Analysis of Variance Report 1

Melinda Higgins

October 1, 2016



R Exercises Logo

## R Exercises - One-Way Analysis of Variance

The following exercise comes from [R Exercises at http://r-exercises.com/](http://r-exercises.com/), which is a site created by Research for Decisions, a Dutch research/consulting firm. They provide many exercises for learning R and practicing R coding skills. They also provide a large list of courses available for learning R - both free and paid - with detailed summaries of each option at <http://r-exercises.com/r-courses/>.

## Assumptions

For one way ANOVA results to be valid there are several assumptions that need to be satisfied:

1. The dependent variable is required to be continuous *(not ordinal, not continuous)*
2. The independent variable is required to be categorical with two or more categories *(these could also be ordinal levels being treated as non-sequential categories)*.
3. The dependent and independent variables have values for each row of data *(no missing data - any rows or subjects with missing data for the independent or dependent variables are removed before analysis)*.
4. Observations in each group are independent *(subjects cannot have data in more than 1 category - this cannot be used for dependent "repeated measures")*.
5. The dependent variable is approximately normally distributed in each group *(ideally at least symmetric, not skewed, no significant outliers)*.
6. There is approximate equality of variance in all the groups *(we'll see how to test for this below)*.
7. We should not have any *"significant"* outliers *(also tested below)*.

When the data shows non-normality, unequal variance or presence of outliers, the data can be "transformed" *(such as square-root or log transformations to help skewness)* or a non-parametric test like Kruskal-Wallis can be used. It is good to note Kruskal-Wallis does not require normality of data but still requires equal variance in your groups. *[Note: There are transformation alternatives for handling unequal variances which will not be covered in this exercise.]*

## Load Data

For this exercise, the data come from patients having stomach, colon, ovary, brochus, or breast cancer. The objective of the study was to identify if the number of days a patient survived was influenced by the organ affected. Our dependent variable is Survival measured in days. Our independent variable is Organ.

The data is available at <http://lib.stat.cmu.edu/DASL/Datafiles/CancerSurvival.html>. The dataset we're going to upload is the cancer-survival.csv file.

Since this dataset is a CSV (comma-delimited) formatted file, we can use the r function read.csv() to read in the dataset. This function returns a data.frame which we "assign" to the object cancer.survival.

## Look at data

Let's "look" at the data. We can use the head() function to look at the top 5-6 rows of the dataset.

head(cancer.survival)

## Survival Organ  
## 1 124 Stomach  
## 2 42 Stomach  
## 3 25 Stomach  
## 4 45 Stomach  
## 5 412 Stomach  
## 6 51 Stomach

We can clean up the looks for this output by using the kable() function from the knitr package to put the output into a table.

knitr::kable(head(cancer.survival))

|  |  |
| --- | --- |
| Survival | Organ |
| 124 | Stomach |
| 42 | Stomach |
| 25 | Stomach |
| 45 | Stomach |
| 412 | Stomach |
| 51 | Stomach |

## Select data and get stats

To select one of the columns in the dataset we can use the $ selector to "select" by the variable name. You can see the column names using the names() function.

names(cancer.survival)

## [1] "Survival" "Organ"

For example, we can find the mean survival time using the mean() function and selecting the 1st column which is Survival in the data frame cancer.survival using the followng code mean(cancer.survival$Survival).

mean(cancer.survival$Survival)

## [1] 558.625

We can also get summary statistics for the Survival column using the summary() function.

summary(cancer.survival$Survival)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 20.0 102.5 265.5 558.6 721.0 3808.0

Since the 1st column Survival is a continuous variable, the summary() function gives us the min, max, median, 1st and 3rd quartiles. Let's see what we get for the 2nd column, which is the list of Organs - it is a non-numeric text categorical variable.

summary(cancer.survival[,2])

## Breast Bronchus Colon Ovary Stomach   
## 11 17 17 6 13

We basically got the list of the Organs and the count (or frequency) of how many subjects survival times (i.e. number of data points) we have for each Organ. We can use the summary() command for the dataset as a whole and get the summary stats for both variables at the same time. If we use the knitr::kable() function around this summary, we get a nice summary stats table. Let's also add a caption.

knitr::kable(summary(cancer.survival),  
 caption = "Table of Summary Statistics for 2 Variables in Dataset")

Table of Summary Statistics for 2 Variables in Dataset

|  |  |  |
| --- | --- | --- |
|  | Survival | Organ |
|  | Min. : 20.0 | Breast :11 |
|  | 1st Qu.: 102.5 | Bronchus:17 |
|  | Median : 265.5 | Colon :17 |
|  | Mean : 558.6 | Ovary : 6 |
|  | 3rd Qu.: 721.0 | Stomach :13 |
|  | Max. :3808.0 | NA |

## Let's try some inline code

Suppose I want to write a sentence providing the mean and standard deviation of the survival times seen across this dataset. We can use the functions mean() and sd() and use them "inline" - see example below.

The average survival time was 558.625 days with a standard deviation of 776.4786713 days.

Another way to do this is to assign the mean and standard deviation to objects and then just call those objects. This simplifies the inline code commands and can help in cleaning up your formatting. We can also use the round() function with the digits= option to set the number of digits to 1 or 2. Let's create 2 objects one for the mean mnsurv and one for the standard deviation sdsurv and then use these to re-write the sentence. We'll also use the round() function and set the number of digits (after the decimal place) to 2.

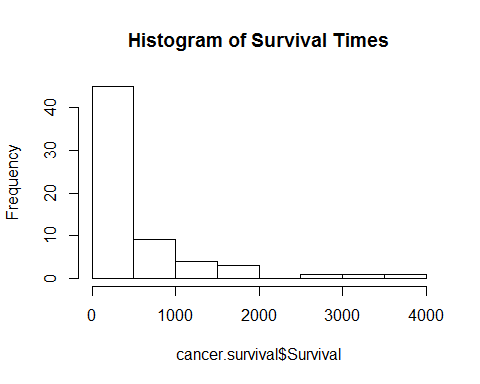
mnsurv <- mean(cancer.survival$Survival)  
sdsurv <- sd(cancer.survival$Survival)

The average survival time was 558.62 days with a standard deviation of 776.48 days.

## Let's make a histogram of the survival times.

Using the hist() function, let's make a histogram of the survival times overall and add a title.

hist(cancer.survival$Survival,  
 main="Histogram of Survival Times")

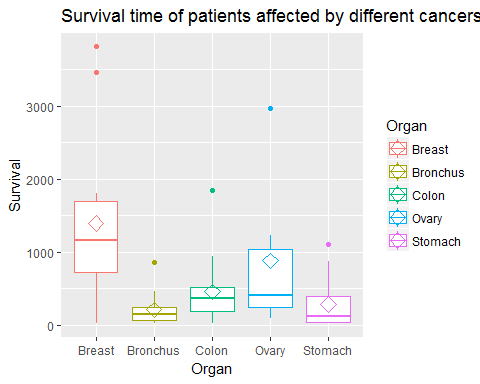


**What do you notice about the distribution?**

## Make side-by-side boxplots of survival times by organ

For this plot we are going to use the ggplot2 package and functions. So we have to install the package and load it.

# install.packages("ggplot2")  
library(ggplot2)  
ggplot(cancer.survival,  
 aes(x = Organ,y=Survival, color = Organ)) +  
 geom\_boxplot() +   
 stat\_summary(fun.y=mean,   
 geom="point",   
 shape=23,   
 size=4) +   
 ggtitle("Survival time of patients affected by different cancers")



## Summary Statistics for Survival by Organ

We can use the psych package and the describeBy() function to get summary statisics by group.

#describe the stats by group  
library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

describeBy(cancer.survival$Survival,cancer.survival$Organ)

## group: Breast  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 11 1395.91 1238.97 1166 1280.33 662.72 24 3808 3784 0.81  
## kurtosis se  
## X1 -0.7 373.56  
## --------------------------------------------------------   
## group: Bronchus  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 17 211.59 209.86 155 181.2 133.43 20 859 839 1.75 2.66  
## se  
## X1 50.9  
## --------------------------------------------------------   
## group: Colon  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 17 457.41 427.17 372 394.2 244.63 20 1843 1823 1.96  
## kurtosis se  
## X1 3.76 103.6  
## --------------------------------------------------------   
## group: Ovary  
## vars n mean sd median trimmed mad min max range skew  
## X1 1 6 884.33 1098.58 406 884.33 386.96 89 2970 2881 1.01  
## kurtosis se  
## X1 -0.75 448.49  
## --------------------------------------------------------   
## group: Stomach  
## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 13 286 346.31 124 234.64 121.57 25 1112 1087 1.27 0.25  
## se  
## X1 96.05

## Perform a One-way Analysis of Variance

We'll begin by checking assumptions.

### Check normality assumptions

For this we will use the shapiro.test() function along with tapply() and with().

with(cancer.survival,  
 tapply(Survival,Organ,shapiro.test))

## $Breast  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.86857, p-value = 0.07431  
##   
##   
## $Bronchus  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.76596, p-value = 0.0007186  
##   
##   
## $Colon  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.76056, p-value = 0.0006134  
##   
##   
## $Ovary  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.76688, p-value = 0.029  
##   
##   
## $Stomach  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.75473, p-value = 0.002075

### Check for equality of variance

To check the assumption of equal variances between the groups (Organ in this case), we'll use leveneTest() from the car package.

#Check for equality of variance  
library(car)

##   
## Attaching package: 'car'

## The following object is masked from 'package:psych':  
##   
## logit

leveneTest(Survival~Organ, data = cancer.survival)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)   
## group 4 4.4524 0.003271 \*\*  
## 59   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Apply a log transformation to survival time and check for normality and equality of variance.

cancer.survival$log.survival = log(cancer.survival$Survival)  
with(cancer.survival,tapply(log.survival,Organ,shapiro.test))

## $Breast  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.802, p-value = 0.009995  
##   
##   
## $Bronchus  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.98047, p-value = 0.9613  
##   
##   
## $Colon  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.92636, p-value = 0.1891  
##   
##   
## $Ovary  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.983, p-value = 0.9655  
##   
##   
## $Stomach  
##   
## Shapiro-Wilk normality test  
##   
## data: X[[i]]  
## W = 0.92837, p-value = 0.3245

### Check levene's test again

leveneTest(log.survival~Organ, data = cancer.survival)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)  
## group 4 0.6685 0.6164  
## 59

### Perform one way anova

aov1 = aov(log.survival~Organ,cancer.survival)  
saov1 <- summary(aov1)  
saov1

## Df Sum Sq Mean Sq F value Pr(>F)   
## Organ 4 24.49 6.122 4.286 0.00412 \*\*  
## Residuals 59 84.27 1.428   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Clean up ANOVA model fit summary

knitr::kable(saov1[[1]])

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| Organ | 4 | 24.48656 | 6.121639 | 4.285967 | 0.004122 |
| Residuals | 59 | 84.26959 | 1.428298 | NA | NA |

## Alternative Non-parametric One-way ANOVA

### Use a kruskal-wallis test

kruskal.test(log.survival~Organ,cancer.survival)

##   
## Kruskal-Wallis rank sum test  
##   
## data: log.survival by Organ  
## Kruskal-Wallis chi-squared = 14.954, df = 4, p-value = 0.004798