

One-Way Between-Groups ANOVA

A clinical study was conducted to assess the effect of three formulations of the same drug on reducing cholesterol. The formulations were 20mg at once (1time), 10mg twice a day (2times), and 5mg four times a day (4times). In addition, two competing drugs were used as a control group (drugD and drugE). Fifty patients were randomly selected for the study and 10 patients were randomly assigned to each treatment. The purpose of the study was to find which of the formulations, if any, were the most effective at reducing cholesterol and how these formulations compare with the existing drugs. We will do this by conducting a One-Way ANOVA F-Test to compare treatment means in a Completely Randomized Design.

Examining the Data

The following code loads the packages needed for the analysis.

```
library(multcomp)
library(tidyverse)
library(rstatix)
library(ggpubr)
library(gplots)
library(car)
```

The table below shows how some lesser known packages are used in the analysis.

Package	Usage in Project
<code>multcomp</code>	Our data comes from the dataset <i>cholesterol</i> in <code>multcomp</code> . This package also provides methods for multiple mean comparisons between treatments which will be used later in the analysis.
<code>rstatix</code>	<code>rstatix</code> provides a framework that allows us to use statistics with the <code>tidyverse</code> by providing pipeable statistical functions.
<code>ggpubr/gplots</code>	<code>ggpubr</code> and <code>gplots</code> allow us to visualize data easier instead of spending forever customizing <code>ggplot2</code> plots.
<code>car</code>	<code>car</code> has functions which are useful for detecting outliers in our data.

Our data comes from the dataset *cholesterol* in `multcomp`. The following code converts the dataset to a `tibble`, which makes it easier to use the `tidyverse` with our data.

```
cholesterol <- tibble(cholesterol)
```

The `head` function is used to get an idea of how the data is stored. The levels of the factor `trt` represent the five different drug formulations which are believed to have an effect on our response variable (cholesterol reduction).

```
head(cholesterol)
```

```
## # A tibble: 6 x 2
##   trt    response
##   <fct>    <dbl>
## 1 1time     3.86
## 2 1time    10.4
## 3 1time     5.91
## 4 1time     3.06
## 5 1time     7.72
## 6 1time     2.71
```

To examine the treatments of the experiment, we use the function `levels` on our factor `trt`. The treatments appear to be in order and spelled correctly as stated in the introduction.

```
levels(cholesterol$trt)
```

```
## [1] "1time" "2times" "4times" "drugD" "drugE"
```

The function `sample_n_by` is used to return two random rows from each treatment. This gives us a better overview of data and all of the treatments found by the `levels` function are included.

```
cholesterol %>%
  sample_n_by(trt, size = 2)
```

```
## # A tibble: 10 x 2
##   trt    response
##   <fct>    <dbl>
## 1 1time     4.92
## 2 1time     2.71
## 3 2times     3.51
## 4 2times     7.77
## 5 4times    14.0
## 6 4times    10.4
## 7 drugD    10.9
## 8 drugD    17.6
## 9 drugE    17.3
## 10 drugE    21.5
```

We use the `table` function to find the sample size for each treatment. In general, when sample sizes are unequal across treatments, ANOVA becomes more complicated.

Having unequal sample sizes can affect the homogeneity of variance assumption which will result in a lower statistical power for the test. Looking below, we see that our experiment has a balanced design with 10 experimental units for each treatment.

```
table(cholesterol$trt)
```

```
##
## 1time 2times 4times drugD drugE
##    10    10    10    10    10
```

The function `summary` is used to view the distribution of our response variable.

```
summary(cholesterol$response)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      2.304   8.409  12.605  12.738  17.519  27.244
```

The following code computes the mean, standard deviation, and sample size for each treatment.

```
group_stats <- cholesterol %>%
  group_by(trt) %>%
  summarise(n=n(), group_mean = mean(response), group_sd = sd(response))

group_stats
```

```
## # A tibble: 5 x 4
##   trt      n group_mean group_sd
##   <fct> <int>      <dbl>      <dbl>
## 1 1time     10       5.78       2.88
## 2 2times     10       9.22       3.48
## 3 4times     10      12.4       2.92
## 4 drugD      10      15.4       3.45
## 5 drugE      10      20.9       3.35
```

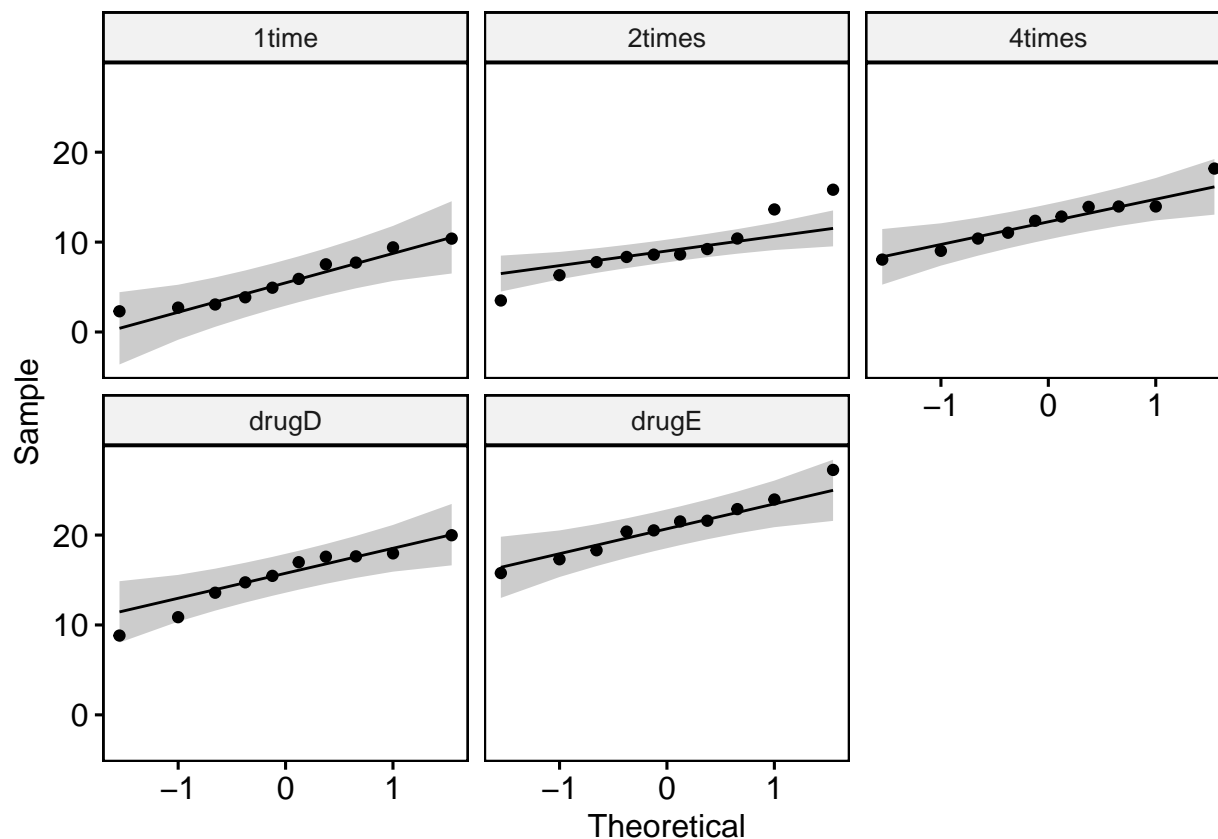
Conditions Required for a Valid Anova F-Test: Completely Randomized Design

1. The samples are randomly selected in an independent manner from the treatment populations.
2. All k sampled populations have distribution that are approximately normal.
3. The k population variances are equal.

A patient is the experimental unit for this experiment. For a Completely Randomized Design, independent random samples of experimental units are selected for each treatment or k treatments can be randomly assigned to a random sample of experimental units. Our experiment has the latter, since 50 patients were randomly selected and then randomly assigned to treatments. We will consider the first assumption to be fulfilled.

For the second assumption, we can either check the normality of the residuals of our ANOVA model or we can check the normality of each treatment population. The following code uses the function `ggqqplot` function to plot a Q-Q plot of the response variable for each treatment. Looking at the plots, we can see that the k populations are normally distributed.

```
ggqqplot(cholesterol, "response", facet.by = "trt")
```



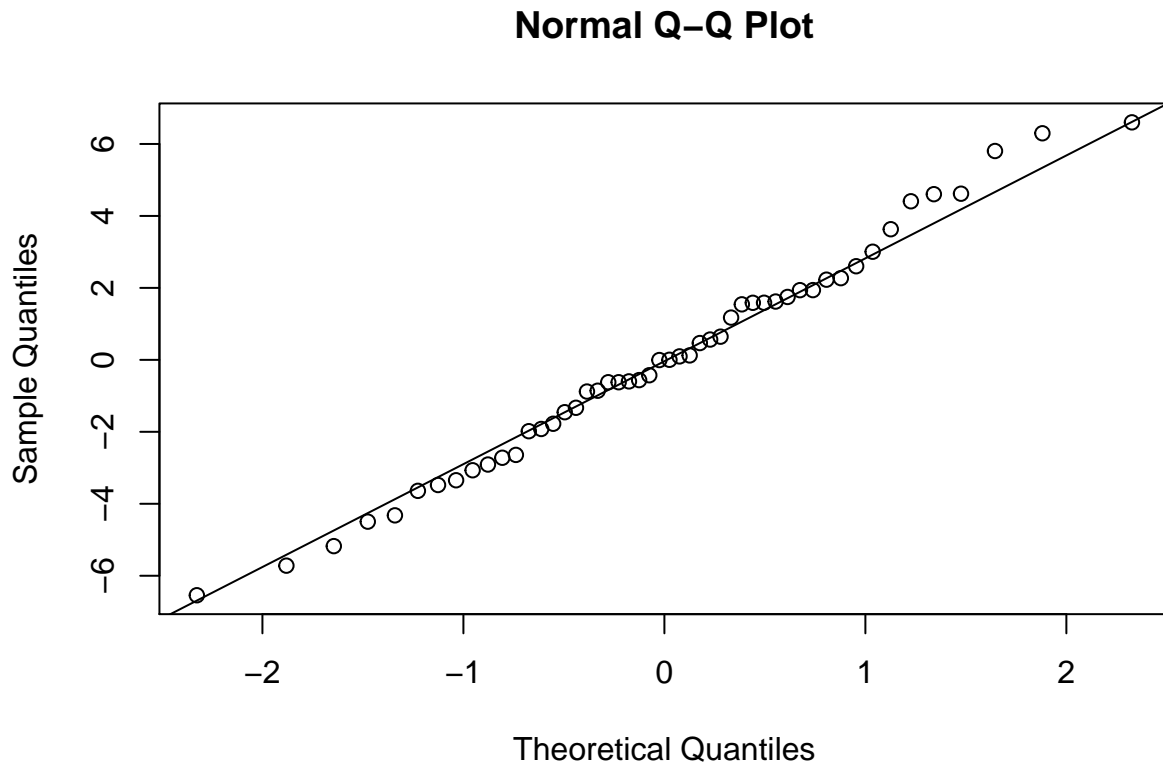
Recall that Anova and Regression are both special cases of the general linear model, which is why we can use the same `lm` function for regression to find the residuals for ANOVA. We can then conduct the Shapiro-Wilk test on the model's residuals to test for normality. Since the p-value is larger than our significance level(0.05), we do not reject the null hypothesis which means our k treatment populations are most likely normal.

```
model <- lm(response ~ trt,data = cholesterol)
shapiro.test(model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model$residuals
## W = 0.98864, p-value = 0.9094
```

The following code make a Q-Q plot of the above model's residuals.

```
qqnorm(model$residuals)
qqline(model$residuals)
```



The following code computes the Shapiro-Wilk test for each treatment. Since we have large p-values, we have enough evidence to prove that the treatment populations are normal.

```
cholesterol %>%
  group_by(trt) %>%
  summarize(W=shapiro.test(response)$statistic, p_value=shapiro.test(response)$p.value)
```

```
## # A tibble: 5 x 3
##   trt      W p_value
##   <fct> <dbl> <dbl>
## 1 1time 0.931 0.454
## 2 2times 0.943 0.589
## 3 4times 0.955 0.726
## 4 drugD 0.934 0.489
## 5 drugE 0.981 0.970
```

The following code does the same thing as the code above but it is a lot cleaner. The `shapiro_test` function is from the package `rstatix` which allows us to use the pipe with statistical tests.

```
cholesterol %>%
  group_by(trt) %>%
  shapiro_test(response)
```

```
## # A tibble: 5 x 4
##   trt      variable statistic      p
```

```
##   <fct>  <chr>          <dbl> <dbl>
## 1 1time  response      0.931 0.454
## 2 2times response      0.943 0.589
## 3 4times response      0.955 0.726
## 4 drugD  response      0.934 0.489
## 5 drugE  response      0.981 0.970
```

Looking at all the code above, we will consider the second assumption to be fulfilled. For the third assumptions we can use Bartlett's Test of Homogeneity of Variances to find out if the k population variances are equal. We are using Bartlett's test here instead of Levene's test because Bartlett's test performs better when we know that our data is normal. Since the p-value is large for Bartlett's test, the variances among treatments do not differ significantly.

```
bartlett.test(response~trt,data = cholesterol)
```

```
##
## Bartlett test of homogeneity of variances
##
## data: response by trt
## Bartlett's K-squared = 0.57975, df = 4, p-value = 0.9653
```

Outliers inflate the MST and MSE, which leads to a smaller F Statistic. This means that the chance of rejecting the null hypothesis is also lowered. It's important to make sure there are no data collection errors and to see the effect of outliers on our analysis. Using the `identify_outliers` function allows us to identify outliers using boxplot methods. Looking below, we do not have any extreme outliers so we will keep the data the same.

```
cholesterol %>%
  group_by(trt) %>%
  identify_outliers(response)
```

```
## # A tibble: 3 x 4
##   trt    response is.outlier is.extreme
##   <fct>    <dbl> <lgl>      <lgl>
## 1 2times    13.6 TRUE       FALSE
## 2 2times     3.51 TRUE       FALSE
## 3 2times    15.8 TRUE       FALSE
```

Hypotheses for ANOVA F-Test to Compare k Treatment Means

We will denote the population means of k treatments as $\mu_1, \mu_2, \mu_3, \dots$ and so on. We will test the null hypothesis that treatment means are equal against the alternative hypothesis that at least two treatment means differ. We will use a level of significance of .05 for this test.

Ho: $\mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$

Ha: At least two of these treatment means differ.

Conducting the Test

To conduct a one-way ANOVA test, we use the `aov` function which requires a formula and a dataset. The formula below tells R to conduct the ANOVA test where response is the response variable and trt is the factor. We use the `summary` function on `fit` to summarize the results from the ANOVA tests.

Looking at the results, we see that our F test statistic is quite big and our p-value is small. Recall that the F statistic is the average variation between treatment means over the average variation within treatments. The bigger the F statistic is, the more likely there is statistically significant difference between means. Since the p-value is smaller than our alpha and the F statistic is quite large, we can reject the null hypotheses and we have enough evidence that to prove that the treatment means are not equal.

```
fit <- aov(response ~ trt,data=cholesterol)
summary(fit)
```

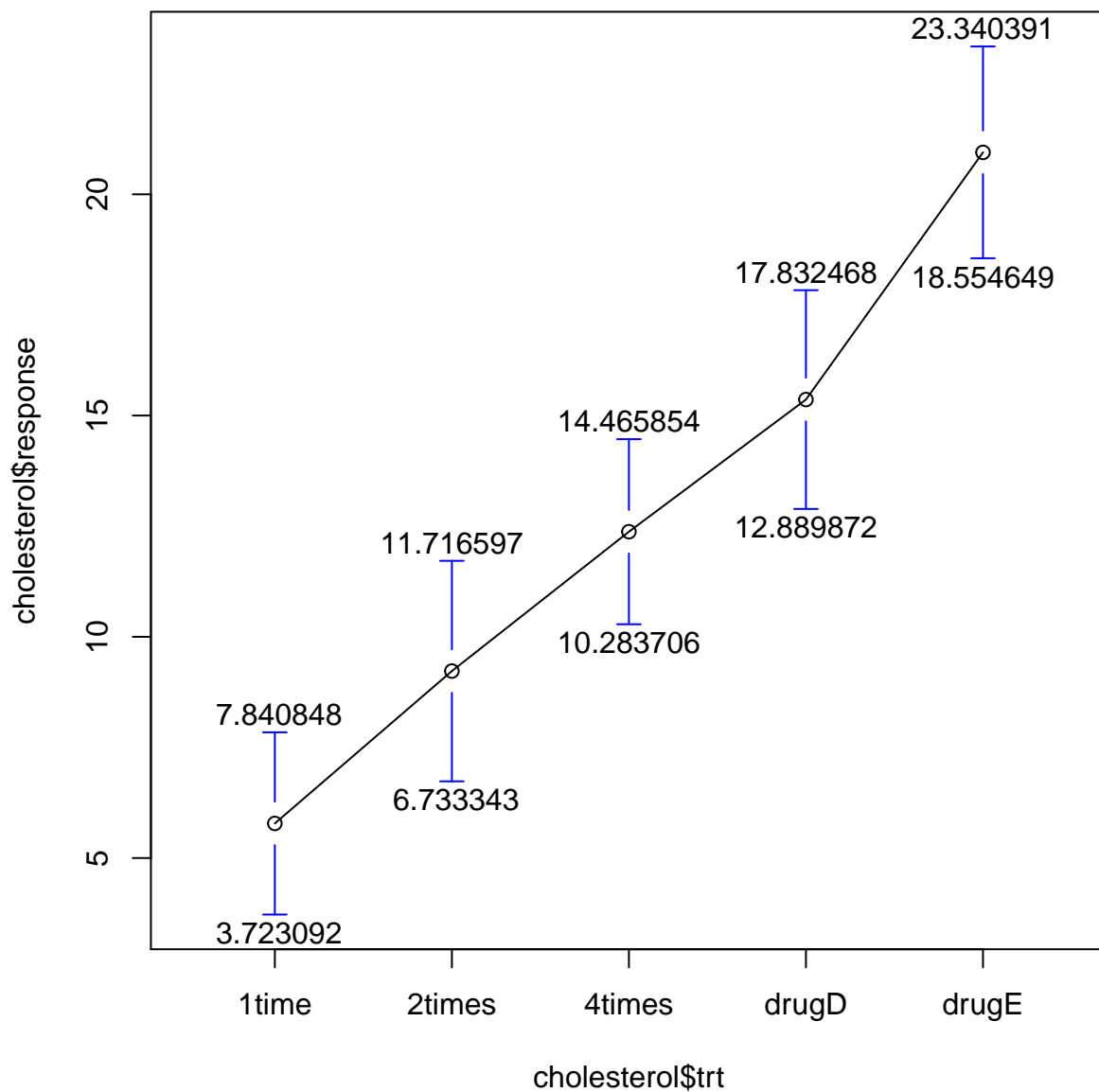
```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## trt           4 1351.4    337.8    32.43 9.82e-13 ***
## Residuals    45  468.8     10.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since we rejected the null hypothesis, we know that at least two treatment means differ but we are not sure which treatments differ significantly between each other. To make multiple comparisons of treatment means, we will use Tukey's method which works best with equal sample sizes and pairwise comparisons.

Before we use Tukey's Method, recall that two independent sample t-tests are equivalent to two independent sample f-tests. Because of this, we can construct a one sample 95% confidence interval for each treatment. If two treatment's confidence intervals intersect, they are not significantly different.

The `plotmeans` function plots the mean and ci for each treatment. From this plot, we see that 4times is the best drug formulation and is significantly different from 1time. We also see that drugE is statistically better different than the other treatments. We also see that 2times is not significantly different from 1times or 4times. Finally we see that DrugD is statistically different from 2times , 1time, and drugE but not statistically different from 4times.

```
plotmeans(cholesterol$response~cholesterol$trt, ci.label = TRUE,n.label = FALSE)
```



The following code shows the confidence intervals above are two-sided t-test confidence intervals. The two-sided t-test confidence interval for the treatment 1time is the same as the one above.

```
g <- filter(cholesterol, trt == "1time")
t.test(x = g$response, conf.level = .95, alternative = "two.sided")
```

```
##
## One Sample t-test
##
## data:  g$response
## t = 6.3528, df = 9, p-value = 0.0001324
```



```
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 3.723092 7.840848
## sample estimates:
## mean of x
## 5.78197
```

Why don't we just conduct multiple t-tests to find the confidence intervals between groups? The problem with this and the approach above is that it increases the likelihood of a Type 1 error. Using Tukey's method, we can say that collectively 95% of the intervals include the true difference between groups. This is because Tukey's method adjusts for the amount of groups being compared unlike the t-test.

Looking below, any p-value below our significance level means those two groups are significant different. A 95% confidence interval is also provided which contains the true difference between those groups. For example, we that drugE-drugD has a low p-value meaning they are significantly different. Looking at the ci, we see that the true mean of drugE is (1.48,9.68) higher than drugD.

TukeyHSD(fit)

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = response ~ trt, data = cholesterol)
##
## $trt
##          diff          lwr          upr          p adj
## 2times-1time 3.44300 -0.6582817 7.544282 0.1380949
## 4times-1time 6.59281 2.4915283 10.694092 0.0003542
## drugD-1time 9.57920 5.4779183 13.680482 0.0000003
## drugE-1time 15.16555 11.0642683 19.266832 0.0000000
## 4times-2times 3.14981 -0.9514717 7.251092 0.2050382
## drugD-2times 6.13620 2.0349183 10.237482 0.0009611
## drugE-2times 11.72255 7.6212683 15.823832 0.0000000
## drugD-4times 2.98639 -1.1148917 7.087672 0.2512446
## drugE-4times 8.57274 4.4714583 12.674022 0.0000037
## drugE-drugD 5.58635 1.4850683 9.687632 0.0030633
```

The following code shows another way to compute tukey's method for comparing treatments.

```
pwc <- cholesterol %>% tukey_hsd(response~trt)
pwc
```

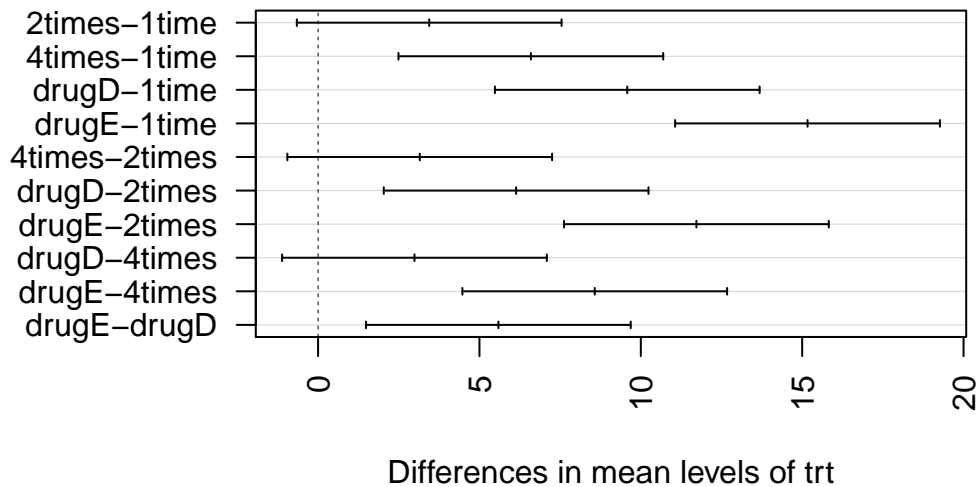
```
## # A tibble: 10 x 8
##   term  group1 group2 estimate conf.low conf.high  p.adj p.adj.signif
## * <chr> <chr> <chr>    <dbl>    <dbl>    <dbl>    <dbl> <chr>
## 1 trt    1time  2times     3.44   -0.658     7.54 1.38e- 1 ns
## 2 trt    1time  4times     6.59     2.49    10.7 3.54e- 4 ***
## 3 trt    1time  drugD      9.58     5.48    13.7 3.46e- 7 ****
## 4 trt    1time  drugE     15.2    11.1    19.3 1.35e-12 ****
## 5 trt    2times  4times     3.15   -0.951     7.25 2.05e- 1 ns
## 6 trt    2times  drugD      6.14     2.03    10.2 9.61e- 4 ***
## 7 trt    2times  drugE     11.7     7.62    15.8 2.27e- 9 ****
## 8 trt    4times  drugD      2.99    -1.11     7.09 2.51e- 1 ns
```

```
## 9 trt 4times drugE 8.57 4.47 12.7 3.72e- 6 ****
## 10 trt drugD drugE 5.59 1.49 9.69 3.06e- 3 **
```

The following code visualizes the confidence intervals of differences between group means. If a confidence interval includes 0, those groups are not statistically different from each other.

```
par(las=2,mar = c(7,7,7,7))
plot(TukeyHSD(fit))
```

95% family-wise confidence level



Finally, we use the following code which gives us the boxplots for each treatment along with Tukey's compact letter ranking. If two boxplots have the same letter, they are not statistically different. Overall, 4times is the best formulation for a certain drug for reducing cholesterol but falls short when compared to the drug E.

```
par(las=1,mar=c(6,6,6,6))
tuk <- glht(fit,linfct=mcp(trt="Tukey"))
plot(cld(tuk,level = .05),col = "lightblue")
```

