# Lab Exercise 3: Basic Statistics, Visualization, and Hypothesis Tests

|  |  |
| --- | --- |
| **Purpose:** | The lab introduces you to the analysis of data using the R statistical package within the Data Science and Big Data Analytics environment. After completing the tasks in this lab you should able to:   * Perform summary (descriptive) statistics on the data sets * Create basic visualizations using R both to support investigation of the data as well as exploration of the data * Create plot visualizations of the data using a graphics package * Test a hypothesis about the data |
|  | |
| **Tasks:** | Tasks you will complete in this lab include:   * Reload data sets into the R statistical package * Perform summary statistics on the data * Remove outliers from the data * Plot the data using R * Plot the data using lattice and ggplot * Test a hypothesis about the data |
|  | |
| **References:** | References used in this lab are located in your ***Student Resource Guide Appendix****.* See the Appendix for:   * R Commands – Quick Reference * Surviving LINUX – Quick Reference |

Part 1 – Basic Statistics and Visualization Using R

### Workflow Overview

### LAB Instructions

| **Step** | **Action** |
| --- | --- |
| 1 | **Prepare working environment for the Lab and load data files**   1. Set the working directory to LAB01 where we have stored the data. On the console window type:   **setwd("~/LAB01")**   1. In the script window, open the script called “Module3Lab2.R”. (Click on “File”, “Open File” and Navigate to directory LAB03 and click on file “Module3Lab2.R”).   Start R and Read the Data Set Back Into Your Workspace:   1. Execute the following commands from the script window:   **options(digits=3) options(width=68)**  **ls() load(file=”Labs.Rdata”) ls()**  **rm(lab2)**  **ds <- lab1 colnames(ds) <- c("income", "rooms")** |
| 2 | **Obtain summary statistics for Household Income and visualize data:**   1. Execute the following commands from the script window:   **summary(ds$income) range(ds$income) sd(ds$income) var(ds$income)**  **plot(density(ds$income)) # left skewed**   1. What is the mean? \_\_\_\_\_\_\_\_\_\_\_ 2. What is the median? \_\_\_\_\_\_\_\_\_\_ 3. What is the standard deviation? \_\_\_\_\_\_\_\_\_\_\_\_\_ |
| 3 | **Obtain summary statistics for Number of rooms and visualize data:**  Execute the following commands from the script window:  **summary(ds$rooms) range(ds$rooms) sd(ds$rooms) plot(as.factor(ds$rooms))**  What is the mean?  What is the median?  What is the standard deviation? |
| 4 | **Remove Outliers**  In a previous lab, you recorded the range of income. You observed that the minimum household income is 4, and the maximum is 1,620,560.   1. Does this make sense to you? Why? \*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2. What happens if you throw out the top and bottom 10%? Execute the following line from the script window     **(m <- mean(ds$income, trim=0.10) )**   1. How does this compare to the previous mean of this variable? 2. Execute the following commands from the script window:   **ds <- subset(ds, ds$income >= 10000 & ds$income < 1000000) summary(ds) quantile(ds$income, seq(from=0, to=1, length=11))**   1. How do these values vary from the values in the original data set? 2. Do they make more sense? 3. Which data set would you prefer to use?   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \*We might consider the high and low value as outliers, and get rid of them. On the other hand, as we will discover, income is best described via a lognormal distribution, and hence these values are in the extreme ends +- 3 sds from the mean. |
| 5 | **Stratify Variable – Household Income and plot the results:**  Stratify breaks that occur close to U.S. Guidelines for Poverty, Median Income, Wealth, and Rich (> $250k @ year)   1. Execute the following code (listed under comment heading “step 5” in the script file):   **breaks <- c(0, 23000, 52000, 82000, 250000, 999999) labels <- c("Poverty", "LowerMid", "UpperMid", "Wealthy", "Rich")  wealth <- cut(ds$income, breaks, labels) # add wealth as a column to ds ds <- cbind(ds, wealth) # show the 1st few lines. head(ds)**   1. Continue to execute the remaining part of the code in Step 5   **wt <- table(wealth) percent <- wt/sum(wt)\*100 wt <- rbind(wt, percent) wt plot(wt)**   1. Take another look at the relationship between wealth and income. Execute the following lines:   **# take another look -- wealth by rooms  nt <- table(wealth, ds$rooms) print(nt) plot(nt) # nice mosaic plot**   1. Execute this code from the script file. These lines will remove the variables wealth, breaks and labels, and then save the variables data set and write into a file named “Census.Rdata”.   **rm(wealth,breaks,labels) save(ds, wt, nt, file="Census.Rdata")** |
| 6 | **Plot Histogram and Distributions:**  Problem: How do you represent income given the range of values?   1. Select and execute the code under Step 6 Histograms and distributions in the script file.   **library(MASS)**  **with(ds, {  hist(income, main="Distribution of Household Income", freq=FALSE)  lines(density(income), lty=2, lwd=2)** # line type (lty) 2 is dashed **xvals = seq(from=min(income), to=max(income), length=100)  param = fitdistr(income, "lognormal")  lines(xvals, dlnorm(xvals, meanlog=param$estimate[1], sdlog=param$estimate[2]), col=”blue”)**  **})**   1. Now try the same thing with log10(income)   **logincome = log10(ds$income)**  **hist(logincome, main="Distribution of Household Income", freq=FALSE)** # line type lty(2) is a dashed line **lines(density(logincome), lty=2, lwd=2)  xvals = seq(from=min(logincome), to=max(logincome), length=100) param = fitdistr(logincome, "normal") lines(xvals, dnorm(xvals, param$estimate[1], param$estimate[2]), lwd=2, col=”blue”)** |
| 7 | **Compute Correlation between income and number of rooms:**  1. You need to consider your hypothesis.   * Your hypothesis is that the number of rooms in a house is predicted by household income (the rich can buy bigger houses), e.g. *lm(rooms ~ income)* * Therefore, our null hypothesis: no correlation between income and number of rooms. * Alternate hypothesis: there is a correlation between income and the number of rooms.  1. Execute the following code (listed after the comment line “Step7 in the script file).   **with(ds, cor(income, rooms))**  **with(ds, cor(log(income), rooms))) # this will give a better correlation**   1. For comparison, correlate rooms with a completely unrelated variable.   **n = length(ds$income) with(ds, cor(runif(n), rooms))** |
| 8 | **Create a Boxplot - Distribution of income as a factor of number of rooms:**   1. Select and execute the code (Listed after the comment line “Step 8”) in the script window. 2. Plot the distribution of income as a factor of # of rooms. ‘log=”y”’ plots income on log scale. We will suppress the outlier points and let the whiskers cover the full range of the data.   **boxplot(income ~ as.factor(rooms), data=ds, range=0, outline=F, log=”y”,** **xlab="# rooms", ylab="Income")**   1. Plot the # of rooms as a function of wealth level.   **boxplot(rooms ~ wealth, data = ds,** **main="Room by Wealth", Xlab="Category", ylab="# rooms")**  **# we’ll keep the outlier points in this one** |
| 9 | **Exit R:**   1. Type the following command into the RStudio command window:   **q()**   1. R will ask you if you want to save your workspace. Answer “**no**.” |

*End of Lab Exercise*

Part 2 – Graphics Package Plots and Hypothesis Tests

### Workflow Overview

### Lab Instructions

| **Step** | **Action** |
| --- | --- |
| 1 | **Define problem - Analysis of Variance (ANOVA):**  Suppose we are evaluating our marketing department’s incentive campaign that is trying to increase the amount of money that customers spend when they visit our online site. We ran a short experiment, where visitors to our site randomly received one of two incentive offers or got no offer at all. |
| 2 | **Generate the Data:**  **offers = sample(c("noffer", "offer1", "offer2"), size=500, replace=T)**  **purchasesize = ifelse(offers=="noffer", rlnorm(500, meanlog=log(25)), ifelse(offers=="offer1", rlnorm(500, meanlog=log(50)), rlnorm(500, meanlog=log(55))))**  **offertest = data.frame(offer=offers, purchase\_amt=purchasesize)** |
| 3 | **Examine the Data:**  **summary(offertest)**  The following command does the equivalent of the SQL command “SELECT avg(purchase\_amt) FROM offertest GROUP BY offer”,  **aggregate(x=offertest$purchase\_amt, by=list(offertest$offer), FUN="mean")** |
| 4 | **Plot and determine how purchase size varies within the three groups:**  1. The ‘log=”y”’ argument plots the y axis on the log scale. Does it appear that making offers increases purchase amount?  **boxplot(purchase\_amt ~ as.factor(offers), data=offertest, log="y")** |
| 5 | **Use lm() to do the ANOVA:**  1. Execute the following commands:  **model = lm(log10(purchase\_amt) ~ as.factor(offers), data=offertest)**  **summary(model)**  2. What is the p-value on the F-stat? Can we reject the null hypothesis? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  The intercept of the model is the mean value of log10(purchase\_amt | no offer), (call it m0) and the other coefficients are:  mean(log10(purchase\_amt |offer1)) – m0, and  mean(log10(purchase\_amt |offer2)) – m0, respectively.  3. What are the p-values on those coefficients? \_\_\_\_\_\_\_\_\_  4. Can we reject the null hypotheses that the mean purchase amount for offer1 was different from that of no offer, and similarly for offer2 vs. no offer? \_\_\_\_\_\_\_\_\_\_\_ |
| 6 | **Use Tukey’s test to check all the differences of means:**  1. Execute the following command:  **TukeyHSD(aov(model))**  1. Did offer1 and offer2 increase purchase size to different amounts (to the p<0.05 significance level)? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  2. What would you recommend to the marketing department, based on these results? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| 7 | **Use the lattice package for density plot:**  For this course, you are only expected to become familiar with the base graphics capabilities of R; however, there are other graphics packages available for R that makes certain kinds of visualizations easier to produce. If you continue to use R in the future, it will be helpful to be aware of these alternatives to base graphics.  The lattice package makes it easy to split data into different groups to highlight the differences between the groups. Here, we split the purchase\_amt data by offer, and plot the three offer-specific purchase\_amt densityplots on the same graph.  **library(lattice)**  **densityplot(~ purchase\_amt, group=offers, data=offertest, auto.key=T)** |
| 8 | **Plot the Logarithms of the Data:**  1. Because the data is so left-skewed, we may want to plot the logarithms of the data to more clearly see the differences in the distributions, and the different locations of the modes.  **densityplot(~ log10(purchase\_amt), group=offers, data=offertest, auto.key=T)**  2. Also try the plots:  **densityplot(~purchase\_amt | offers, data=offertest)**  **densityplot(~log10(purchase\_amt) | offers, data=offertest)**  3. Which style of graph do you find more helpful? |
| 10 | **Use ggplot() package:**    The ggplot2 package is based on a theory of the “algebra of graphs”. The syntax is rather complex, but ggplot excels at assembling rich composite graphs that use a variety of different graphic techniques. Here, we show how to produce a variation of the scatterplot + box-and-whisker plot that we saw earlier in the course, to plot the distributions of purchase amounts against offer.  1. Execute the following commands:  **library(ggplot2)**  **ggplot(data=offertest, aes(x=as.factor(offers), y=purchase\_amt)) + geom\_point(position="jitter", alpha=0.2) + geom\_boxplot(alpha=0.1, outlier.size=0) +**  **scale\_y\_log10()**  The function geom\_point() plots scatterplots. The function geom\_boxplot() plots box-and-whisker plots; outlier.size()=0 removes the outlier points beyond the whiskers that normally would be plotted. The function scale\_y\_log10() plots the y axis on a log10 scale.  2. You need to plot at least one geom\_xxx to get a graph. Try adding and removing the different terms of the graphing command to create simpler scatterplots or box-and-whisker plots, with and without log scaling.  3. Here’s how you would create the densityplots that you created in lattice. Execute the following commands.  **ggplot(data=offertest) + geom\_density(aes(x=purchase\_amt, colour=as.factor(offers)))**  **ggplot(data=offertest) + geom\_density(aes(x=purchase\_amt, colour=as.factor(offers))) + scale\_x\_log10()** |
| 11 | **Generate the example data to perform a Hypothesis Test with manual calculations:**  Hopefully, you won’t have to do this too often. Most statistical packages have functions that calculate a test statistic and evaluate it against the proper distribution, for the most common hypothesis tests. On occasion, you may need to calculate the p-values yourself. For our example, we will calculate the Student’s t-test for difference of means (unlike Welch’s test, Student’s t-test assumes identical variances), under the alternative hypothesis that the means are not equal.  1. Select and execute the following commands:  **x = rnorm(10) # distribution centered at 0**  **y = rnorm(10,2) # distribution centered at 2** |
| 12 | **Create a function to calculate the pooled variance, which is used in the Student’s t statistic:**  1. Select and execute the following commands. This will create a function named *pooled.var.*  **pooled.var = function(x, y) {**  **nx = length(x)**  **ny = length(y)**  **stdx = sd(x)**  **stdy = sd(y)**  **num = (nx-1)\*stdx^2 + (ny-1)\*stdy^2**  **denom = nx+ny-2 # degrees of freedom**  **(num/denom) \* (1/nx + 1/ny)**  **}** |
| 13 | **Examine the Data:**  Select and execute the following commands:  **mx = mean(x) my = mean(y)**  **mx - my**  **pooled.var(x,y)** |
| 14 | **Calculate the t statistic for Student's t-test:**  1. Select and execute the following commands:  tstat = (mx - my)/sqrt(pooled.var(x,y)) tstat |
| 15 | **Calculate the degrees of freedom:**  Under the null hypothesis, the t statistic is distributed in a Student’s distribution with nx+ny-2 degrees of freedom. Calculate the degrees of freedom for our problem.  Select and execute the following commands:  dof = length(x) + length(y) – 2  dof |
| 16 | **Compute the area under the curve:**  The function pt(x, dof) gives the area under the curve from *-Inf* to *x* for the Student's distribution with dof degrees of freedom. Since in this case we have a negative tstat, pt(tstat, dof) will give us the area under the left tail.  1. Select and execute the following commands:  tailarea = pt(tstat, dof)  2. Since our null hypothesis is that m1 <> m2, we need the area under both tails.  pvalue = 2\*tailarea  3. Are the means different (to the p<0.05 significance level)? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| 17 | **Perform Student’s t-test directly and compare the results:**  1.Execute the following command:  t.test(x, y, var.equal=T)  2. Does t.test() give the same results? \_\_\_\_\_\_\_\_\_\_\_\_ |

*End of Lab Exercise*