|  |
| --- |
| **Loading a Package**  library(PACKAGE NAME) |
| **Reading in Data**  NAME OF DATASET <- read\_csv(“PATH & NAME OF DATASET.csv”)  ***Note:*** The name of the dataset will change, but it will always need to have the .csv at the end of its name!  ***Note:*** Do not put spaces in the name you give the data set. |
| **Preview a Dataset**  glimpse(NAME OF DATASET)  head(NAME OF DATASET) – shows first 6 rows  names(NAME OF DATASET) – outputs the names of the columns/variables |
| **Plotting a One Categorical Variable Bar Plot with Counts**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF VARIABLE)) +  geom\_bar(stat = “count”) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** This bar plot has the variable names on the x-axis. If the names are squished, then you should use  **y =** NAME OF VARIABLE instead of **x =** NAME OF VARIABLE. |
| **Plotting a One Categorical Variable Bar Plot with Proportions**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF VARIABLE)) +  geom\_bar(stat = “count”, aes(y = ..prop.., group = 1)) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** This bar plot has the variable names on the x-axis. If the names are squished, then you should use  **y =** NAME OF VARIABLE instead of **x =** NAME OF VARIABLE. |
| **Creating a Summary Table of Observations of One Categorical Variable**  NAME OF DATASET |>  count(NAME OF VARIABLE) |
| **Conducting an Exact Binomial Hypothesis Test for One Proportion**  binom.test(x = NUMBER OF SUCCESSES, n = SAMPLE SIZE, p = NULL VALUE, alternative = “DIRECTION”)  ***Note:*** The alternative direction can be “greater”, “less”, or “two.sided” |
| **Performing a Chi-Squared Goodness-of-Fit Test (One Categorical Variable)**  chisq\_test(x = NAME OF DATASET,  response = NAME OF VARIABLE,  p = c(EXPECTED PROP 1, EXPECTED PROP 2,…, EXPECTED PROP LAST)  )  \*Make sure expected proportions are in alphabetical order!  \*Make sure to check conditions first! |
| **Plotting a Two Categorical Variable Bar Plot**  ggplot(data = NAME OF DATASET,  mapping = aes(x = EXPLANATORY VARIABLE,  fill = RESPONSE VARIABLE)  ) +  geom\_bar(stat = “count”,  position = “fill”) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** If you want a side-by-side bar plot you need to change position to “dodge”. If you want a stacked bar plot, you need change position to “stack”. |
| **Creating a Contingency Table of Observed Counts from Two Categorical Variables**  NAME OF DATASET |>  tabyl(EXPLANATORY VARIABLE, RESPONSE VARIABLE) |>  adorn\_totals(where = c(“row”, “col”))  ***Note:*** Requires the janitor package to be loaded – library(janitor) |
| **Creating a Contingency Table of Observed Proportions from Two Categorical Variables**  NAME OF DATASET |>  tabyl(EXPLANATORY VARIABLE, RESPONSE VARIABLE) |>  adorn\_totals(where = c(“row”, “col”)) |>  adorn\_percentages(denominator = “row”)  ***Note:*** Since your explanatory variable (groups) should be in your rows from above, we want to calculate our proportions in respect to the group totals. |
| **Performing a Chi-Square Test (Two Categorical Variables)**  chisq\_test(x = NAME OF DATASET,  response = RESPONSE VARIABLE,  explanatory = EXPLANATORY VARIABLE,  correct = FALSE,  simulate.p.value = FALSE / TRUE  )  \*Make sure to check conditions first! If conditions are not met, must include the simulate.p.value = TRUE. |
| **Calculating Summary Statistics for One Numeric Variable**  favstats(~ NAME OF VARIABLE, data = NAME OF DATASET)  ***Note:*** The ~ (top left keyboard) **must** be included *before* the variable's name! |
| **Histogram for One Numeric Variable**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF VARIABLE)) +  geom\_histogram(binwidth = WIDTH OF BINS, color = “white”) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** A histogram **must** have a numeric variable on the x-axis! If your variable has a space in it, you will need to use tick marks. |
| **Dotplot for One Numeric Variable**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF VARIABLE)) +  geom\_dotplot() +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** A dotplot **must** have the variable on the x-axis! |
| **Boxplot for One Numeric Variable**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF VARIABLE)) +  geom\_boxplot() +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “”)  ***Note:*** This boxplot is horizontal. If you want for your boxplot to be vertical, in the mapping aes(), you use **y =** instead of **x =** . Keep in mind you will need to change the location of you axis label, too! |
| **Performing a t-test for One Mean (and Confidence Interval)**  t\_test(x = NAME OF DATASET,  response = NAME OF VARIABLE,  mu = VALUE FROM NULL HYPOTHESIS FOR Mu,  alternative = "two-sided",  conf\_level = 0.95)  ***Note:*** If you want a 90% confidence interval, you change conf\_level to 0.90. If you want a 99% confidence interval, you change conf\_level to 0.99  ***Note:*** If you are doing a one-sided hypothesis test, you change alternative to either “greater” or “less” |
| **Calculating Summary Statistics for One Numerical Variable and One Categorical Variable**  favstats(NAME OF NUMERICAL VARIABLE ~ NAME OF CATEGORICAL VARIABLE,  data = NAME OF DATASET)  ***Note:*** The ~ **must** be included! This is from the mosaic plot. |
| **Faceted Histograms**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF NUMERICAL VARIABLE)) +  geom\_histogram(binwidth = WIDTH OF BINS, color = “white”) +  facet\_wrap(~NAME OF CATEGORICAL VARIABLE) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** A histogram **must** have the variable on the x-axis! |
| **Side-by-Side Boxplots**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF NUMERICAL VARIABLE,  y = NAME OF CATEGORICAL VARIABLE)  ) +  geom\_boxplot()+  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** For **vertically stacked** boxplots, the categorical variable should be on the **y-axis.** For **horizontally stacked** boxplots, the categorical variable should be on the **x-axis**. |
| **Performing a Two-Sample Independent t-test (Difference in Means)**  t\_test(x = NAME OF DATASET,  response = NAME OF NUMERICAL VARIABLE,  explanatory = NAME OF CATEGORICAL VARIABLE,  mu = 0,  conf\_int = TRUE,  conf\_level = 0.95,  alternative = "two-sided")  ***Note:*** If you want a 90% or 99% confidence interval, you change conf\_level to 0.90 or 0.99  ***Note:*** If you are doing a one-sided hypothesis test, you change alternative to either “greater” or “less” |
| **Performing an ANOVA (F-test)**  model <- aov(NAME OF NUMERICAL VARIABLE ~ NAME OF CATEGORICAL VARIABLE,  data = NAME OF DATA SET)  model |>  tidy()  ***Note:*** The **~** is necessary! It has to be there!  ***Note:*** *tidy() is from the broom package – library(broom)* |
| **Performing an ANOVA (Pairwise Comparisons)**  library(emmeans)  emmeans(NAME\_OF\_MODEL, specs = ~ NAME OF CATEGORICAL VARIABLE) |>  pairs(adjust = “MULTIPLICITY ADJUSTMNET METHOD”)  ***Note:*** *The multiplicity adjustment method can be* tukey, bonf, none, sidak*, to name a few.* |
| **Scatterplot between two numeric variables**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF EXPLANATORY-VARIABLE,   y = NAME OF RESPONSE-VARIABLE)  ) +  geom\_point() +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”) |
| **Scatterplot with Regression Line**  ggplot(data = NAME OF DATASET,  mapping = aes(x = NAME OF EXPLANATORY-VARIABLE,   y = NAME OF RESPONSE-VARIABLE)  ) +  geom\_point() +  geom\_smooth(method = “lm”, se = F) +  labs(title = “TITLE FOR GRAPH”,  x = “TITLE FOR THE X-AXIS”,  y = “TITLE FOR THE Y-AXIS”)  ***Note:*** *The line is added to the above scatterplot by adding in the geom\_smooth() code line.*  ***Note:*** *Just because you have added a line to your scatterplot doesn’t mean* |
| **Fitting a Linear Regression Model**  model <- lm(NAME OF RESPONSE-VARIABLE ~ NAME OF EXPLANATORY-VARIABLE,  data = NAME OF DATASET)  model |>  tidy(conf.int = TRUE,  conf.level = 0.95)  ***Note:*** The **~** is necessary! It has to be there!  ***Note:*** If you want a 90% confidence interval, you change conf.level to 0.90  ***Note:*** *tidy() is from the broom package* |
| **Checking Assumptions for Linear Regression**  library(easystats)  check\_model(model)  ***Note****: some of these are checked with the scatterplot, histograms, and critical thinking instead!*  ***Note:*** *model is the name of the model from fitting the linear regression model.* |