Pol Sci 630: Problem Set 3 - Comparisons and Inference - Solutions

Prepared by: Jan Vogler (jan.vogler@duke.edu)

Grading Due Date: Friday, September 18th, 12.15 PM (Beginning of Lab)

Insert your comments on the assignment that you are grading above the solution in bold and red text. For example write: "GRADER COMMENT: everything is correct! - 4/4 Points" Also briefly point out which, if any, problems were not solved correctly and what the mistake was.

Use the following scheme to assign points: For correctly solved problems, assign the point value that is stated in the problem (see original problem set for exact point values). For correctly solved bonus problems, add that value to the total score for a problem but do not go above 4 points per problem. If there are mistakes in any problem, subtract points according to the extent of the mistake. If you subtract points, explain why.

In order to make your text bold and red, you need to insert the following line at the beginning of the document:

```
\usepackage{color}
```

and the following lines above the solution of the specific task:

```
\textbf{\color{red} GRADER COMMENT: everything is correct! - 4/4 Points}
```

R Programming

Problem 1

```
### Problem 1:

### a

x = seq(1, 1000, by = 1)
y = 2 * x - 5

cov(x, y)

## [1] 166833.3
```

Interpretation: the covariance indicates that there might be a positive linear relationship of x and y. However, because the covariance is not to-scale, the number itself is not very meaningful.

Note: if someone created a different kind of linear function, the result might be a covariance that indicates a negative linear relationship.

```
cor(x, y)

## [1] 1
```

Interpretation: the correlation is bound between -1 and 1. The correlation value of 1 here means that x and y have a perfect positive linear relationship. As x goes above its mean, y goes above its mean. This is not surprising as we created y through a linear function of x.

Note: if someone created a different kind of linear function, the result might be a correlation value of -1, indicating a perfect negative linear relationship. As x goes above its mean, y goes below its mean.

```
noise = rnorm(1000, mean = 0, sd = 10)
y2 = y + noise

cov(x, y2)
```

```
## [1] 166740.4
```

Interpretation: same as above - the covariance indicates that there might be a linear positive relationship of x and y. However, because the covariance is not to-scale, the number itself is not very meaningful.

```
cor(x, y2)

## [1] 0.9998636
```

The correlation is bound between -1 and 1. The correlation value here should be very close to 1 or -1 but not be exactly that value due to the random error. As long as some random error has been introduced to a formerly perfect linear relationship, even if that relationship is still generally linear, there will be a reduction in the absolute value of the correlation. Accordingly, the result you can expect to get here is an absolute value of approximately 0.99. The exact value, however, depends on the size of the random error that you introduced. If you introduce a random error that has a greater variance, then the value of your correlation will go down further.

Interpretation: a correlation close to 1 or -1 indicates a nearly perfect linear relationship of two variables. If the relationship is positive, the following is true: as x goes above its mean, y goes above its mean. If the relationship is negative, the following is true: as x goes above its mean, y goes below its mean. In most cases the randomly distributed error will not change the generally strong linear relationship between the two variables.

```
### b

correlation = function(v1, v2) {
    numerator = sum((v1 - mean(v1)) * (v2 - mean(v2)))/(length(a) - 1)
    denominator = sd(v1) * sd(v2)
    print(numerator/denominator)
}

### Let's try this function.

a = seq(1, 10, by = 1)
```

```r
noise2 = rnorm(10, mean = 0, sd = 1)
b = a + noise2

cor(a, b)
```

```
## [1] 0.9629126
```

```r
correlation(a, b)
```

```
## [1] 0.9629126
```

```r
# These two return the same result, meaning that we did it correctly.

### Bonus problem:

correlation2 = function(v1, v2) {
    if (length(v1) == length(v2)) {
        if (is.numeric(v1) & is.numeric(v2)) {
            numerator = sum((v1 - mean(v1)) * (v2 - mean(v2)))/(length(a) -
                1)
            denominator = sd(v1) * sd(v2)
            print(numerator/denominator)
        } else {
            print("The two vectors need to be numeric.")
        }
    } else {
        print("The two vectors need to be of the same length.")
    }
}

### Let's plug in vectors that do not work.

c = seq(1, 11)
length(c)
```

```
## [1] 11
```

```r
correlation2(a, c)

## [1] "The two vectors need to be of the same length."

# Returns the correct error message.

d = c("a", "b", "c", "d", "e", "f", "g", "h", "i", "j")
is.numeric(d)

## [1] FALSE

correlation2(a, d)

## [1] "The two vectors need to be numeric."

# Returns the correct error message.
```

Problem 2

```r
### a

set.seed(2)
vec1 = rpois(50, lambda = 10)
set.seed(2)
vec2 = rpois(50, lambda = 12)

t.test(vec1, vec2)

##
##  Welch Two Sample t-test
##
## data:  vec1 and vec2
## t = -3.2796, df = 94.967, p-value = 0.001454
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -3.3711865 -0.8288135
## sample estimates:
## mean of x mean of y
##       9.5      11.6
```

b) The mean of the first vector is 9.5, the mean of the second vector is 11.6. Given the number of observations – 50 in each vector – the t.test function returns a t-value of -3.2796. The 95-percent confidence interval shows us that we are 95 percent confident that the first distribution has a mean that is lower by a number between -3.3711865 and -0.8288135. The fact that 0 is not included in this interval implies a p-value of $p < 0.05$. Indeed, given 98 degrees of freedom, the t-value is associated with a p-value of $p = 0.001454$. This p-value implies that the probability of obtaining this result or a result with more extreme evidence against H_0 if vector 1 and vector 2 were from a distribution with the same mean is approximately 0.15 percent ($= 0.0015$). This is very strong evidence against the null hypothesis (H_0), which is that they have the same mean. Given a type-1 error rate (α) of 0.1, 0.5, or 0.01, we can reject H_0. (However, we cannot reject H_0 at a type-1 error rate of 0.001.)

Problem 3

```
### a
tTestFunction = function(a, b) {
    numerator = mean(a) - mean(b)
    denominator = sqrt(var(a)/length(a) + var(b)/length(b))
    print(numerator/denominator)
}

### b
tTestFunction(vec1, vec2)

## [1] -3.279649

### Bonus problem:
```

```
tTestFunction2 = function(a, b) {
    if (is.numeric(a) & is.numeric(b)) {
        numerator = mean(a) - mean(b)
        denominator = sqrt(var(a)/length(a) + var(b)/length(b))
        print(numerator/denominator)
    } else {
        print("Both vectors must be numeric.")
    }
}
```

Probability Theory: Covariance and Correlation

Problem 4

a) $\mathbb{E}(X) = \frac{1}{3} * (-1) + \frac{1}{3} * (0) + \frac{1}{3} * (1) = 0$

$\mathbb{E}(Y) = \frac{1}{3} * (1) + \frac{1}{3} * (0) + \frac{1}{3} * (1) = \frac{2}{3}$

$\mathbb{E}(X * Y) = \frac{1}{3} * (-1) + \frac{1}{3} * (0) + \frac{1}{3} * (1) = 0$

$Cov(X, Y) = \mathbb{E}(X * Y) - \mathbb{E}(X) * \mathbb{E}(Y) = 0 - \frac{2}{3} * 0 = 0$

Are X and Y independent? No. We don't need to prove this because we know that $Y = X^2$, so Y was defined to be a function of X. The reason why we don't capture their dependence is that they are not *linearly dependent*. Instead, they are dependent through a quadratic function. Alternatively, one can also show that they're not independent because they violate $Pr(X = x \cap Y = y) = Pr(X = x) * Pr(Y = y)$ for some values of X and Y (for example X=0 and Y=1). This would be a proof by contradiction.

b) In order to solve this problem, as in the above problems, we need to calculate the following:

$Cov(X, Y) = \mathbb{E}(X * Y) - \mathbb{E}(X) * \mathbb{E}(Y)$

In order to do this, R is extremely helpful.

```
### a
```

```
### In order to calculate E(XY), use the following:

sum = 0
for (i in 1:6) {
    for (j in 1:6) {
        sum = (i + j) * (i - j) + sum
    }
}
```

This calculation returns the sum of all 36 outcomes, for all possible combinations of the first and the second die. Each of the above outcomes is equally likely with probability 1/36. We can either multiply every single value by 1/36 or we can, alternatively, simply divide the sum by 36 to get of $E(X * Y)$. The same applies to $E(X)$ and $E(Y)$ in the calculations below.

```
exy = sum/36  # = 0
exy

## [1] 0

### In order to calculate E(X), use the following:

sum2 = 0
for (i in 1:6) {
    for (j in 1:6) {
        sum2 = (i + j) + sum2
    }
}

ex = sum2/36
ex

## [1] 7

### In order to calculate E(Y), use the following:
```

```
sum3 = 0
for (i in 1:6) {
    for (j in 1:6) {
        sum3 = (i - j) + sum3
    }
}

ey = sum3/36
ey
```

```
## [1] 0
```

```
### The covariance is given by the following formula:

exy - ex * ey  # Returns 0.
```

```
## [1] 0
```

```
### Note: We can alternatively use vectorized operations to calculate this

### For E(X)

is <- rep(1:6, 6)
js <- rep(1:6, each = 6)

ex2 = sum(is + js)/36

### For E(Y)

is <- rep(1:6, 6)
js <- rep(1:6, each = 6)

ey2 = sum(is - js)/36

### For E(XY)
```

```
exy2 = sum((is + js) * (is - js))/36

### To get the result, we calculate

exy2 - ex2 * ey2  # Also returns zero

## [1] 0
```

Why aren't X and Y independent? Let's try to find a similar contradiction like above.

$Pr(X = 12 \cap Y = 1) = 0$ because this event never occurs.

$Pr(X = 12) = \frac{1}{36}$ and $Pr(Y = 1) = \frac{4}{36} = \frac{1}{9}$, meaning that $Pr(X = 12) * Pr(Y = 1) = \frac{1}{324}$

Accordingly, $Pr(X = 12) * Pr(Y = 1) = \frac{1}{324} \neq 0 = Pr(X = 12 \cap Y = 1)$