

```
##
##  One Sample t-test
##
## data:  fhs100$SYSBP
## t = 6.4023, df = 99, p-value = 0.000000005153
## alternative hypothesis: true mean is not equal to 120
## 95 percent confidence interval:
##  129.982 138.948
## sample estimates:
## mean of x
##   134.465
```

```
# Used to pick out pieces of the results in the inline R code below
tt <- t.test(fhs100$SYSBP, mu = 120)   # default is two.sided test
```

There is strong evidence to **reject** H_0 in favor of H_1 and conclude that the average systolic blood pressure in our FHS population is significantly different from 120 at the α = 0.05-level. The point estimate mean systolic blood pressure, $\bar{x} =$ 134.46, suggests that average systolic blood pressure in the FHS population is greater than 120. The confidence interval for μ, as expected, does not include the hypothesized value of μ under the null hypothesis, $\mu_0 = 120$.

To run a **one-sided, upper-tailed** test (i.e., $H_1 : \mu > 120$), use the option, `alternative = "greater"`. Because the point estimate $\bar{x}$ is in the direction of an upper-tailed alternative, the upper-tailed test also supports rejection of H_0 and a conclusion that $\mu > 120$:

```
t.test(fhs100$SYSBP,
       mu = 120,                    # mu0
       alternative = "greater")    # direction of H1
```

```
##
##  One Sample t-test
##
## data:  fhs100$SYSBP
## t = 6.4023, df = 99, p-value = 0.000000002577
## alternative hypothesis: true mean is greater than 120
## 95 percent confidence interval:
##  130.7136      Inf
## sample estimates:
## mean of x
##   134.465
```

To run a **one-sided, lower-tailed** test (i.e., $H_1 : \mu < 120$), use the option, `alternative = "less"` . The sample does not provide evidence to conclude that $\mu < 120$:

```
t.test(fhs100$SYSBP,
       mu = 120,                   # mu0
       alternative = "less")       # direction of H1
```

```
##
##  One Sample t-test
##
## data:  fhs100$SYSBP
## t = 6.4023, df = 99, p-value = 1
## alternative hypothesis: true mean is less than 120
## 95 percent confidence interval:
##      -Inf 138.2164
## sample estimates:
## mean of x
##   134.465
```

Two-Sample CI for $\mu_1 - \mu_2$

We would like to construct a CI for the difference in mean systolic blood pressure in non-overweight/obese and overweight/obese individuals, $\mu_1 - \mu_2$. The confidence interval is centered at the difference in sample means, $\bar{x}_1 - \bar{x}_2$. The individual group means $\bar{x}_1$, $\bar{x}_2$ and the difference in group means $\bar{x}_1 - \bar{x}_2$ are computed below:

```
xbar1_y <- mean(fhs100$SYSBP[fhs100$OVERWEIGHTOBESE_factor == "Yes"], na.rm=TRUE)  # xbar1
xbar2_n <- mean(fhs100$SYSBP[fhs100$OVERWEIGHTOBESE_factor == "No"], na.rm=TRUE)   # xbar2
diffxbar <- xbar1_y - xbar2_n                                                      # difference in
 sample means

# Printing results xbar1, xbar2, xbar1 - xbar2      show results pretty
c(xbar1_obese = xbar1_y, xbar2_notobese = xbar2_n, diff = diffxbar)
```

```
##    xbar1_obese xbar2_notobese           diff
##      140.33929      126.98864       13.35065
```

We use model formula notation in the `t.test()` function to identify our analysis variable and our group identifier (e.g., `formula=analysis_variable ~ explanatory/grouping_variable`).

```
t.test(SYSBP ~ OVERWEIGHTOBESE_factor, data = fhs100)
```

```
##
##  Welch Two Sample t-test
##
## data:  SYSBP by OVERWEIGHTOBESE_factor
## t = 3.1103, df = 97.175, p-value = 0.002453
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.831572 21.869726
## sample estimates:
## mean in group Yes  mean in group No
##          140.3393          126.9886
```

The 95% confidence interval for the difference in mean systolic BP in overweight/obese vs. non-overweight/obese individuals in the Framingham population is (4.83, 21.87).

Note: The CI for the **difference in means** above is computed for overweight/obese individuals — non-overweight/obese because of the way we specified the factor levels of `OVERWEIGHTOBESE_factor` (i.e., as `c(1, 0)` or “Yes” then “No”). To change the order of the computed difference, re-order the levels of your factor variable using the `relevel()` function to change the reference category. For example, in order for the difference to be computed as `No` (not overweight/obese) — `Yes` (overweight/obese), specify the reference category `ref = "No"` in the `relevel()` function. The reference category is the group with subscript #1 (i.e., μ_1). Notice how the confidence interval below is now flipped and opposite in sign:

```
# Re-ordering OVERWEIGHTOBESE_factor using the relevel() function
t.test(SYSBP ~ relevel(OVERWEIGHTOBESE_factor, ref = "No"), data = fhs100)
```

```
##
##  Welch Two Sample t-test
##
## data:  SYSBP by relevel(OVERWEIGHTOBESE_factor, ref = "No")
## t = -3.1103, df = 97.175, p-value = 0.002453
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -21.869726  -4.831572
## sample estimates:
##  mean in group No mean in group Yes
##          126.9886          140.3393
```

Two-Sample t-Test for μ_1 vs. μ_2

We would like to determine if mean systolic blood pressure is significantly different in the population of overweight/obese individuals and non-overweight/obese individuals. By default, a two-sided (`alternative="two.sided"`) unpooled (`var.equal=FALSE`) t-test is reported, but the pooled t-test can be requested using `var.equal=TRUE` . The results of both tests are provided below. An upper-tailed test is requested using `alternative="greater"` ; a lower-tailed test is requested using `alternative="less"` .

```
# Pooled t-test
t.test(SYSBP ~ OVERWEIGHTOBESE_factor, data = fhs100,
       var.equal = TRUE,
       alternative = "two.sided")  # default
```

```
##
##  Two Sample t-test
##
## data:  SYSBP by OVERWEIGHTOBESE_factor
## t = 3.0541, df = 98, p-value = 0.002907
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.675694 22.025605
## sample estimates:
## mean in group Yes  mean in group No
##          140.3393          126.9886
```

```
# Unpooled t-test
t.test(SYSBP ~ OVERWEIGHTOBESE_factor, data = fhs100,
       var.equal = FALSE,          # default
       alternative = "two.sided")  # default
```

```
##
##  Welch Two Sample t-test
##
## data:  SYSBP by OVERWEIGHTOBESE_factor
## t = 3.1103, df = 97.175, p-value = 0.002453
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.831572 21.869726
## sample estimates:
## mean in group Yes  mean in group No
##          140.3393          126.9886
```

The sample provides evidence to reject $H_0 : \mu_1 = \mu_2$ and conclude $H_1 : \mu_1 \neq \mu_2$. That is, there is evidence to conclude that mean systolic blood pressure is significantly different in those who are overweight/obese and those who are not overweight/obese (p-value from unpooled test = 0.002).

```
# 95% unpooled CI for mu1-mu2
ci95 <- t.test(SYSBP ~ OVERWEIGHTOBESE_factor, data = fhs100,
               var.equal = FALSE, alternative = "two.sided")$conf.int[1:2]
ci95
```

```
## [1]  4.831572 21.869726
```

The 95% unpooled confidence interval for the mean difference in systolic blood pressure in the Framingham population of overweight/obese individuals vs. non-overweight/obese individuals is (4.83, 21.87). Notice that this confidence interval does not contain 0, the value hypothesized for the difference in means under H_0 (i.e., $H_0 : \mu_1 - \mu_2 = 0$).

Exercise: Is there a significant difference in mean `SYSBP` in those who are overweight/obese vs. not overweight/obese in the population of **females** only? Assess using a two-sided unpooled t-test. *Hint:* Use the code above for the unpooled t-test comparing overweight/obese vs. non-overweight/obese but specify the `subset()` of `fhs100` that includes only `SEX_factor == "Female"` in `data=` .

▶ Answer:

Proportions

Categorical variables are summarized using **frequency tables** and **cross-tabulations** that summarize counts and sample proportions. We would like to draw conclusions about the population proportion p.

Confidence Interval and Hypothesis Test

In a **one-sample setting**, the goal is to estimate and test a specific hypothesis about the proportion of successes, p, in a population of interest. A **confidence interval** is an interval that is likely to contain the true population proportion p with high probability, $1 - \alpha$. The confidence level is traditionally set to 95%.

The **one-sample test for proportions** compares the proportion of successes p in the current population to a hypothesized value p_0. Under the null hypothesis, the proportion of successes in the population is equal to a hypothesized proportion, $H_0 : p = p_0$. Depending on the research question, either a one-sided or two-sided alternative hypothesis H_1 is used. One-sided tests are either upper-tailed ($p > p_0$) or lower-tailed ($p < p_0$).

One-Sided (Upper-Tailed)	One-Sided (Lower-Tailed)	Two-Sided or Two-Tailed
$H_0 : p = p_0$	$H_0 : p = p_0$	$H_0 : p = p_0$
$H_1 : p > p_0$	$H_1 : p < p_0$	$H_1 : p \neq p_0$

In a **two-sample setting**, the goal is to compare the proportion of successes in the two exposure groups p_1 and p_2. Confidence intervals are constructed for the difference in proportions $p_1 - p_2$. Under the null hypothesis, there is no difference in the two population proportions, $H_0 : p_1 = p_2$, or, equivalently, their difference is equal to zero $H_0 : p_1 - p_2 = 0$. Depending on the research question, either a one-sided or two-sided alternative hypothesis H_1 is used. One-sided tests are either upper-tailed ($p_1 > p_2$ or $p_1 - p_2 > 0$) or lower-tailed ($p_1 < p_2$ or $p_1 - p_2 < 0$).

One-Sided (Upper-Tailed)	One-Sided (Lower-Tailed)	Two-Sided or Two-Tailed
$H_0 : p_1 - p_2 = 0$	$H_0 : p_1 - p_2 = 0$	$H_0 : p_1 - p_2 = 0$
$H_1 : p_1 - p_2 > 0$	$H_1 : p_1 - p_2 < 0$	$H_1 : p_1 - p_2 \neq 0$

The `prop.test()` function in **R** can be used to carry out a one-sample and two-sample **test for binomial proportions** based on the Normal approximation to the Binomial distribution, although a Chi-square statistic is reported. The output also includes a confidence interval for p (one-sample case) or $p_1 - p_2$ (two-sample case). By default, a two-sided (`alternative="two.sided"`) test is performed and a 95% confidence interval (`conf.level=0.95`) is produced. A continuity correction is also applied by default, which helps improve the accuracy of the Normal approximation. To request this “uncorrected” test, use the `correct=FALSE` option in the `prop.test()` function. Arguments of the `prop.test()` function are listed in the table below:

`prop.test()` Function Arguments	Option Definition
x=	Count of successes or table containing number of successes as first entry (one-sample)/first column (two-sample)

`prop.test()` Function Arguments	Option Definition
`n=`	Number of trials (ignored if `x=` is a table)
`alternative=`	`"two.sided"` (default), `"less"` , `"greater"`
`p=`	Hypothesized proportion p_0 under H_0 (one-sample)/difference in proportions $p_1 - p_2$ under H_0 (default `=0`)
`correct=`	Continuity correction (default `=TRUE`)
`conf.level=`	Confidence level, C (default `=0.95`)

One-Sample CI for p

We would like to construct a CI for the proportion of individuals who develop CVD in the Framingham population, also known as the risk of developing CVD in this population. In our sample of 100, cardiovascular disease developed in 27 individuals during follow-up.

```
# One-way frequency table
tab1p <- table(fhs100$CVD_factor, dnn = "Cardiovascular Disease")
tab1p
```

```
## Cardiovascular Disease
## Yes  No
##  27  73
```

27% of individuals develop CVD during follow-up.

```
prop.table(tab1p)
```

```
## Cardiovascular Disease
##  Yes   No
## 0.27 0.73
```

Since the `prop.test()` function reports both the test results and the confidence interval, we can pick out the confidence interval portion of the output by appending `$conf.int[1:2]` to the `prop.test()` function. The 95% CI for the proportion of individuals who develop CVD in the Framingham population is reported below:

```
prop.test(tab1p)$conf.int[1:2]
```

```
## [1] 0.1883648 0.3696071
```

The sample proportion of CVD cases equals 0.27 and the 95% confidence interval for the proportion of individuals with CVD is (0.19, 0.37).

One-Sample Test for p

We would like to determine if the risk of CVD in our population of Framingham individuals is different from the present-day prevalence of CVD of 10.6%[1]. That is, $H_0 : p = 0.106$ vs. $H_1 : p \neq 0.106$. We observed `x=` 27 successes (i.e., CVD cases) in our sample of `n=` 100 trials (i.e., individuals).

The `prop.test()` function in **R** can be used to carry out a one-sample test of proportions. **R** will model the probability of the *first* level of the binary variable displayed in the input table (`CVD_factor` = Yes, here). Therefore, `p=` specified under H_0 should correspond to the hypothesized probability of this outcome.

```
# One-sample test of binomial proportions (inputting table of sample data)
prop.test(tab1p,                        # input one-way frequency table
          p = 0.106,                  # p0
          alternative = "two.sided")  # direction of H1 (can omit if 2-sided)
```

```
## 
##  1-sample proportions test with continuity correction
## 
## data:  tab1p, null probability 0.106
## X-squared = 26.678, df = 1, p-value = 0.0000002404
## alternative hypothesis: true p is not equal to 0.106
## 95 percent confidence interval:
##  0.1883648 0.3696071
## sample estimates:
##    p 
## 0.27
```

The data provide evidence against H_0. **R** reports a chi-square test statistic, which is the square of the Z-statistic that we would have manually calculated from the one-sample Z-test of binomial proportions. Recall that a standard Normal random variable squared follows a chi-square distribution with 1 degree of freedom. The proportion individuals with CVD in our Framingham population is significantly different from 0.106 (p-value <.0001).

The `binom.test()` function in **R** can be used to carry out the **exact binomial test** and should be used when our Normal approximation assumptions are not met. The syntax of `binom.test()` is the same as `prop.test()`, except there is no continuity correction option. An exact confidence interval for p is also produced. As with most exact tests, an exact p-value is computed. Here, the p-value is computed directly using the binomial distribution. Again, we have evidence to reject H_0 that $p = 0.106$ (p-value <.0001).

```
# One-sample exact binomial test
binom.test(tab1p,
           p = 0.106,
           alternative = "two.sided")
```

```
## 
##  Exact binomial test
## 
## data:  tab1p
## number of successes = 27, number of trials = 100, p-value = 0.000003719
## alternative hypothesis: true probability of success is not equal to 0.106
## 95 percent confidence interval:
##  0.1860664 0.3680163
## sample estimates:
## probability of success 
##                   0.27
```

Two-Sample CI for $p_1 - p_2$

交叉表

Cross-tabulations are used to describe the relationship between two categorical variables. We would like to determine if those who are overweight/obese have a higher risk of CVD. That is, $H_0 : p_1 - p_2 = 0$ vs. $H_0 : p_1 - p_2 > 0$, where p_1 is the proportion of individuals with CVD in the overweight/obese population and p_2 is the proportion of individuals with CVD in the non-overweight/obese population. We observed `x=` 21 CVD cases out of `n=` 56 in the exposed group (i.e., overweight/obese) and we observed `x=` 6 CVD cases out of `n=` 44 in the unexposed group (i.e., non-overweight/obese).

```
# 2-way frequency table of exposure*disease
tab2p <- table(fhs100$OVERWEIGHTOBESE_factor, fhs100$CVD_factor,
               dnn = c("Overweight/Obese", "Cardiovascular Disease"))
tab2p
```

```
##                Cardiovascular Disease
## Overweight/Obese Yes No
##              Yes  21 35
##              No    6 38
```

```
prop.table(tab2p, margin = 1)          # row proportions (^P(D|E) and ^P(D|E'))
```

```
##                Cardiovascular Disease
## Overweight/Obese       Yes        No
##              Yes 0.3750000 0.6250000
##              No  0.1363636 0.8636364
```

When inputting a two-way contingency table into the `prop.test()` function, R will assume the first column of the table is the event (CVD = yes, here) and compare risks over the two rows (exposure categories). The confidence interval is centered at the difference in sample proportions $\hat{p}_1 - \hat{p}_2$. The individual group proportions $\hat{p}_1, \hat{p}_2$ and the difference in group means $\hat{p}_1 - \hat{p}_2$ are computed below:

```
phats <- prop.table(tab2p, margin = 1)  # row proportions
phats
```

```
##                Cardiovascular Disease
## Overweight/Obese       Yes        No
##              Yes 0.3750000 0.6250000
##              No  0.1363636 0.8636364
```

```
phat_y <- phats[1,1]          # phat1 ^P(D|E)
phat_n <- phats[2,1]          # phat2 ^P(D|E')
diffphat <- phat_y - phat_n   # difference in sample proportions

# Printing results phat1, phat2, phat1 - phat2
c(phat1_obese = phat_y, phat2_notobese = phat_n, diff = diffphat)
```

```
##    phat1_obese phat2_notobese           diff
##      0.3750000      0.1363636      0.2386364
```

The 95% confidence interval for $p_1 - p_2$ is reported below:

```
prop.test(tab2p)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  tab2p
## X-squared = 5.9599, df = 1, p-value = 0.01464
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.0559883 0.4212844
## sample estimates:
##    prop 1    prop 2
## 0.3750000 0.1363636
```

The proportion of the exposed group who developed CVD is equal to 0.375; the proportion of the unexposed group who developed CVD is equal to 0.136. The 95% confidence interval for the difference in the risk of CVD in overweight/obese vs. non-overweight/obese individuals in the Framingham population is (0.056, 0.421).

Two-Sample Test for p_1 vs. p_2

We would like to test the null hypothesis $H_0 : p_1 - p_2 = 0$, or a null hypothesis of no association between the two variables (e.g., exposure or being overweight/obese and disease or CVD).

```
prop.test(tab2p,
          alternative = "two.sided")   # default
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  tab2p
## X-squared = 5.9599, df = 1, p-value = 0.01464
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.0559883 0.4212844
## sample estimates:
##    prop 1    prop 2
## 0.3750000 0.1363636
```

The data provide evidence against H_0. **R** again reports a chi-square test statistic. The proportion overweight/obese (exposed) individuals with CVD in our Framingham population is significantly different from the proportion of non-overweight/obese (unexposed) individuals with CVD (p-value = 0.015).

Chi-Square Test Independence

Two random variables are **independent** if the probability distribution of one variable is not affected by the presence of the other variable. For example, if exposure and disease are independent, then exposure status does not affect the risk of disease. The **chi-square test of independence** is used to determine if there is an association between two categorical variables. Since we often summarize the association between two categorical variables using a contingency table, the chi-square test can be thought of as a test of the association between the rows and columns of a contingency table. Under H_0, there is no association between the row and column variable and under H_1, there is an association between the row and column variable.

The `chisq.test()` function in **R** can be used to carry out the chi-square test of independence. A continuity correction is also applied by default. To request the test without the continuity correction, use the `correct=FALSE` option in the `chisq.test()` function.

`chisq.test()` Function Arguments	**Option Definition**
`x=`	Test of independence: contingency table created using the `table()` function
`correct=`	Continuity correction (default `=TRUE`)

Repeating our test above, we would like to determine if there is a significant association between overweight/obesity and development of cardiovascular disease. We use the 2x2 contingency table object created earlier `tab2p` in the `chisq.test()` function:

```
# Chi-square test of independence between overweight/obesity and CVD
chisq.test(tab2p)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab2p
## X-squared = 5.9599, df = 1, p-value = 0.01464
```

There is a significant association between exposure (overweight/obesity) and disease (CVD) (p-value = 0.015). The observed test statistic, 5.96, is compared to a chi-square distribution with 1 degree of freedom. This is the same test statistic computed earlier using the `prop.test()` function for the two-sample case.

Exercise: Is there a significant association between sex (`SEX_factor`) and CVD? That is, is the proportion of males who develop CVD significantly different from the proportion of females who develop CVD in our Framingham population?

▶ Answer:

1. https://www.ahajournals.org/doi/epub/10.1161/CIR.0000000000000757 (https://www.ahajournals.org/doi/epub/10.1161/CIR.0000000000000757)↩