HW5

Ben Tankus

2/4/2021

```
df = data.frame(case1002)

#head(df)
```

Recall the Lab 5 data and models: Model 1: Both birds and non-echolocating bats have possibly different energy costs in flight to echolocating bats, after accounting for a linear relationship between log energy and log mass,

$$log(Energy_i) = \beta_0 + \beta_1 log(Mass_i) + \beta_2 non - ebat_i + \beta_3 bird_i + \epsilon_i$$

Model 2: The energy costs for non-echolocating bats and echolocating bats is the same, but possibly different to birds, after accounting for a linear relationship between log energy and log mass,

$$log(Energy_i) = \beta_0 + \beta_1 log(Mass_i) + \beta_3 bird_i + \epsilon_i$$

(a) (2 points) Use these two models to demonstrate that the Extra Sum of Squares F-test comparing models that only differ by one parameter is equivalent to a t-test of that parameter, and that the F-statistic is the t-statistic squared.

```
mod1 <- lm(log(Energy) ~ log(Mass)+ factor(Type) , data = df)
mod2 <- lm(log(Energy) ~ log(Mass)+ (Type == "non-echolocating birds") , data = df)

anova(mod1,mod2)
```

```
## Analysis of Variance Table
##
## Model 1: log(Energy) ~ log(Mass) + factor(Type)
## Model 2: log(Energy) ~ log(Mass) + (Type == "non-echolocating birds")
##   Res.Df     RSS Df  Sum of Sq      F Pr(>F)
## 1     16 0.55332
## 2     17 0.55853 -1 -0.0052094 0.1506  0.703
```

```
print("Mod 2 Summary")
```

```
## [1] "Mod 2 Summary"
```

```r
summary(mod1)
```

```
##
## Call:
## lm(formula = log(Energy) ~ log(Mass) + factor(Type), data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23224 -0.12199 -0.03637  0.12574  0.34457
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       -1.49770    0.14987  -9.993 2.77e-08 ***
## log(Mass)                          0.81496    0.04454  18.297 3.76e-12 ***
## factor(Type)non-echolocating bats -0.07866    0.20268  -0.388    0.703
## factor(Type)non-echolocating birds  0.02360    0.15760   0.150    0.883
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.186 on 16 degrees of freedom
## Multiple R-squared:  0.9815, Adjusted R-squared:  0.9781
## F-statistic: 283.6 on 3 and 16 DF,  p-value: 4.464e-14
```

```r
print(0.38^2)
```

```
## [1] 0.1444
```

The pvalue for type non-echolocating bats (mismatch term) is 0.703 which is identical to the model comparison pvalue, meaning they are equivilent tests. The t-statistic is magnitude 0.388, which is mathematically similar to the square root of the fstatistic 0.387.

(b) (2 points) Consider these two model specified in R's lm() notation: lm(log(Energy) ~ log(Mass), data = case1002) lm(log(Energy) ~ log(Mass) + Type, data = case1002) Describe in non-technical terms, (i.e. to someone who doesn't use R), why these two models, that look like they only differ by one parameter, cannot be compared with a single t-test.

These cannot be compared with a single t-test because when we include the 'Type' variable we are including more than one additional variable. While these two models are nested, the two models can only be compared if they are nested *and* only differ by one parameter.

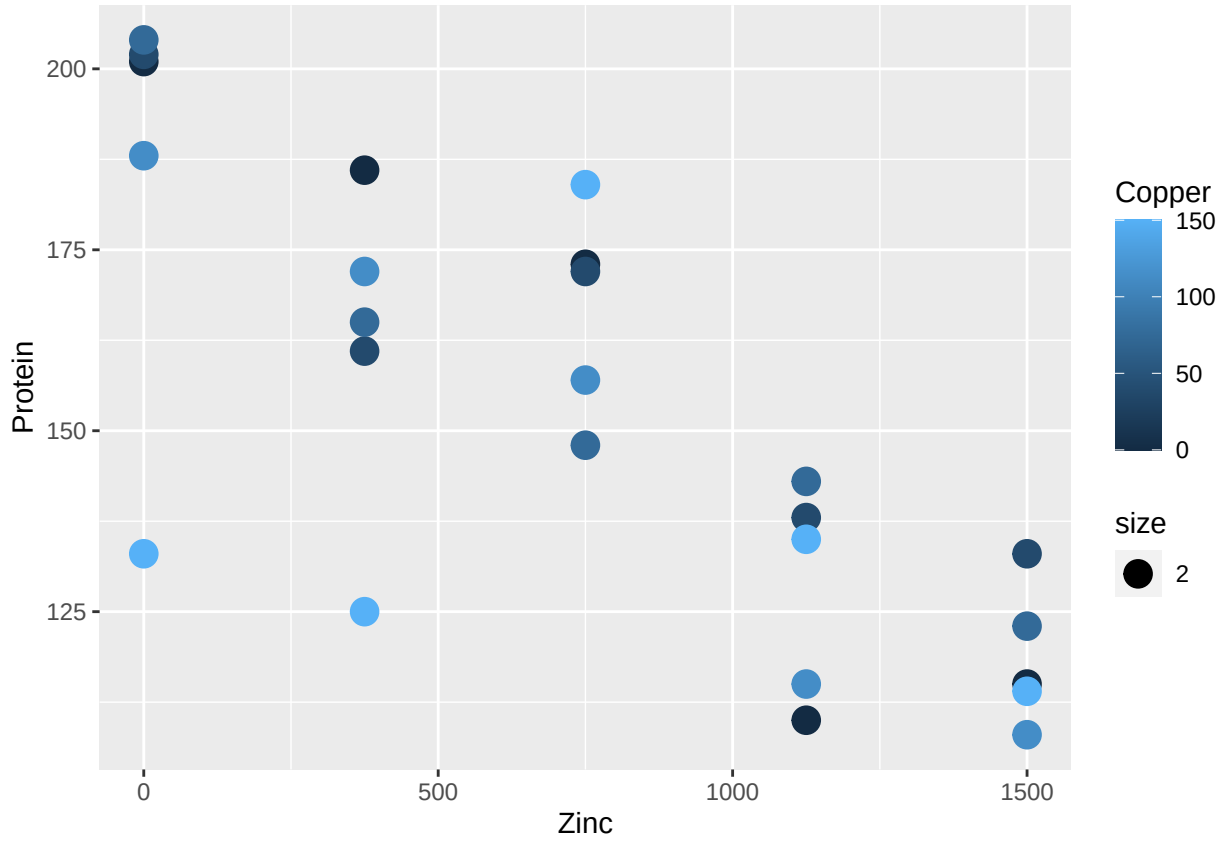
2. (6 points) Consider the data ex1014 in the Sleuth3 package. The data describes an experiment in which researchers randomly allocated 25 beakers containing minnow larvae to treatment combinations of zinc and copper. After four days, the minnows in each beaker were measured for their protein levels.

(a) Create a plot of protein against zinc, with points colored by the level of copper, and a plot of protein against copper, with points colored by the level of zinc. Describe any relationships you see.

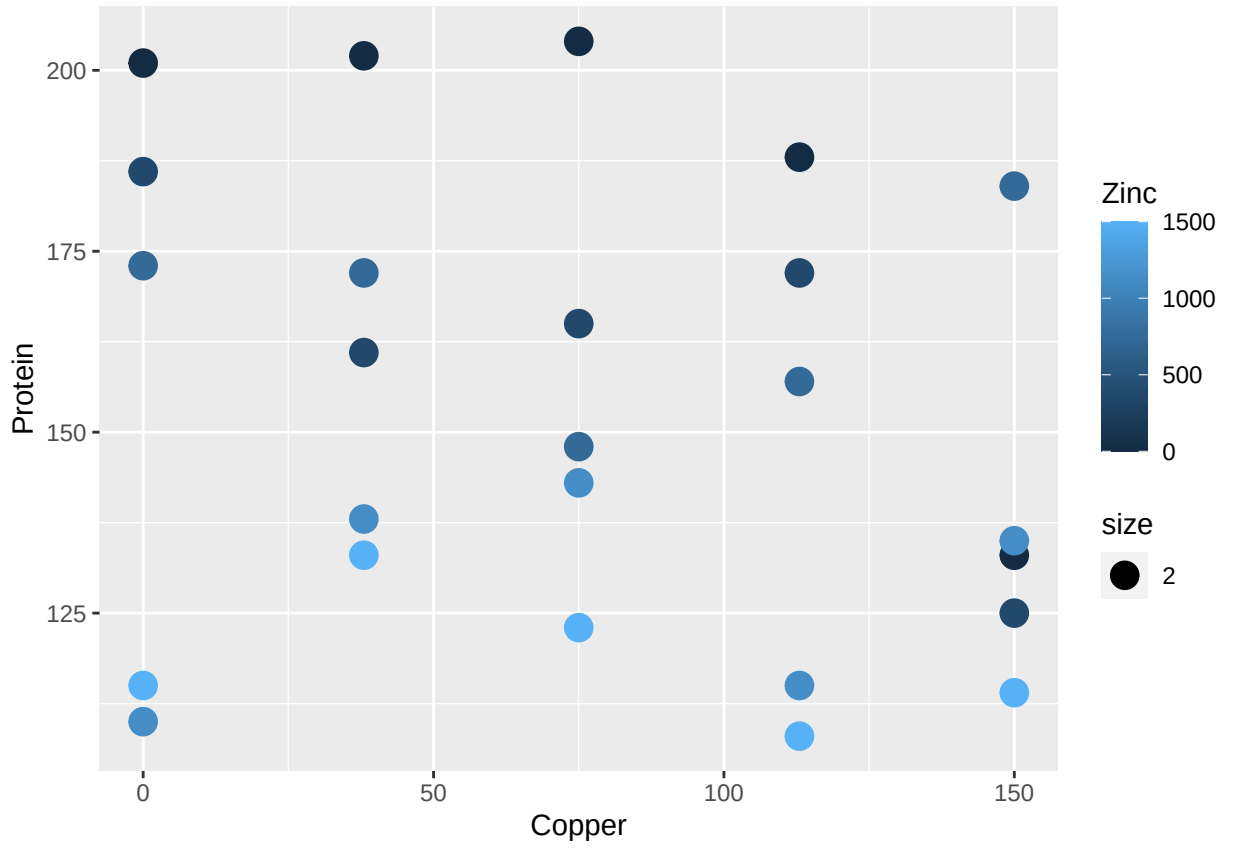
```
head(ex1014)
```

```
##   Copper Zinc Protein
## 1      0    0     201
## 2      0  375     186
## 3      0  750     173
## 4      0 1125     110
## 5      0 1500     115
## 6     38    0     202
```

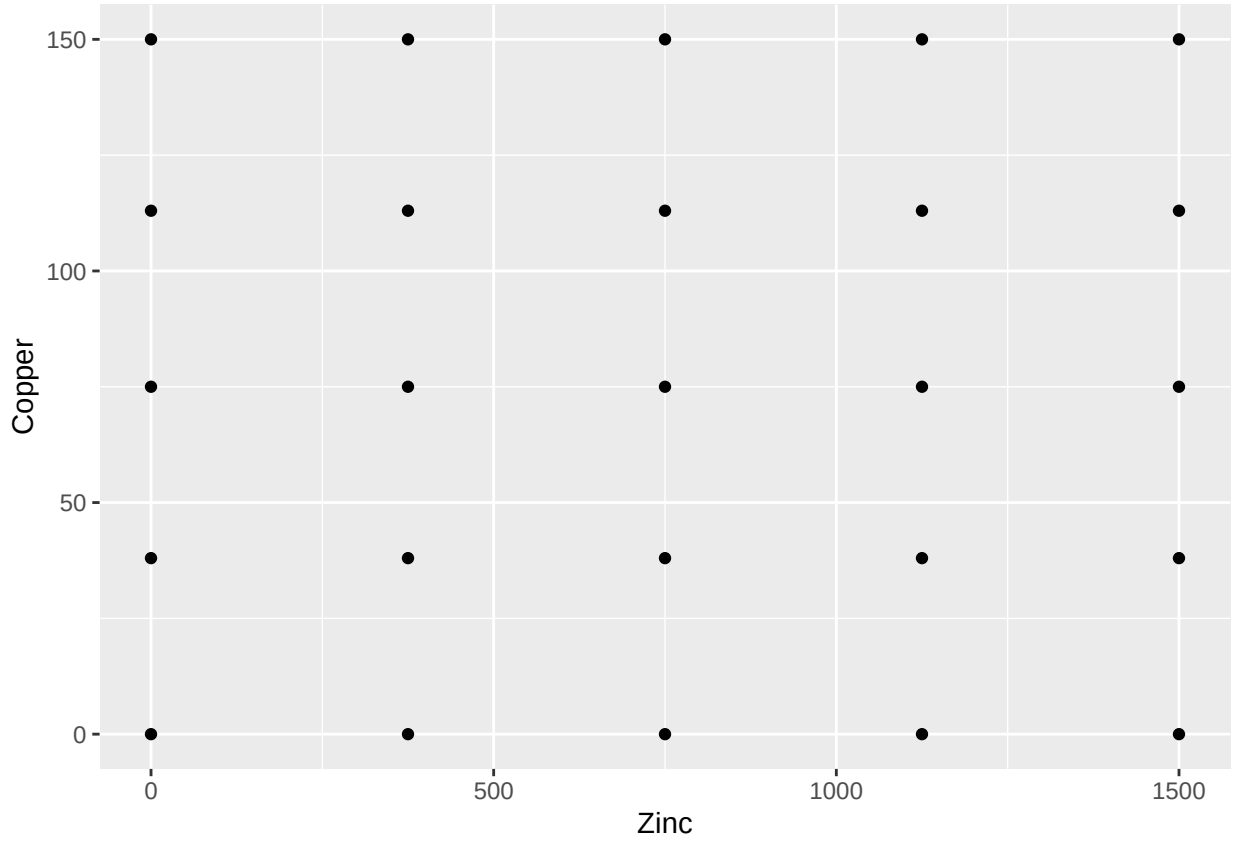
```
qplot(Zinc, Protein ,color = Copper, size = 2  ,data = ex1014)
```




```
qplot(Copper, Protein, color = Zinc, size = 2  ,data = ex1014)
```




```
qplot(Zinc, Copper  ,data = ex1014)
```



It looks like there is no clear relationship between protein and copper. Protien and Zinc looks like a negative relationship. There is no clear relationship between copper and zinc.

(b) Fit a model for protein that includes both main and interaction terms for Zinc and Copper. Examine the residual plots and comment on the validity of the assumptions.

```
modPro <- lm(Protein ~ Zinc + Copper + (Zinc : Copper), data = ex1014)
summary(modPro)
```

```
##
## Call:
## lm(formula = Protein ~ Zinc + Copper + (Zinc:Copper), data = ex1014)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.925 -10.272   1.524  10.626  41.920
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.113e+02  1.062e+01  19.903 4.13e-15 ***
## Zinc        -6.520e-02  1.156e-02  -5.642 1.34e-05 ***
## Copper      -3.398e-01  1.154e-01  -2.945  0.00772 **
## Zinc:Copper  2.727e-04  1.256e-04   2.171  0.04153 *
```

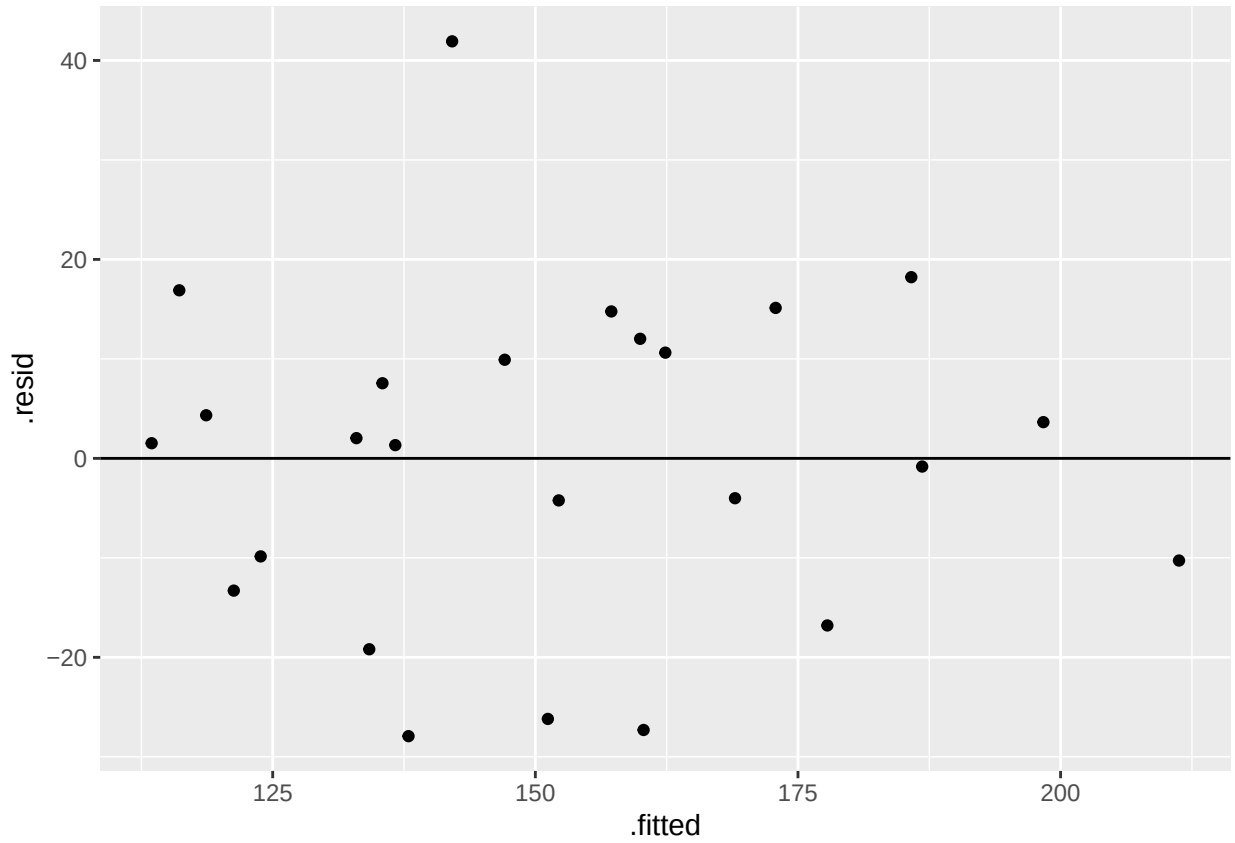
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.66 on 21 degrees of freedom
## Multiple R-squared:  0.7195, Adjusted R-squared:  0.6794
## F-statistic: 17.96 on 3 and 21 DF,  p-value: 5.21e-06
```

```
aug.ModPro <- augment(modPro)

(aug.ModPro)
```

```
## # A tibble: 25 x 9
##    Protein  Zinc Copper .fitted  .resid   .hat .sigma  .cooksd .std.resid
##      <int> <int>  <int>   <dbl>   <dbl>  <dbl>  <dbl>    <dbl>      <dbl>
##  1     201     0      0    211. -10.3   0.361    17.9 0.0749      -0.728
##  2     186   375      0    187.  -0.823 0.181    18.1 0.000146    -0.0515
##  3     173   750      0    162.  10.6   0.120    17.9 0.0141       0.642
##  4     110  1125      0    138. -27.9   0.181    16.7 0.168       -1.75
##  5     115  1500      0    113.   1.52  0.361    18.1 0.00165      0.108
##  6     202     0     38    198.   3.64  0.179    18.1 0.00282      0.227
##  7     161   375     38    178. -16.8   0.0895   17.7 0.0244      -0.997
##  8     172   750     38    157.  14.8   0.0597   17.8 0.0118       0.862
##  9     138  1125     38    137.   1.33  0.0895   18.1 0.000153     0.0790
## 10     133  1500     38    116.  16.9   0.179    17.6 0.0608       1.06
## # ... with 15 more rows
```

```
qplot(.fitted, .resid, data = aug.ModPro) + geom_hline(yintercept = 0)
```



Residual plot is clear of any assumption violations. There is one outlier point, but the leverage of said point is not outside of the normal range in comparison to the other points.

(c) Fit a model for log protein that includes both main and interaction terms for Zinc and Copper. Examine the residual plots and comment on the validity of the assumptions. Is there evidence the model on the log scale better satisfies the assumptions?

```
modLogPro <- lm(log(Protein) ~ Zinc + Copper + (Zinc:Copper), data = ex1014)
summary(modLogPro)
```

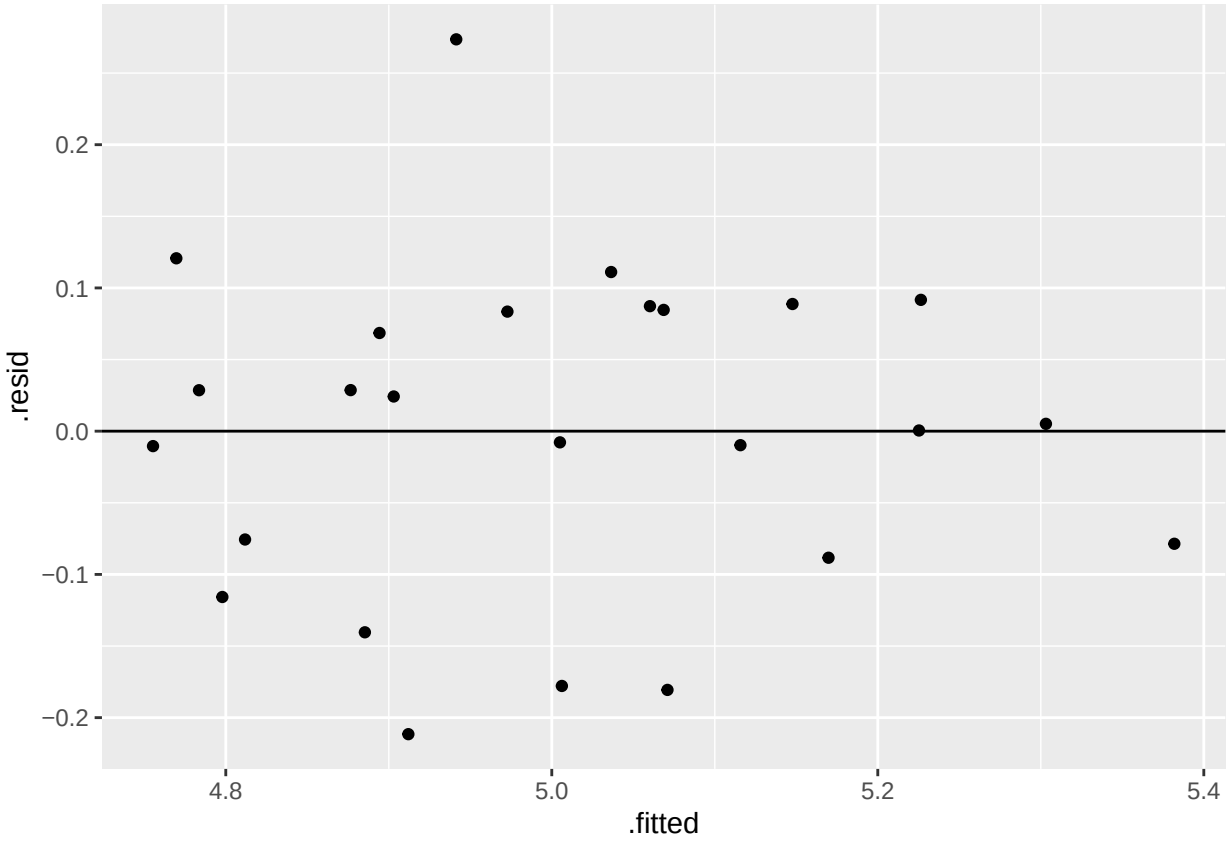
```
##
## Call:
## lm(formula = log(Protein) ~ Zinc + Copper + (Zinc:Copper), data = ex1014)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.211520 -0.078607  0.005128  0.084654  0.273542
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.382e+00  7.289e-02  73.839  < 2e-16 ***
## Zinc        -4.177e-04  7.935e-05  -5.264 3.22e-05 ***
## Copper      -2.073e-03  7.921e-04  -2.617   0.0161 *
```

```
## Zinc:Copper  1.633e-06  8.623e-07   1.894   0.0721 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1213 on 21 degrees of freedom
## Multiple R-squared:  0.6983, Adjusted R-squared:  0.6552
## F-statistic:  16.2 on 3 and 21 DF,  p-value: 1.105e-05
```

```
aug.ModLogPro <- fortify(modLogPro)
head(aug.ModLogPro)
```

```
##   log(Protein) Zinc Copper      .hat    .sigma      .cooksd  .fitted
## 1     5.303305    0      0 0.3612766 0.1222966 9.302758e-02 5.381912
## 2     5.225747  375      0 0.1806383 0.1242584 1.019673e-06 5.225275
## 3     5.153292  750      0 0.1204255 0.1226083 1.896478e-02 5.068637
## 4     4.700480 1125      0 0.1806383 0.1127384 2.046623e-01 4.912000
## 5     4.744932 1500      0 0.3612766 0.1242242 1.638070e-03 4.755363
## 6     5.308268    0     38 0.1790426 0.1242520 1.187702e-04 5.303140
##          .resid    .stdresid
## 1 -0.0786068156 -0.811096229
## 2  0.0004721311  0.004301237
## 3  0.0846542329  0.744356483
## 4 -0.2115198148 -1.927000508
## 5 -0.0104308713 -0.107629857
## 6  0.0051281131  0.046673018
```

```
qplot(.fitted, .resid, data = aug.ModLogPro) + geom_hline(yintercept = 0) # WHAT IS STD.RESID VS RES?
```




```
AIC(modLogPro, modPro)
```

```
##           df       AIC
## modLogPro  5 -28.90123
## modPro     5 220.15526
```

```
BIC(modLogPro, modPro)
```

```
##           df       BIC
## modLogPro  5 -22.80685
## modPro     5 226.24964
```

It looks as though the assumptions are not violated in either model. Both residual plots have a similar shape, but the log plot has a much smaller y-axis. This artificially deflates the AIC and BIC values so we can't really say with confidence one way or another which model is better. It's not an apples-to-apples comparison.

(d) Conduct a test for the interaction term in the model in (b). Make sure you include completely specify your model, hypotheses, test statistic and p-value. Write a short non-technical summary based on your result in the context of the study.

Model:

$$protein = \beta_0 + \beta_1 Zinc + \beta_2 Copper + \beta_3 * (Zinc : Copper)$$

Hypothesis:

H_0 :

$$\beta_0 = 0, \beta_1 = 0, \beta_2 = 0, \beta 3 = 0$$

H_A : At least one of the β values is not equal to zero

```
summary(modPro)
```

```
##
## Call:
## lm(formula = Protein ~ Zinc + Copper + (Zinc:Copper), data = ex1014)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.925 -10.272   1.524  10.626  41.920
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.113e+02  1.062e+01  19.903 4.13e-15 ***
## Zinc        -6.520e-02  1.156e-02  -5.642 1.34e-05 ***
## Copper      -3.398e-01  1.154e-01  -2.945  0.00772 **
## Zinc:Copper  2.727e-04  1.256e-04   2.171  0.04153 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.66 on 21 degrees of freedom
## Multiple R-squared:  0.7195, Adjusted R-squared:  0.6794
## F-statistic: 17.96 on 3 and 21 DF,  p-value: 5.21e-06
```

Test statistic is 2.171, pvalue is 0.04153. This means we have strong statistical evidents to suggest that the β_3 iteraction term is non-zero. This means there is significant contribution to the model from the interaction of zinc and copper.

(e) Produce mean protein levels, along with confidence intervals, for all combinations of Zinc and Copper based on the model in (b). Describe the effect of the interaction between Zinc and Copper on mean protein. (Hint: making a plot of these predictions might help).

```
modpro.Aug <- augment(modPro)
conf <- predict(modPro, interval = 'confidence')
conf
```

```
##         fit       lwr      upr
## 1  211.2718 189.19625 233.3473
## 2  186.8229 171.21316 202.4327
## 3  162.3740 149.62873 175.1194
## 4  137.9252 122.31541 153.5349
```

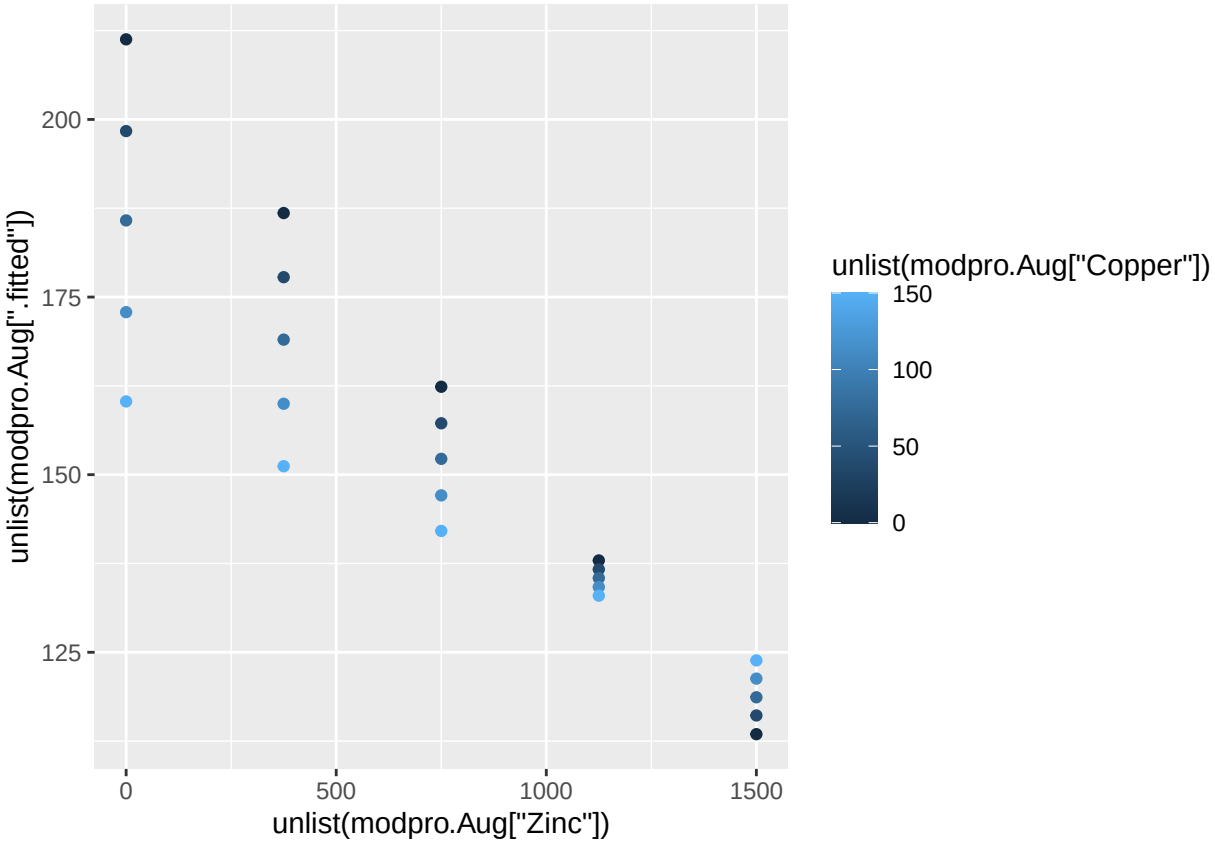
```
## 5  113.4763  91.40077 135.5518
## 6  198.3600 182.81932 213.9006
## 7  177.7964 166.80754 188.7853
## 8  157.2329 148.26050 166.2053
## 9  136.6694 125.68046 147.6583
## 10 116.1058 100.56517 131.6465
## 11 185.7880 173.06509 198.5108
## 12 169.0075 160.01108 178.0039
## 13 152.2271 144.88151 159.5726
## 14 135.4466 126.45018 144.4430
## 15 118.6662 105.94329 131.3890
## 16 172.8761 157.25239 188.4999
## 17 159.9810 148.93337 171.0287
## 18 147.0859 138.06554 156.1063
## 19 134.1908 123.14314 145.2385
## 20 121.2957 105.67193 136.9195
## 21 160.3041 138.30695 182.3013
## 22 151.1921 135.63774 166.7465
## 23 142.0801 129.38000 154.7802
## 24 132.9680 117.41369 148.5224
## 25 123.8560 101.85884 145.8532
```

```
modpro.Aug['lower'] = conf[,2]
modpro.Aug['upper'] = conf[,3]

(modpro.Aug)
```

```
## # A tibble: 25 x 11
##    Protein  Zinc Copper .fitted  .resid   .hat .sigma .cooksd .std.resid lower
##      <int> <int>  <int>   <dbl>   <dbl>  <dbl>  <dbl>   <dbl>      <dbl> <dbl>
##  1     201     0      0    211. -10.3   0.361    17.9 7.49e-2    -0.728  189.
##  2     186   375      0    187.  -0.823 0.181    18.1 1.46e-4    -0.0515 171.
##  3     173   750      0    162.  10.6   0.120    17.9 1.41e-2     0.642  150.
##  4     110  1125      0    138. -27.9   0.181    16.7 1.68e-1    -1.75   122.
##  5     115  1500      0    113.   1.52  0.361    18.1 1.65e-3     0.108   91.4
##  6     202     0     38    198.   3.64  0.179    18.1 2.82e-3     0.227  183.
##  7     161   375     38    178. -16.8   0.0895   17.7 2.44e-2    -0.997  167.
##  8     172   750     38    157.  14.8   0.0597   17.8 1.18e-2     0.862  148.
##  9     138  1125     38    137.   1.33  0.0895   18.1 1.53e-4     0.0790 126.
## 10     133  1500     38    116.  16.9   0.179    17.6 6.08e-2     1.06   101.
## # ... with 15 more rows, and 1 more variable: upper <dbl>
```

```
qplot(x = unlist(modpro.Aug['Zinc']), y = unlist(modpro.Aug['.fitted']), color = unlist(modpro.Aug['Copp
```



It looks as though in general, as the concentration of zinc and copper increase, the fitted value of protien decreases, as well as the variance between the fitted values. As the concentration of zinc increases, the interaction between zinc and copper matters less and less to the values of protien.

(f) Try fitting a model where the levels of both Zinc and Copper are treated as categories, and include an interaction between the now categorical Zinc and Copper. Examine the model. Describe the problem with this model that prevents inference from proceeding

```
mod2F <- lm(Protein ~ factor(Zinc) + factor(Copper) + (Zinc:Copper), data =  ex1014)
summary(mod2F)
```

```
##
## Call:
## lm(formula = Protein ~ factor(Zinc) + factor(Copper) + (Zinc:Copper),
##     data = ex1014)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.3041 -12.8649  -0.2071  11.7111  31.2000
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         2.058e+02  1.259e+01  16.340 5.77e-11 ***
## factor(Zinc)375    -3.149e+01  1.158e+01  -2.720   0.0158 *
## factor(Zinc)750    -3.418e+01  1.307e+01  -2.615   0.0195 *
## factor(Zinc)1125   -8.047e+01  1.523e+01  -5.283 9.20e-05 ***
```

```
## factor(Zinc)1500  -9.776e+01  1.782e+01  -5.484 6.29e-05 ***
## factor(Copper)38  -3.571e+00  1.159e+01  -0.308   0.7622
## factor(Copper)75  -1.574e+01  1.306e+01  -1.205   0.2468
## factor(Copper)113 -3.211e+01  1.524e+01  -2.106   0.0524 .
## factor(Copper)150 -4.947e+01  1.780e+01  -2.780   0.0140 *
## Zinc:Copper        2.727e-04  1.241e-04   2.197   0.0441 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.45 on 15 degrees of freedom
## Multiple R-squared:  0.8044, Adjusted R-squared:  0.6871
## F-statistic: 6.854 on 9 and 15 DF,  p-value: 0.0006119
```

One cannot make inferences outside of the model has been trained on (our existing dataframe) because categorical values cannot mathemateically interpolate or extrapolate categorical data.