C2M2: Peer Reviewed Assignment

Outline:

The objectives for this assignment:

1. Utilize contrasts to see how different pairwise comparison tests can be conducted.
2. Understand power and why it's important to statistical conclusions.
3. Understand the different kinds of post-hoc tests and when they should be used.

General tips:

1. Read the questions carefully to understand what is being asked.
2. This work will be reviewed by another human, so make sure that you are clear and concise in what your explanations and answers.

Problem 1: Contrasts and Coupons

Consider a hardness testing machine that presses a rod with a pointed tip into a metal specimen with a known force. By measuring the depth of the depression caused by the tip, the hardness of the specimen is determined.

Suppose we wish to determine whether or not four different tips produce different readings on a hardness testing machine. The experimenter has decided to obtain four observations on Rockwell C-scale hardness for each tip. There is only one factor - tip type - and a completely randomized single-factor design would consist of randomly assigning each one of the 4×4=16 runs to an experimental unit, that is, a metal coupon, and observing the hardness reading that results. Thus, 16 different metal test coupons would be required in this experiment, one for each run in the design.

In [2]:
```
tip    <- factor(rep(1:4, each = 4))
coupon <- factor(rep(1:4, times = 4))
y <- c(9.3, 9.4, 9.6, 10,
       9.4, 9.3, 9.8, 9.9,
       9.2, 9.4, 9.5, 9.7,
       9.7, 9.6, 10, 10.2)
hardness <- data.frame(y, tip, coupon)
hardness
```

A data.frame: 16 × 3

y	tip	coupon
<dbl>	<fct>	<fct>
9.3	1	1
9.4	1	2

Processing math: 100%

9.6	1	3
10.0	1	4
9.4	2	1
9.3	2	2
9.8	2	3
9.9	2	4
9.2	3	1
9.4	3	2
9.5	3	3
9.7	3	4
9.7	4	1
9.6	4	2
10.0	4	3
10.2	4	4

1. (a) Visualize the Groups

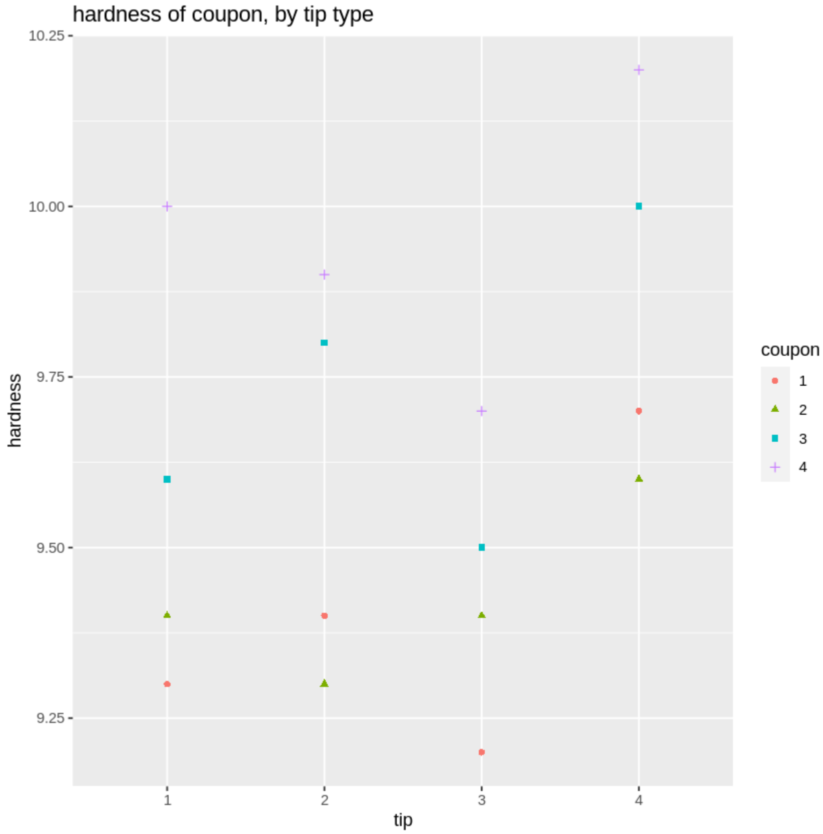
Before we start throwing math at anything, let's visualize our data to get an idea of what to expect from the eventual results.

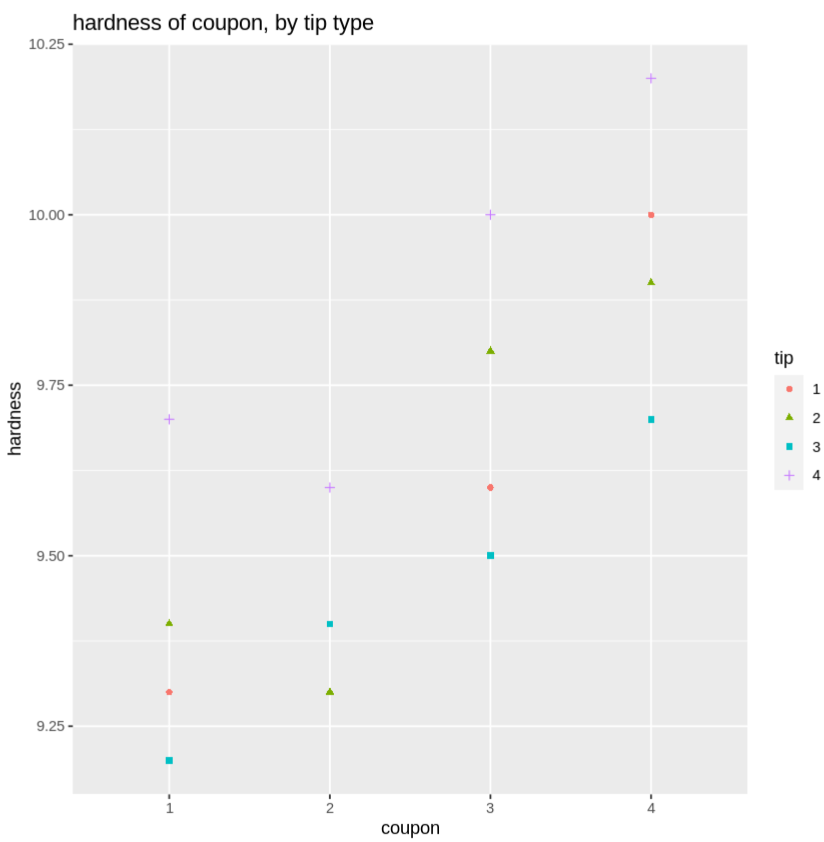
Construct interaction plots for `tip` and `coupon` using ggplot(). Be sure to explain what you can from the plots.

In [8]:

```
# Your Code Here
library(ggplot2)
p1 = ggplot(data = hardness) +
        geom_point(aes(x = tip, y = y, shape = coupon, color=coupon)) +
        xlab('tip') + ylab('hardness') +
        ggtitle('hardness of coupon, by tip type')
p1

p2 = ggplot(data = hardness) +
        geom_point(aes(x = coupon, y = y, shape = tip, color=tip)) +
        xlab('coupon') + ylab('hardness') +
        ggtitle('hardness of coupon, by tip type')
p2
```





Solution

The first plot shows that, for tip type 1 and 3, coupons show increasing hardness starting from coupon 1, up to 4.

For tip 2, coupon 1 is harder than coupon 2, with 3 and 4 unchanged.

In the second plot we can see that there is a slight dip in hardness for coupon 2 for all tip types. Coupon 1 has the second lowest overall hardness value against all tips, increasing as we go to coupon 3 and 4.

In general, there are no noticeable interactions.

1. (b) Interactions

Should we test for interactions between `tip` and `coupon`? Maybe there is an interaction between the different metals that goes beyond our current scientific understanding!

Fit a linear model to the data with predictors `tip` and `coupon`, and an interaction between the two. Display the summary and explain why (or why not) an interaction term makes sense for this data.

In [4]:

```
# Your Code Here
library(tidyverse)
hardness = hardness %>%
    mutate(tip = as.factor(tip))

mod1 = lm(y ~ coupon + tip + coupon:tip, data = hardness)
mod1
summary(mod1)

mod2 = lm(y ~ coupon + tip, data = hardness)
mod2
summary(mod2)
```

```
── Attaching packages ─────────────────────────────────────── tidyverse 1
.3.0 ──

   tibble  3.0.1      dplyr   0.8.5
   tidyr   1.0.2      stringr 1.4.0
   readr   1.3.1      forcats 0.5.0
   purrr   0.3.4

── Conflicts ──────────────────────────────────────────── tidyverse_conflicts
() ──
   dplyr::filter() masks stats::filter()
   dplyr::lag()    masks stats::lag()
```

```
Call:
lm(formula = y ~ coupon + tip + coupon:tip, data = hardness)

Coefficients:
 (Intercept)       coupon2       coupon3       coupon4          tip2
   9.300e+00     1.000e-01     3.000e-01     7.000e-01     1.000e-01
        tip3          tip4   coupon2:tip2  coupon3:tip2  coupon4:tip2
  -1.000e-01     4.000e-01    -2.000e-01     1.000e-01    -2.000e-01
coupon2:tip3  coupon3:tip3  coupon4:tip3  coupon2:tip4  coupon3:tip4
   1.000e-01    -3.689e-15    -2.000e-01    -2.000e-01    -3.784e-15
coupon4:tip4
  -2.000e-01
```

```
Call:
lm(formula = y ~ coupon + tip + coupon:tip, data = hardness)

Residuals:
ALL 16 residuals are 0: no residual degrees of freedom!

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.300e+00         NA      NA       NA
coupon2       1.000e-01         NA      NA       NA
coupon3       3.000e-01         NA      NA       NA
coupon4       7.000e-01         NA      NA       NA
tip2          1.000e-01         NA      NA       NA
tip3         -1.000e-01         NA      NA       NA
tip4          4.000e-01         NA      NA       NA
coupon2:tip2 -2.000e-01         NA      NA       NA
coupon3:tip2  1.000e-01         NA      NA       NA
coupon4:tip2 -2.000e-01         NA      NA       NA
coupon2:tip3  1.000e-01         NA      NA       NA
coupon3:tip3 -3.689e-15         NA      NA       NA
coupon4:tip3 -2.000e-01         NA      NA       NA
coupon2:tip4 -2.000e-01         NA      NA       NA
coupon3:tip4 -3.784e-15         NA      NA       NA
coupon4:tip4 -2.000e-01         NA      NA       NA

Residual standard error: NaN on 0 degrees of freedom
Multiple R-squared:      1,     Adjusted R-squared:    NaN
F-statistic:   NaN on 15 and 0 DF,  p-value: NA

Call:
lm(formula = y ~ coupon + tip, data = hardness)

Coefficients:
(Intercept)      coupon2      coupon3      coupon4         tip2         ti
p3
      9.350        0.025        0.325        0.550        0.025       -0.1
25
       tip4
      0.300

Call:
lm(formula = y ~ coupon + tip, data = hardness)

Residuals:
     Min       1Q   Median       3Q      Max
-0.10000 -0.05625 -0.01250  0.03125  0.15000

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.35000    0.06236 149.934  < 2e-16 ***
coupon2      0.02500    0.06667   0.375 0.716345
coupon3      0.32500    0.06667   4.875 0.000877 ***
coupon4      0.55000    0.06667   8.250 1.73e-05 ***
tip2         0.02500    0.06667   0.375 0.716345
tip3        -0.12500    0.06667  -1.875 0.093550 .
tip4         0.30000    0.06667   4.500 0.001489 **
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09428 on 9 degrees of freedom
Multiple R-squared:  0.938,     Adjusted R-squared:  0.8966
F-statistic: 22.69 on 6 and 9 DF,  p-value: 5.933e-05
```

Solution
As pointed out in the first summary with the interactions, there are no residual degrees of freedom, meaning that all the data were used to compute the parameters. We would need additional data to compute other statistical information. Therefore, it doesn't make sense to include interaction terms.

1. (c) Contrasts

Let's take a look at the use of contrasts. Recall that a contrast takes the form
$t\sum i=1 c_i \mu_i = 0$,

where $c = (c_1, \ldots, c_t)$ is a constant vector and $\mu = (\mu_1, \ldots, \mu_t)$ is a parameter vector (e.g., μ_1 is the mean of the i^{th} group).

We can note that $c = (1, -1, 0, 0)$ corresponds to the null hypothesis $H_0: \mu_2 - \mu_1 = 0$, where μ_1 is the mean associated with tip1 and μ_2 is the mean associated with tip2. The code below tests this hypothesis.

Repeat this test for the hypothesis $H_0: \mu_4 - \mu_3 = 0$. Interpret the results. What are your conclusions?

In [11]:
```
library(multcomp)
lmod = lm(y~tip+coupon, data=hardness)
fit.gh2 = glht(lmod, linfct = mcp(tip = c(1,-1,0,0)))
summary(fit.gh2)
#estimate of mu_2 - mu_1
with(hardness, sum(y[tip == 2])/length(y[tip == 2]) -
     sum(y[tip == 1])/length(y[tip == 1]))


lmod2 = lm(y~tip+coupon, data=hardness)
fit.gh3= glht(lmod2, linfct = mcp(tip = c(0,0,1,-1)))
#estimate of mu_4 - mu_3
with(hardness, sum(y[tip == 4])/length(y[tip == 4]) -
     sum(y[tip == 3])/length(y[tip == 3]))

summary(fit.gh3)
```

```
         Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: User-defined Contrasts


Fit: lm(formula = y ~ tip + coupon, data = hardness)

Linear Hypotheses:
       Estimate Std. Error t value Pr(>|t|)
1 == 0 -0.02500    0.06667  -0.375    0.716
```

```
(Adjusted p values reported -- single-step method)
```

0.0250000000000021

0.425000000000001

```
         Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: User-defined Contrasts


Fit: lm(formula = y ~ tip + coupon, data = hardness)

Linear Hypotheses:
       Estimate Std. Error t value Pr(>|t|)    
1 == 0 -0.42500    0.06667  -6.375 0.000129 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- single-step method)
```

Solution
The p-value of 0.000129 leads to the conclusion that the null hypothesis can be rejected and there is a statistically significant difference between the parameters, μ_4 and μ_3.

1. (d) All Pairwise Comparisons

What if we want to test all possible pairwise comparisons between treatments. This can be done by setting the treatment factor (`tip`) to "Tukey". Notice that the p-values are adjusted (because we are conducting multiple hypotheses!).

Perform all possible Tukey Pairwise tests. What are your conclusions?

In [12]:
```
# Your Code Here
lmod = lm(y ~ tip + coupon, data = hardness)
fit.gh = glht(lmod, linfct = mcp(tip = "Tukey"))
summary(fit.gh)
```

```
         Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts


Fit: lm(formula = y ~ tip + coupon, data = hardness)

Linear Hypotheses:
           Estimate Std. Error t value Pr(>|t|)    
2 - 1 == 0  0.02500    0.06667   0.375  0.98088    
3 - 1 == 0 -0.12500    0.06667  -1.875  0.30279    
4 - 1 == 0  0.30000    0.06667   4.500  0.00664 ** 
3 - 2 == 0 -0.15000    0.06667  -2.250  0.18150    
4 - 2 == 0  0.27500    0.06667   4.125  0.01142 *  
4 - 3 == 0  0.42500    0.06667   6.375  < 0.001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Adjusted p values reported -- single-step method)
```

Solution
The p-values from the above summary show that there is a statistically significant difference between tip pairs 4-1, 4-2, and 4-3, assuming an alpha level of 0.05.

Problem 2: Ethics in my Math Class!

In your own words, answer the following questions:

- What is power, in the statistical context?
- Why is power important?
- What are potential consequences of ignoring/not including power calculations in statistical analyses?

Solution

- Power is the probability of Type II error described by statisticians. It is the probability of rejecting the null hypothesis when it is false.
- Power is important because it impacts critical statistical decisions such as correctly or incorrectly rejecting the test under some hypothesis.
- Potential consequences of ingnoring power calculations are more frequent type I and type II errors and poor/inappropriate sample size in conducting tests.

Problem 3: Post-Hoc Tests

There's so many different post-hoc tests! Let's try to understand them better. Answer the following questsions in the markdown cell:

- Why are there multiple post-hoc tests?
- When would we choose to use Tukey's Method over the Bonferroni correction, and vice versa?
- Do some outside research on other post-hoc tests. Explain what the method is and when it would be used.

Solution

- There are multiple post-hoc tests because there can be more than one Type 1 error in a familiy of comparisons. Also, each post-hoc test uses different methods and software, so it can depend on the user's preference or software being used. Some tests are designed for specific situations, such as for conductinhg multiple statistical tests at the same time.
- One should choose to use Tukey's method when one wishes to examine all possible pairwise comparisons. When one needs more flexibility in applicability in terms of hypothesis tests, one should choose the Bonferroni correction. Also, Bonferroni correction is useful for small sets of planned comparisons while having to control the familywise Type 1 error rate.
- Some other post-hoc tests include Dunnett's correction, which is siimilar to Tukey's in the sense that it is used to compare means. The difference lies in comparing every mean to a control

mean. There is also Newman-Keuls which different critical values for comparing pairs of means. It's used hwen we wish to find significant differences.

```
In [ ]:
```