|  |  |  |  |
| --- | --- | --- | --- |
| **Basics**  print("hello")  # assigns value to object  my\_number <- 5  # combines elements into vector  my\_vector <- c(1,2,3)  # first element in vector  my\_vector[1]  # variable in data frame  Fingers$Sex | **Tables**  tally(my\_vector)  tally(~ Condition, data = MindsetMatters)  tally(~ Thumb > 65, data = Fingers)  tally(Thumb ~ Sex, data = Fingers, margins = TRUE, format = "proportion") | **Fitting and Evaluating Models**  empty\_model <- lm(Thumb ~ NULL, data = Fingers)  Sex\_model <- lm(Thumb ~ Sex, data = Fingers)  predict(empty\_model)  resid(empty\_model)  anova(empty\_model)  supernova(Sex\_model)  t.test(Tip ~ Condition, data = TipExperiment, var.equal=TRUE)  pairwise(game\_model, correction="none") | **Data**  str(MindsetMatters)  head(MindsetMatters)  tail(MindsetMatters)  sort(my\_vector)  arrange(Fingers, Thumb)  # selects variables  select(Fingers, Sex, RaceEthnic, Thumb)  # selects cases  filter(Fingers, SSLast != "NA")  head(select(Fingers, Thumb))  as.numeric(Fingers$Interest)  factor(Fingers$Sex)  factor(Fingers$Sex, levels = c(1,2), labels = c("female", "male"))  recode(Fingers$Job, "0" = 0, "1" = 50, "2" = 100)  # creates two equal sized groups  ntile(Fingers$Height, 2)  # creates data frame from csv file  new\_dataframe <- read.csv("long-csv-link-from-published-google-spreadsheet", header = TRUE) |
| **Operators**  sum(1,2,100)  +, -, \*, /  >, <, >=, <=, ==, !=  # results in TRUE or FALSE  Fingers$RingLonger <- Fingers$Ring > Fingers$Index  abs(Fingers$Residual)  Fingers$Residual^2  sqrt(157) | **Simple Statistics**  mean(Fingers$Thumb)  var(Fingers$Thumb)  sd(Fingers$Thumb)  favstats(~ Wt, data = MindsetMatters)  cohensD(Thumb ~ Sex, data = Fingers)  cor(Thumb ~ Height, data = Fingers)  b1(Thumb ~ Sex, data = Fingers)  b1(Sex\_model)  # PRE and fVal work like b1  PRE(Sex\_model)  fVal(Sex\_model) | **Probability Distributions**  xpnorm(65.1, Thumb\_stats$mean, Thumb\_stats$sd)  zscore(Fingers$Thumb)  # returns t at this probability  qt(.975, df = 999)  # returns F at this probability  qf(.95, df1 = 1, df2 = 100)  # CI using t dist.  confint(empty\_model) |
| **Simulation & Resampling**  # sample without replacement  sample(Fingers$Thumb, 10)    # sample with replacement  resample(Fingers$Thumb, 157)  do(3) \* resample (Fingers$Thumb,10)  # mixes up values in a variable  shuffle(servers$random\_groups\_1)  # simulates sampling 10000 Thumbs from a normal dist.  sim\_Thumb <- rnorm(10000, Thumb\_stats$mean, Thumb\_stats$sd) | # puts simulated Thumbs into data frame  sim\_Pop <- data.frame(sim\_Thumb)  # simulates sampling dist.of means  sim\_SDoM <- do(10000) \* mean(rnorm(157, Thumb\_stats$mean, Thumb\_stats$sd))  # bootstraps sampling dist. of means  bootSDoM <- do(10000) \* mean(resample(Fingers$Thumb,157))  # bootstraps sampling dist. of b1s, centered on sample b1  SDob1 <- do(10000) \* b1(Tip ~ Condition, data = resample(TipExperiment, 44)) | # randomizes sampling dist. of b1s, centered on 0  SDob1 <- do(10000) \* b1(Tip ~ shuffle(Condition), data = TipExperiment)  #get the middle 95 percent of the distribution  middle(b1, .95)  # randomizes sampling dist. of PREs  SDoPRE <- do(10000) \* PRE(Tip ~ shuffle(Condition), data = TipExperiment)  # randomizes sampling dist. of Fs  sdoF <- do(10000) \* fVal(Tip ~ shuffle(Condition), data = TipExperiment) | # plots sampling dist.  gf\_histogram(~ fVal, data = SDoF, fill = ~fVal>sample\_F)  # counts extreme Fs  tally(~fVal>sample\_F, data = SDoF)  # count the number of b1s at the upper and lower extreme  tally(sdob1$b1 > sample\_b1 | sdob1$b1 < -sample\_b1)  # fill the lower .95 of this histogram with a different color  gf\_histogram(~ fVal, data = sdoF, fill = ~lower(fVal, .95))  # calculate a p-value using the F-distribution  xpf(sample\_F, df1 = , df2 = ) |

Version 3.0

|  |  |  |
| --- | --- | --- |
| **Plots**  gf\_histogram( ~ Thumb, data = Fingers, fill = "orange", color = "gray", bins = 10) %>%  # changes labels  gf\_labs(title= "Student Thumb Lengths", x = "Thumb Length (mm)") %>%  # adds density curve to a histogram  gf\_density() %>%  # adds vertical line  gf\_vline(xintercept = ~mean, data = Thumb\_stats, color = "blue", size = 2)    gf\_dhistogram(~ Thumb, data = Fingers) %>%  gf\_facet\_grid(Sex ~ .)    gf\_histogram(~b1, data = sdob1, fill = ~middle(b1, .95))  Chart, histogram  Description automatically generated | gf\_dist("norm", color = "blue", params = list(Thumb\_stats$mean, Thumb\_stats$sd))    gf\_boxplot(Wt ~ 1, data = MindsetMatters)    gf\_boxplot(Thumb ~ Height3Group, data = Fingers)    # creates bar graph  gf\_bar( ~ Sex, data = Fingers) | gf\_point(Thumb ~ Sex, data = Fingers, color = "orange", size = 5)    gf\_jitter(Thumb ~ Sex, data = Fingers, color = "orange", size = 5, alpha = .5)    gf\_point(Thumb ~ Height, data = Fingers, size = 2) %>%  # adds a regression line  gf\_lm(color = "orange", size = 2)    pairwise(game\_model, plot = TRUE)  Chart, box and whisker chart  Description automatically generated |