Exam Practice Solutions

Math 141

Fall 2020

```
df <- read_csv("~/Desktop/exam_prep_inference_cep.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   Type = col_character(),
##   Xray = col_character(),
##   Region = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

Problem 1

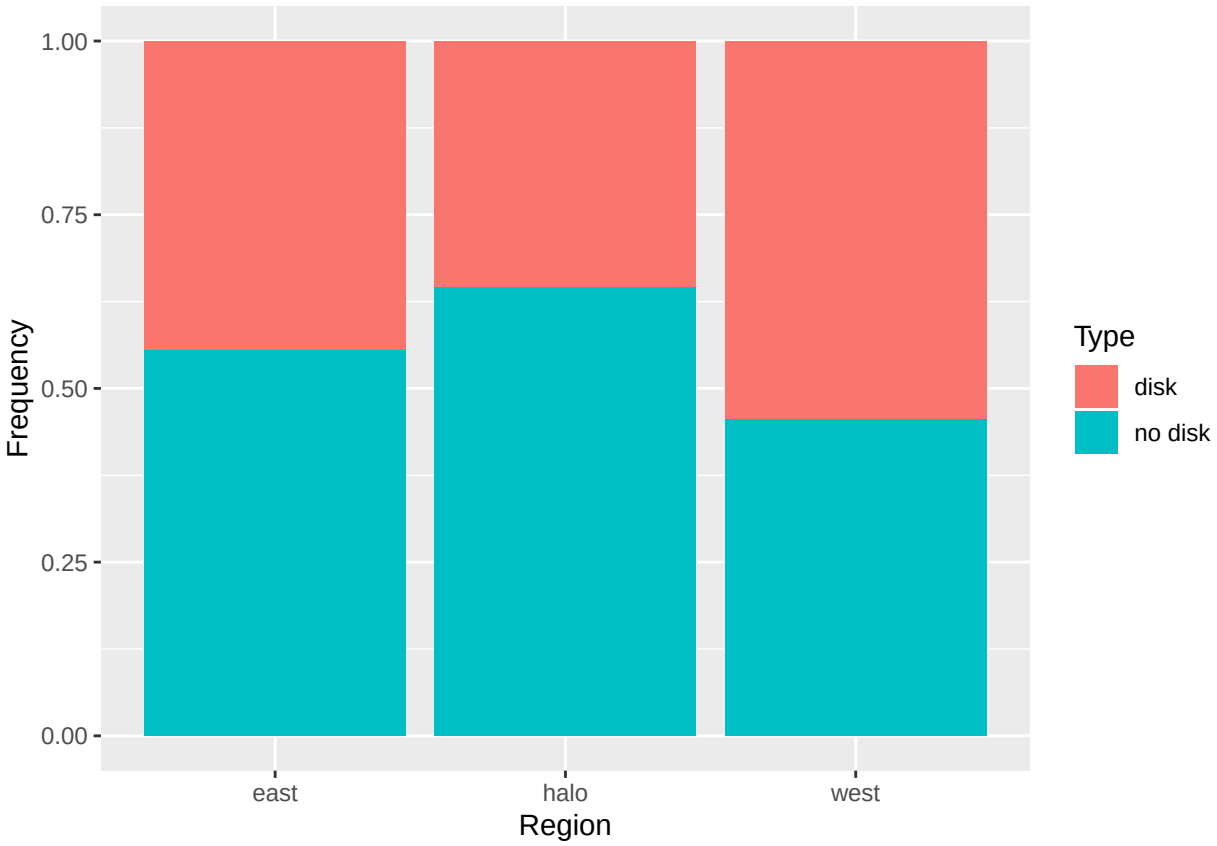
Exploratory Data Analysis

The variables we are interested in are categorical, so we will use a table of frequencies and a bar chart to eplore the relationship between `Type` and `Region`.

```
df_counts <- df %>%
  filter(Type %in% c("disk","no disk")) %>%
  group_by(Region,Type) %>%
  summarize(counts=n()) %>%
  mutate(freq=counts/sum(counts))

df_counts
```

```
## # A tibble: 6 x 4
## # Groups:   Region [3]
##   Region Type    counts  freq
##   <chr>  <chr>    <int> <dbl>
## 1 east   disk       192 0.444
## 2 east   no disk    240 0.556
## 3 halo   disk       507 0.353
## 4 halo   no disk    929 0.647
## 5 west   disk       159 0.543
## 6 west   no disk    134 0.457
```

```
df_counts %>%
  ggplot(aes(x=Region,y=counts,fill=Type)) +
    geom_col(position="fill") +
    scale_y_continuous(name = "Frequency")
```



Two Categories

Confidence intervals

Approximation with probability models

The confidence interval around a single proportion point estimate is

$$\hat{p} \pm MOE$$

$$\hat{p} \pm z^* \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

For the "east" region, since there are 192 stars with a disk and 240 without, $\hat{p} = \frac{192}{192+240} = 0.44$:

$$\hat{p_{east}} \pm z^* \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

$$0.44 \pm 1.96 \times \sqrt{\frac{0.44(1-0.44)}{432}}$$

$$0.44 \pm 0.05$$

```
1.96 * sqrt( (0.44 * (1 - 0.44)) / 432)
```

```
## [1] 0.04680956
```

And for the "west" region, since there are 159 stars with a disk and 134 without, $\hat{p} = \frac{159}{159+134} = 0.54$:

$$\hat{p_{west}} \pm z^* \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

$$0.54 \pm 1.96 \times \sqrt{\frac{0.54(1-0.54)}{293}}$$

$$0.54 \pm 0.06$$

```
1.96 * sqrt( (0.54 * (1 - 0.54)) / 293)
```

```
## [1] 0.05706871
```

Therefore, we think that a plausable range of values that contain the population proportion in the east region is, p_{east}, is $0.39 - 0.49$, and for the west, p_{west}, is $0.48 - 0.6$.

Computational method with Infer

We will use `infer` to construct bootstrap estimations of the sampling distribution.

```
df_two <- df %>%
  filter(Region %in% c("east","west"),
         Type %in% c("disk","no disk")) %>%
  mutate(Region=factor(Region),
         Type=factor(Type))

df_two %>%
  group_by(Region, Type) %>%
  summarise(count = n()) %>%
  mutate(Type = fct_drop(Type))
```

```
## # A tibble: 4 x 3
## # Groups:   Region [2]
##   Region Type    count
##   <fct>  <fct>   <int>
## 1 east   disk      192
## 2 east   no disk   240
## 3 west   disk      159
## 4 west   no disk   134
```

```
# east
prop_east <- df_two %>%
  filter(Region == "east") %>%
  specify(response = Type, success = "disk") %>%
  calculate(stat = "prop")

boot_east <- df_two %>%
  filter(Region == "east") %>%
  specify(response = Type, success = "disk") %>%
  generate(reps=1000,type="bootstrap") %>%
  calculate(stat = "prop")

ci_east <- boot_east %>%
  get_confidence_interval(type = "se", level = 0.95, point_estimate = prop_east)

prop_east
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 0.444
```

```
ci_east
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.399    0.490
```

```
# west
prop_west <- df_two %>%
  filter(Region == "west") %>%
  specify(response = Type, success = "disk") %>%
  calculate(stat = "prop")

boot_west <- df_two %>%
  filter(Region == "west") %>%
  specify(response = Type, success = "disk") %>%
  generate(reps=1000,type="bootstrap") %>%
  calculate(stat = "prop")

ci_west <- boot_west %>%
  get_confidence_interval(type = "se", level = 0.95, point_estimate = prop_west)

prop_west
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 0.543
```

```
ci_west
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.486    0.599
```

Hypothesis testing

Now to answer the question, "Do the proportions of stars with disks differ with region?"

We will formulate our hypotheses as:

$$H_0 : p_{east} - p_{west} = 0$$
$$H_A : p_{east} - p_{west} \neq 0$$

We can calculate a $z - score$ test statistic to test this hypothesis:

$$z = \frac{\hat{p_{east}} - \hat{p_{west}} - 0}{\sqrt{\frac{\hat{p_{east}}(1-\hat{p_{east}})}{n_{east}} + \frac{\hat{p_{west}}(1-\hat{p_{west}})}{n_{west}}}}$$
$$z = \frac{0.44 - 0.54 - 0}{\sqrt{\frac{0.44(1-0.44)}{432} + \frac{0.54(1-0.54)}{293}}}$$
$$z = -2.66$$

```r
z_score <- (0.44 - 0.54 - 0) / sqrt( (0.44*(1-0.44)/432) + (0.54*(1-0.54)/293))

z_score
```

```
## [1] -2.655453
```

Now that we have a z-score, we can calculate a p-value. Our hypotheses calls for a two-sided test, so:

```r
p_value = 2 * pnorm(q=-2.66, mean=0, sd=1)

p_value
```

```
## [1] 0.007814065
```

Computational method with Infer

Using infer we will generate the null distribution using the permutation method. Two appropriate test statistics are, 1) the difference in proportions, 2) a z-score. First calculating a z-score.

```r
test_stat <- df_two %>%
  specify(Type ~ Region, success = "disk") %>%
  calculate(stat = "z", order = c("east","west"))

test_stat
```

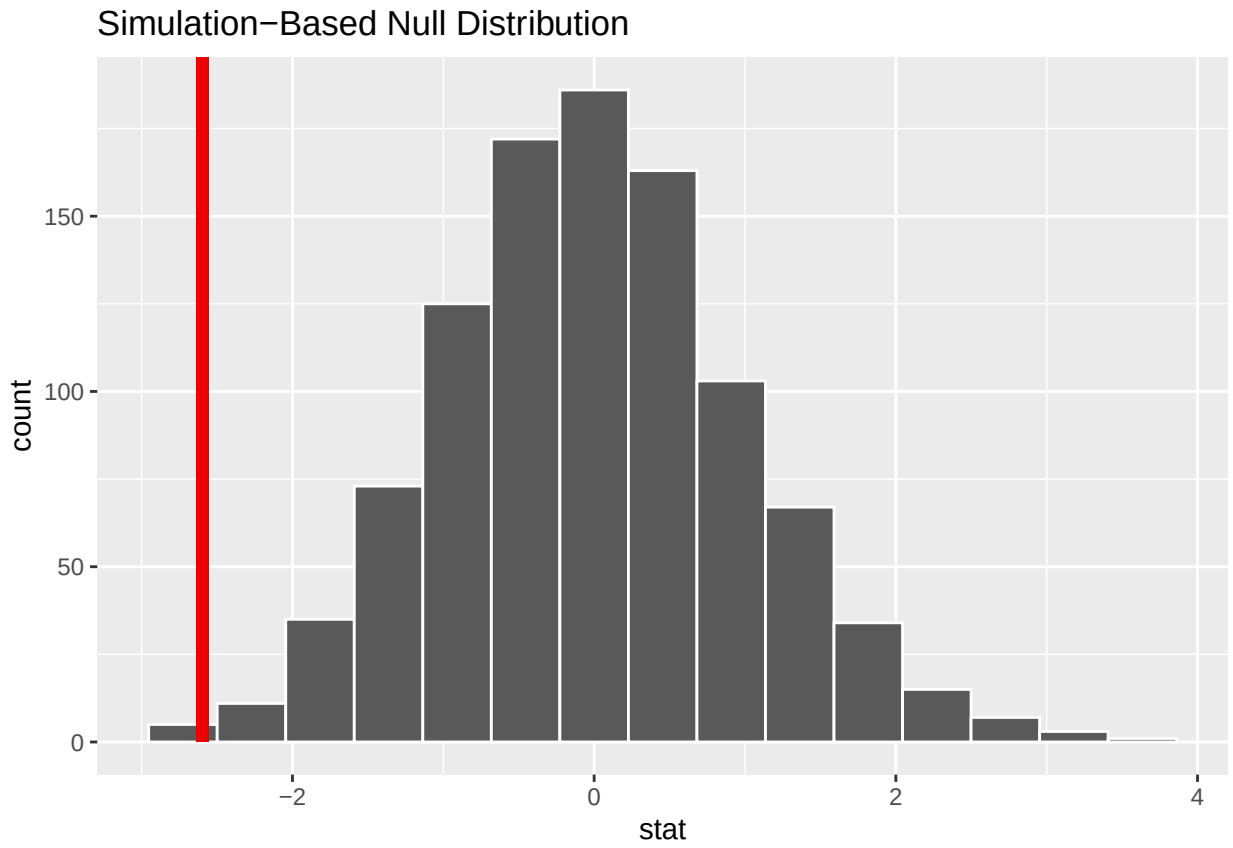
```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 -2.60
```

```r
null_dist <- df_two %>%
  specify(Type ~ Region, success = "disk") %>%
  hypothesise(null="independence") %>%
  generate(reps=1000,type="permute") %>%
  calculate(stat = "z", order = c("east","west"))

p_value <- null_dist %>%
  get_p_value(obs_stat=test_stat, direction="two-sided")

p_value
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1    0.01
```

```r
null_dist %>% visualise(obs_stat = test_stat)
```

```
## Warning: `visualize()` should no longer be used to plot a p-value.
## Arguments `obs_stat`, `obs_stat_color`, `pvalue_fill`, and `direction` are
## deprecated. Use `shade_p_value()` instead.
```



Now we will use the difference in proportions as the test statistic.

```
test_stat <- df_two %>%
  specify(Type ~ Region, success = "disk") %>%
  calculate(stat = "diff in props", order = c("east","west"))

test_stat
```

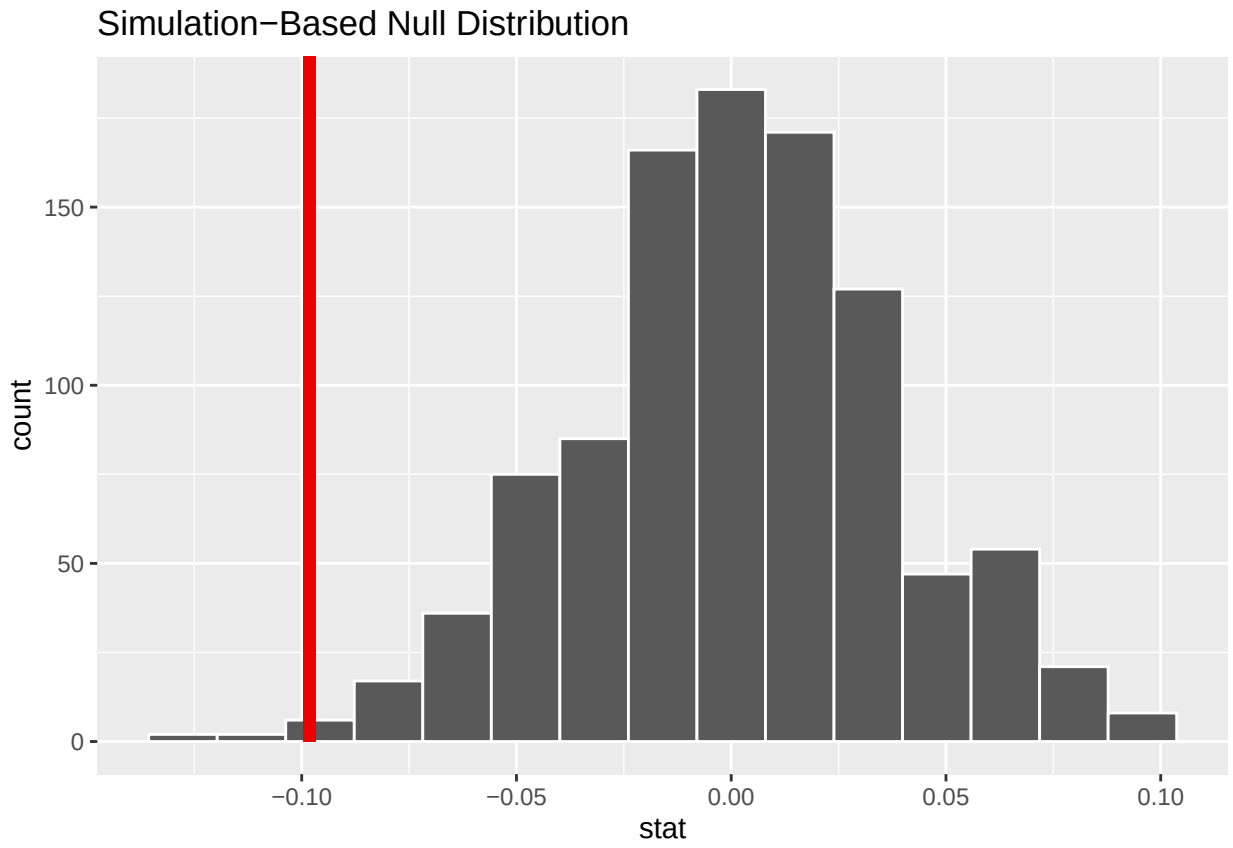
```
## # A tibble: 1 x 1
##      stat
##     <dbl>
## 1 -0.0982
```

```
null_dist <- df_two %>%
  specify(Type ~ Region, success = "disk") %>%
  hypothesise(null="independence") %>%
  generate(reps=1000,type="permute") %>%
  calculate(stat = "diff in props", order = c("east","west"))

p_value <- null_dist %>%
  get_p_value(obs_stat=test_stat, direction="two-sided")

p_value
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1   0.014
```

```
null_dist %>% visualise(obs_stat=test_stat)
```

```
## Warning: `visualize()` should no longer be used to plot a p-value.
## Arguments `obs_stat`, `obs_stat_color`, `pvalue_fill`, and `direction` are
## deprecated. Use `shade_p_value()` instead.
```



This p-value is quite small, and is strong evidence to reject the null hypothesis. Therefore, we conclude that the proportion of stars with disks differs between the "east" and "west" regions.

More than two Categories

Confidence Intervals

The X^2 distribution is not symmetric, so it does not make sense to consider a confidence interval.

Hypothesis testing

We are still asking the question, "Is disk proportion related to Region". Our null hypothesis is that disk proportion is independent of region and thus our alternative hypothesis is that disk proportion is dependent on region.

We will formulate our hypotheses as:

$$H_0 : p_{east} = p_{west} = p_{halo}$$
$$H_A : at\ least\ one\ p\ is\ different$$

Approximation with Probability Models

We can use the X^2 distribution to test this hypothesis. First lets look at the two way table.

counts	east	west	halo	total
disk	192	159	507	858
no disk	240	134	929	1303
total	432	293	1436	2161

We can estimate the number of expected counts in any cell using the following:

$$Expected_{i,j} = \frac{n_{row\ i} \times n_{column\ j}}{table\ total}$$

Our table of expected counts is then

counts	east	west	halo	total
disk	$\frac{858 \times 432}{2161} = 171.5$	$\frac{858 \times 293}{2161} = 116.3$	$\frac{858 \times 1436}{2161} = 570.2$	858
no disk	$\frac{1303 \times 432}{2161} = 260.5$	$\frac{1303 \times 293}{2161} = 176.7$	$\frac{1303 \times 1436}{2161} = 865.9$	1303
total	432	293	1436	2161

To calculate the X^2 test statistic, we first calculate a z-score for each cell.

$$z = \frac{observed\ count - null\ count}{Standard\ Error}$$
$$z = \frac{observed\ count - null\ count}{\sqrt{null\ count}}$$

Then each of these terms are squared and added together.

```
chi_sq_stat <- ( (192 - 171.5)/sqrt(171.5) )^2 +
          ( (159 - 116.3)/sqrt(116.3) )^2 +
          ( (507 - 570.2)/sqrt(570.2) )^2 +
          ( (240 - 260.5)/sqrt(260.5) )^2 +
          ( (134 - 176.7)/sqrt(176.7) )^2 +
          ( (929 - 865.9)/sqrt(865.9) )^2

chi_sq_stat
```

```
## [1] 41.66293
```

If R is the number of categories in the response variable, and C is the number of categories in the explanetory variable, then the degrees of freedom for a X^2 test are given by:

$$df = (R-1) \times (C-1)$$
$$df = (2-1) \times (3-1)$$
$$df = 1 \times 2$$
$$df = 2$$

We can now calculate the p-value

```
p_value <- 1 - pchisq(q=chi_sq_stat,df=2)

p_value
```

```
## [1] 8.974491e-10
```

We can use the R function `chisq.test()` to perform a hypothesis test assuming the X^2 probability model.

```
df_three <- df %>%
  filter(Type %in% c("disk","no disk")) %>%
  select(Type,Region) %>%
  mutate(Type = fct_drop(Type))

chisq.test(table(df_three$Type, df_three$Region))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(df_three$Type, df_three$Region)
## X-squared = 41.609, df = 2, p-value = 9.22e-10
```

Computational method with Infer

Note: `fct_drop()` is used here to remove the empty "envelope" category.

```
df %>%
  filter(Type %in% c("disk","no disk")) %>%
  select(Type,Region) %>%
  group_by(Region,Type) %>%
  summarise(count=n())
```

```
## # A tibble: 6 x 3
## # Groups:   Region [3]
##   Region Type    count
##   <chr>  <chr>   <int>
## 1 east   disk      192
## 2 east   no disk   240
## 3 halo   disk      507
## 4 halo   no disk   929
## 5 west   disk      159
## 6 west   no disk   134
```

```
df_three <- df %>%
  filter(Type %in% c("disk","no disk")) %>%
  select(Type,Region) %>%
  mutate(Type = fct_drop(Type))

test_stat <- df_three %>%
  specify(Type ~ Region) %>%
  calculate(stat = "Chisq")


test_stat
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1  41.6
```

```
null_dist <- df_three %>%
  specify(Type ~ Region) %>%
```

```
  hypothesize(null = "independence") %>%
  generate(reps = 1000, type = "permute") %>%
  calculate(stat = "Chisq")

null_dist
```

```
## # A tibble: 1,000 x 2
##    replicate  stat
##        <int> <dbl>
##  1         1 0.585
##  2         2 1.24
##  3         3 2.06
##  4         4 1.54
##  5         5 2.88
##  6         6 1.48
##  7         7 1.17
##  8         8 1.35
##  9         9 0.396
## 10        10 6.93
## # ... with 990 more rows
```

```
p_value <- null_dist %>%
  get_p_value(obs_stat = test_stat, direction="greater")
```

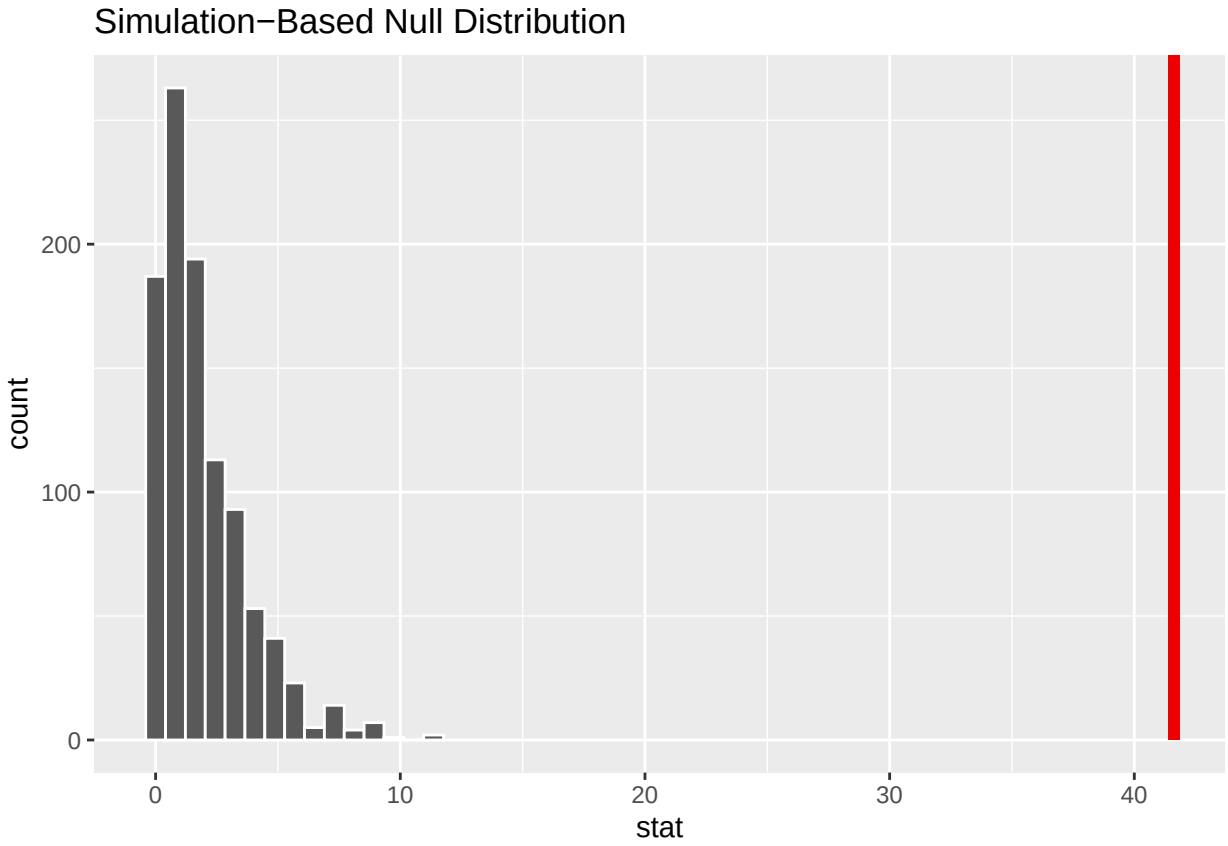
```
## Warning: Please be cautious in reporting a p-value of 0. This result is
## an approximation based on the number of `reps` chosen in the `generate()`
## step. See `?get_p_value()` for more information.
```

```
p_value
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

```
null_dist %>% visualise(obs_stat=test_stat)
```

```
## Warning: `visualize()` should no longer be used to plot a p-value.
## Arguments `obs_stat`, `obs_stat_color`, `pvalue_fill`, and `direction` are
## deprecated. Use `shade_p_value()` instead.
```



As we see, regardless of method used, the test statistic is far into the wings of the null distribution. Therefore, we consider that to be strong evidence to reject the null hypothesis that disk proportion and region are independent. We conclude then, that disk proprtion depends on which region a star is in.

Assumptions and Conditions

Assumptions for probability models

Can refer to the summary tables.

For each proportion we want at least 10 successes and failures, which we have. For the X^2 model the conditions are that each observation is independent, which they are, that each category must have at least 5 counts, they do, and that

Assumptions for computational method

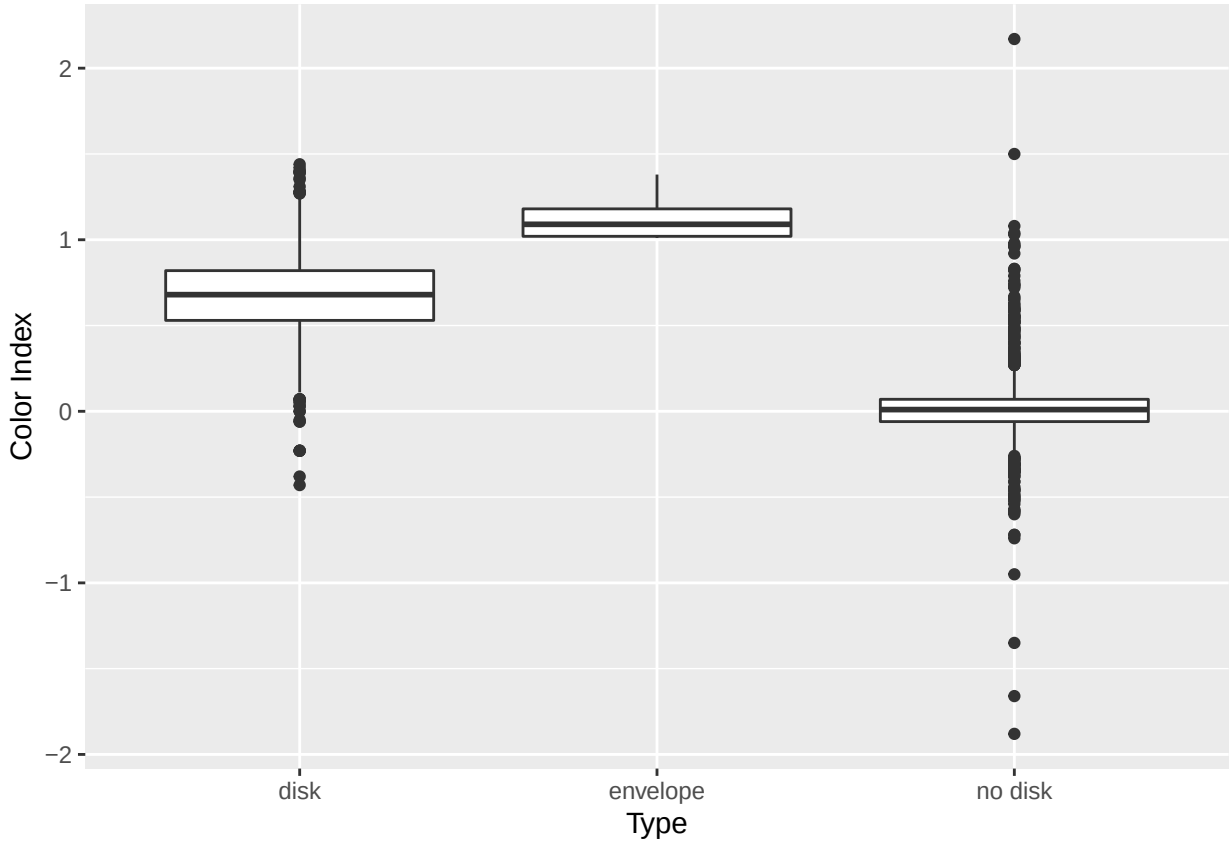
We want independent observations, at least 30 observations and the resulting distribution is nearly normal. All are met.

Problem 2

We are now interested in exploring whether there are difference in the type of light emitted from each `Type` of star. We will do this by creating a new *color index* composed of the measurements made in certain Bands. To do this create a new variable called `Color` made of the difference `Band8-Band9`. You will want to remove `NA`'s from this variable.

Exploratory Data Analysis

```
df_colors <- df %>%
  mutate(Color = Band8 - Band9) %>%
  drop_na(Color)


# Boxplot of Color as function of Type (Disk or No Disk)
df_colors %>%
  ggplot(aes(x=Type, y=Color)) +
    geom_boxplot() +
    scale_y_continuous(name="Color Index")
```




```
df_color_table <- df_colors %>%
  group_by(Type) %>%
  summarize(mean = mean(Color),sd=sd(Color),count=n())

df_color_table
```

```
## # A tibble: 3 x 4
##   Type       mean    sd count
##   <chr>     <dbl> <dbl> <int>
## 1 disk     0.673  0.254   833
## 2 envelope 1.14   0.152     5
## 3 no disk  0.0266 0.229  1206
```

Two Categories

Confidence intervals

First, we just consider the regions "disk" and "no disk", and for eash type estimate the mean value of the color index and determine 95% confidence intervals.

Approximation with probability models

For a single mean point estimate:

$$\bar{x} \pm MOE$$
$$\bar{x} \pm t^*_{df} \times \frac{s}{\sqrt{n}}$$

The "disk" stars have a mean `Color` of 0.6732893 with n = 833 observations, a sample standard deviation, s = 0.2537412 and df = n-1 = 832 degrees of freedom. For a sample of this size $t^*_{df} \simeq z^*$.

$$\bar{x_{disk}} \pm z^* \times \frac{s}{\sqrt{n}}$$
$$0.6732893 \pm 1.96 \times \frac{0.2537412}{\sqrt{833}}$$
$$0.6732893 \pm 0.02$$

R can be used as a calculator.

```
1.96 * df_color_table[df_color_table$Type == "disk","sd"] /
  sqrt(df_color_table[df_color_table$Type == "disk","count"])
```

```
##           sd
## 1 0.01723156
```

The "no disk" stars have a mean `Color` of 0.0265755 with n = 1206 observations, a sample standard deviation, s = 0.2294564 and df = n-1 = 1205 degrees of freedom. Again, for a sample of this size $t^*_{df} \simeq z^*$.

$$\bar{x_{disk}} \pm z^* \times \frac{s}{\sqrt{n}}$$
$$0.0265755 \pm 1.96 \times \frac{0.2294564}{\sqrt{1206}}$$
$$0.0265755 \pm 0.01$$

```
1.96 * df_color_table[df_color_table$Type == "no disk","sd"] /
  sqrt(df_color_table[df_color_table$Type == "no disk","count"])
```

```
##           sd
## 1 0.01295038
```

Computational method with Infer

We will use `infer` to construct bootstrap estimations of the sampling distribution.

```
df_two <- df_colors %>%
  filter(Type %in% c("disk","no disk")) %>%
  mutate(Type = fct_drop(Type))
```

```r
# disk
mean_disk <- df_two %>%
  filter(Type == "disk") %>%
  specify(response = Color) %>%
  calculate(stat = "mean")

boot_disk <- df_two %>%
  filter(Type == "disk") %>%
  specify(response = Color) %>%
  generate(reps=1000,type="bootstrap") %>%
  calculate(stat = "mean")

ci_disk <- boot_disk %>%
  get_confidence_interval(type = "se", point_estimate = mean_disk)
```

```
## Using `level = 0.95` to compute confidence interval.
```

```r
mean_disk
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 0.673
```

```r
ci_disk
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1    0.656    0.691
```

```r
# no disk
mean_nodisk <- df_two %>%
  filter(Type == "no disk") %>%
  specify(response = Color) %>%
  calculate(stat = "mean")

boot_nodisk <- df_two %>%
  filter(Type == "no disk") %>%
  specify(response = Color) %>%
  generate(reps=1000,type="bootstrap") %>%
  calculate(stat = "mean")

ci_nodisk <- boot_nodisk %>%
  get_confidence_interval(type = "se", point_estimate = mean_nodisk)
```

```
## Using `level = 0.95` to compute confidence interval.
```

```r
mean_nodisk
```

```
## # A tibble: 1 x 1
##     stat
##    <dbl>
## 1 0.0266
```

```r
ci_nodisk
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   0.0135   0.0396
```

Hypothesis testing

Now answer the question, "Does `Color` depend on `Type`?" Formulate a null and alterntive hypothesis, calculate a test statistic and a p-value and then make a conclusion about your hypotheis.

We will formulate our hypotheses as:

$$H_0 : \mu_{disk} - \mu_{no\ disk} = 0$$
$$H_A : \mu_{disk} - \mu_{no\ disk} \neq 0$$

Approximation with probability models

We will first approximate the null distribution using a t-distribution.

$$t = \frac{\bar{x_{disk}} - \bar{x_{no\ disk}} - 0}{\sqrt{\frac{s^2_{disk}}{n_{disk}} + \frac{s^2_{no\ disk}}{n_{no\ disk}}}}$$
$$t = \frac{0.67\bar{3}2893 - 0.02\bar{6}5755 - 0}{\sqrt{\frac{0.2294564^2}{833} + \frac{0.2294564^2}{1206}}}$$
$$t = 58.8$$

Using R as a calculator:

```
t_stat <- (df_color_table[df_color_table$Type == "disk","mean"] -
             df_color_table[df_color_table$Type == "no disk","mean"]) /
  sqrt ( (df_color_table[df_color_table$Type == "disk","sd"]^2) /
           df_color_table[df_color_table$Type == "disk","count"] +
           (df_color_table[df_color_table$Type == "no disk","sd"]^2) /
           df_color_table[df_color_table$Type == "no disk","count"] )

d_freedom <- min(df_color_table[df_color_table$Type == "no disk","count"],
         df_color_table[df_color_table$Type == "disk","count"]) - 1

#df
t_stat <- t_stat %>% pull()

t_stat
```

```
## [1] 58.80447
```

And `pt()` can be used to calculate th p-value:

```
p_value <- 2*(1 - pt(q=t_stat,d_freedom))

p_value
```

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

Another approach is to use the R function `t.test()`.

```r
df_two <- df_colors %>%
  filter(Type %in% c("disk","no disk")) %>%
  mutate(Type = fct_drop(Type))

t.test(Color ~ Type,data=df_two)
```

```
##
##  Welch Two Sample t-test
##
## data:  Color by Type
## t = 58.804, df = 1669.5, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.6251431 0.6682846
## sample estimates:
##    mean in group disk mean in group no disk
##             0.67328932             0.02657546
```

Computational method with Infer

```r
test_stat <- df_two %>%
  specify(Color ~ Type) %>%
  calculate(stat = "t",order = c("disk", "no disk"))

test_stat
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1  58.8
```

```r
null_dist <- df_two %>%
  specify(Color ~ Type) %>%
  hypothesise(null="independence") %>%
  generate(reps=1000,type="permute") %>%
  calculate(stat = "t",order = c("disk", "no disk"))

p_value <- null_dist %>%
  get_p_value(obs_stat=test_stat, direction="two-sided")
```

```
## Warning: Please be cautious in reporting a p-value of 0. This result is
## an approximation based on the number of `reps` chosen in the `generate()`
## step. See `?get_p_value()` for more information.
```

```r
p_value
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

No matter the method we use, we find that the t-score is large and the associate p-value is small, giving us confidence that we can reject the null hypothesis.

More than two Categories

We now consider all three types: "envelope", "disk" and "no disk".

We will formulate our hypotheses as:

$$H_0 : \mu_{disk} = \mu_{no\ disk} = \mu_{envelope}$$
$$H_A : at\ least\ one\ \mu\ is\ different$$

Approximation with Probability Models

The `aov()` function will apply the F-distribution model.

```
mod <- aov(Color ~ Type, data=df_colors)

tidy(mod)
```

```
## # A tibble: 2 x 6
##   term         df sumsq   meansq statistic p.value
##   <chr>     <dbl> <dbl>    <dbl>     <dbl>   <dbl>
## 1 Type          2  210. 105.         1827.       0
## 2 Residuals  2041  117.   0.0574       NA       NA
```

Computational method with Infer

```
f_stat <- df_colors %>%
  specify(Color ~ Type) %>%
  calculate(stat = "F")

f_stat
```

```
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1 1827.
```

```
null_dist <- df_colors %>%
  specify(Color ~ Type) %>%
  hypothesise(null="independence") %>%
  generate(reps=1000,type="permute") %>%
  calculate(stat = "F")


p_value <- null_dist %>%
  get_p_value(obs_stat=f_stat,direction="greater")
```

```
## Warning: Please be cautious in reporting a p-value of 0. This result is
## an approximation based on the number of `reps` chosen in the `generate()`
## step. See `?get_p_value()` for more information.
```

```
p_value
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1       0
```

We find a large F-stat and small p-value with each method. We will reject the null hypothesis that `Color` is independent of `Type` (at least if `Color = Band8 - Band9`).

Assumptions and Conditions

Assumptions for probability models

Can refer to the summary tables.

For each mean we want at least 30 observations or the data are normal, which we have. For the ANOVA model the conditions are that each observation is independent, which they are, that the data in each group is nearly normal and each group has similar variance. The "disk" and "no disk" groups have similar variances, however, the "envelope" group has a smaller variance and very few observations.

Assumptions for computational method

We want independent observations, at least 30 observations and the resulting distribution is nearly normal. All are met.