# MAST30027: Modern Applied Statistics

## Assignment 2 Solution 2023

**Solution: There is no unique answer for this problem. This is a sample solution.**
You can read the data using the following command.

```
> data <- read.table(file ="assignment2_2023.txt", header=TRUE)
> dim(data)

[1] 48  5

> names(data)

[1] "numDefects" "prebake"    "flux"        "cooling"     "temp"

> data$prebake <- factor(data$prebake)
> data$flux <- factor(data$flux)
> data$cooling <- factor(data$cooling)
> data$temp <- factor(data$temp)
```

First, we visualize data by plotting the responses against the predictors.
(Predictors are factor, so we don't need to check linearity.)

```
> par(mfrow=c(2,2), mar=c(4,4,1,1))
> plot(numDefects ~ prebake, data)
> plot(numDefects ~ flux, data)
> plot(numDefects ~ cooling, data)
> plot(numDefects ~ temp, data)
```
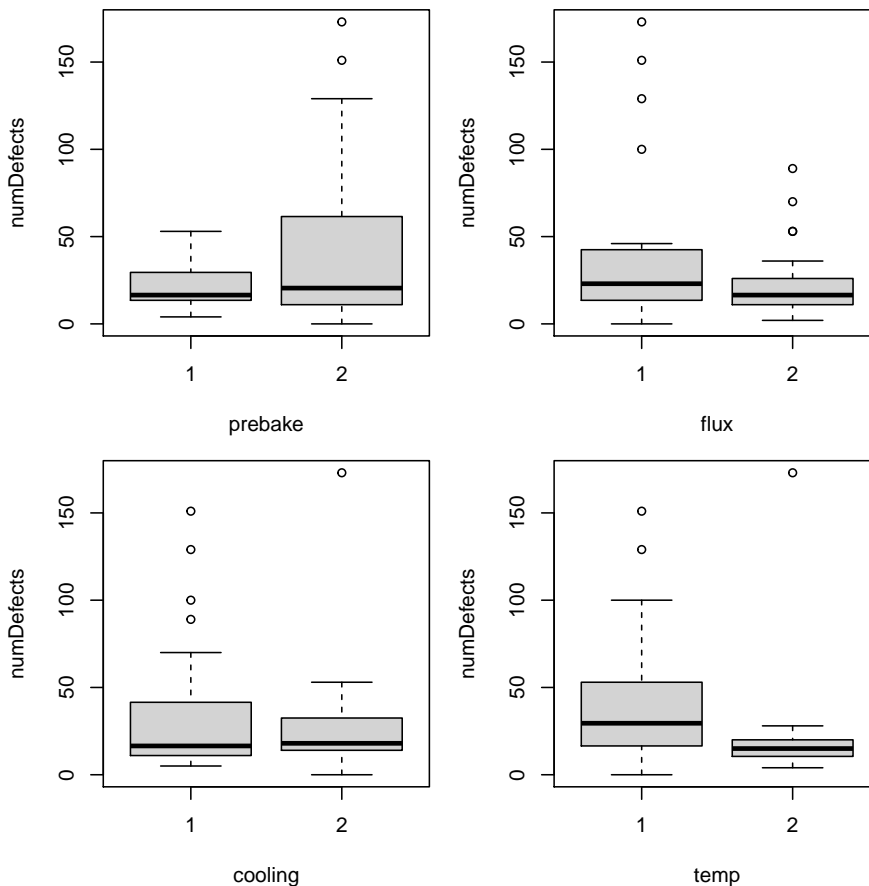
The variable `temp` has a clear effect on the number of defects, but for the other variables things are less clear.

Are there any two way interactions between factors? We use interaction plots to check.

```
> par(mfrow=c(3,2))
> par(mar=c(4,4,1,4))
> with(data, interaction.plot(prebake,flux,numDefects))
> with(data, interaction.plot(prebake,cooling,numDefects))
> with(data, interaction.plot(prebake,temp,numDefects))
> with(data, interaction.plot(flux,cooling,numDefects))
> with(data, interaction.plot(flux,temp,numDefects))
> with(data, interaction.plot(cooling,temp,numDefects))
```
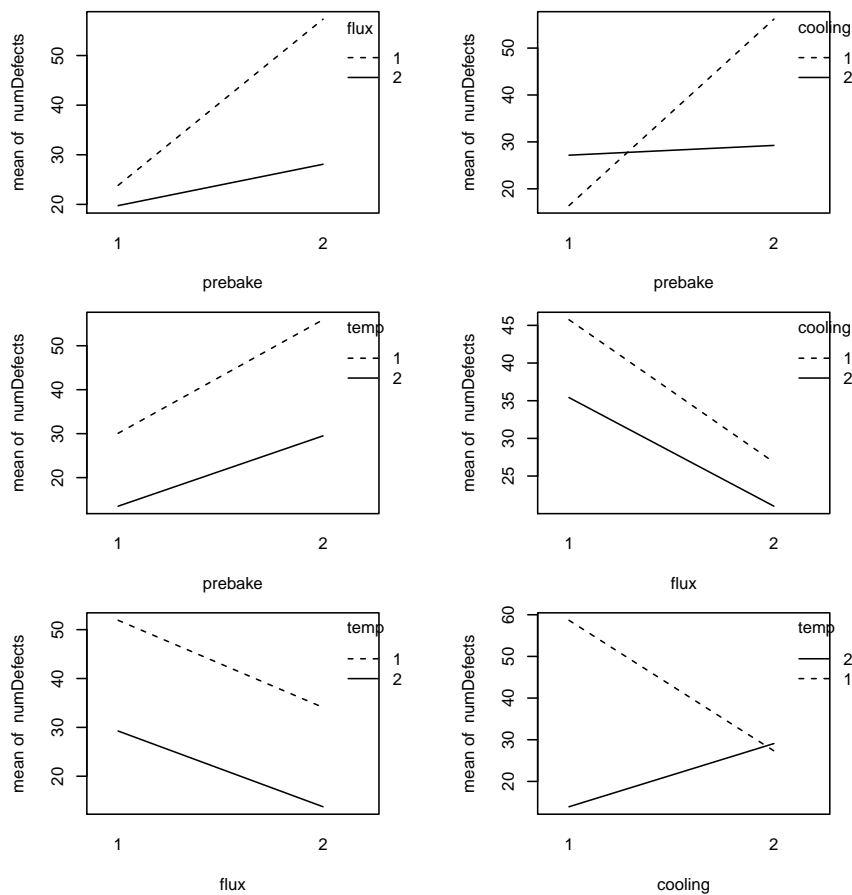


There are clear interactions between `cooling` and `prebake`, and between `temp` and `cooling`.

Given that we have count data, a Poisson regression model with a log link is a good place to start. Since there are interactions, we use the main effects and all two-way interactions at once, then using stepwise selection with the AIC to select a parsimonious model.

```
> pmod <- glm(numDefects ~ (prebake + flux + cooling + temp)^2,
+             family=poisson, data=data)
> pmod.f <- step(pmod, trace=0)
> summary(pmod.f)

Call:
glm(formula = numDefects ~ prebake + flux + cooling + temp +
    prebake:flux + prebake:cooling + prebake:temp + flux:temp +
    cooling:temp, family = poisson, data = data)
```

```
Deviance Residuals:
    Min      1Q   Median      3Q      Max
-7.7921  -2.6541  -0.2946   1.3936  13.5042

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       3.40957    0.08598  39.653  < 2e-16 ***
prebake2          1.33040    0.09625  13.823  < 2e-16 ***
flux2            -0.09016    0.09461  -0.953   0.3406
cooling2          0.07550    0.09794   0.771   0.4408
temp2            -1.86838    0.14883 -12.554  < 2e-16 ***
prebake2:flux2   -0.51723    0.11047  -4.682 2.84e-06 ***
prebake2:cooling2 -1.40241   0.12314 -11.389  < 2e-16 ***
prebake2:temp2    0.66403    0.13185   5.036 4.75e-07 ***
flux2:temp2      -0.31964    0.11465  -2.788   0.0053 **
cooling2:temp2    1.70784    0.12439  13.730  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 1450.52  on 47  degrees of freedom
Residual deviance:  626.99  on 38  degrees of freedom
AIC: 877.73

Number of Fisher Scoring iterations: 5
```

We see that the interaction term between `flux` and `cooling` was removed. We check the model adequacy.

```
> pchisq(deviance(pmod.f), pmod.f$df.residual, lower.tail=FALSE)

[1] 1.005731e-107
```

The p-value is very small, thus something is amiss with the model. The diagnostic plots might be able to tell us what is wrong.

```
> library(faraway)
> par(mfrow=c(2,2))
> plot(predict(pmod.f), residuals(pmod.f), xlab="eta", ylab="deviance res")
> halfnorm(residuals(pmod.f), ylab="deviance res")
> halfnorm(rstudent(pmod.f), ylab="jackknife res")
> halfnorm(cooks.distance(pmod.f), ylab="cooks distance")
```
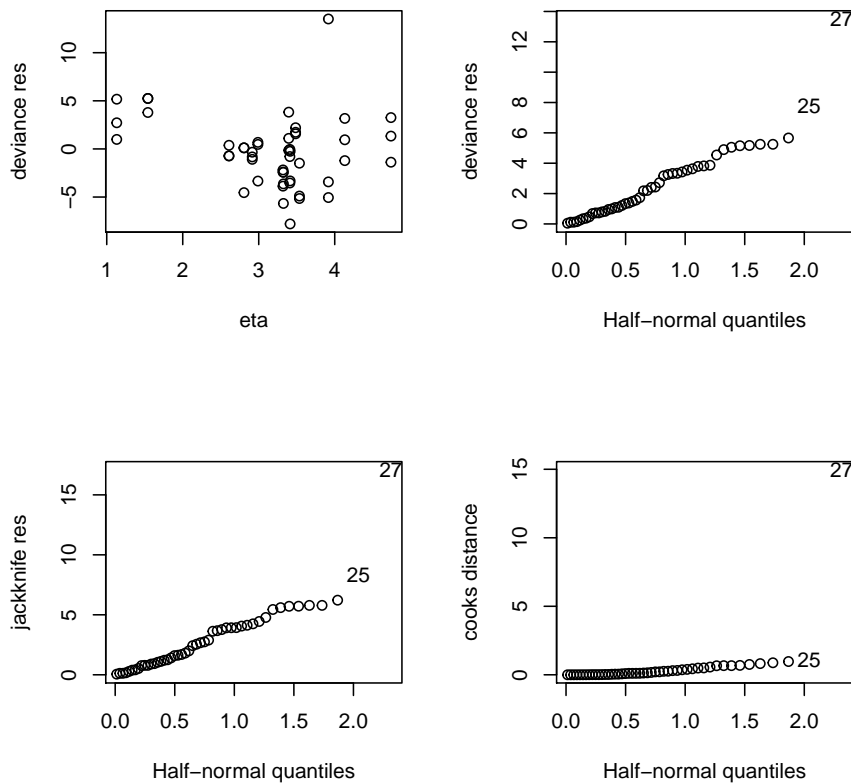
Point 27 has very large Cook's distance, jackknife residual, and deviance residual, suggesting that this point might be an influential point. Thus, we will fit the model without the point 27.

```
> pmod2 <- glm(numDefects ~ (prebake + flux + cooling + temp)^2,
+              family=poisson, subset=-27, data=data)
> pmod2.f <- step(pmod2, trace=0)
> summary(pmod2.f)

Call:
glm(formula = numDefects ~ prebake + flux + cooling + temp +
    prebake:cooling + flux:cooling + flux:temp + cooling:temp,
    family = poisson, data = data, subset = -27)

Deviance Residuals:
    Min      1Q  Median      3Q     Max
-6.5090 -1.5709 -0.5039  1.5787  4.2854

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)       3.52381    0.07795  45.205  < 2e-16 ***
prebake2          1.23003    0.08099  15.187  < 2e-16 ***
flux2            -0.57083    0.07473  -7.638 2.20e-14 ***
cooling2          0.07465    0.10984   0.680    0.497
temp2            -1.51196    0.10010 -15.104  < 2e-16 ***
prebake2:cooling2 -1.77525   0.12359 -14.364  < 2e-16 ***
flux2:cooling2    0.45881    0.11527   3.980 6.88e-05 ***
flux2:temp2       0.19085    0.12844   1.486    0.137
```

4

```
cooling2:temp2     0.85535    0.12840   6.662 2.71e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 1137.32  on 46  degrees of freedom
Residual deviance:  289.02  on 38  degrees of freedom
AIC: 530.77

Number of Fisher Scoring iterations: 5

> pchisq(deviance(pmod2.f), pmod2.f$df.residual, lower.tail=FALSE)

[1] 2.343993e-40

> par(mfrow=c(2,2))
> plot(predict(pmod2.f), residuals(pmod2.f), xlab="eta", ylab="deviance res")
> halfnorm(residuals(pmod2.f), ylab="deviance res")
> halfnorm(rstudent(pmod2.f), ylab="jackknife res")
> halfnorm(cooks.distance(pmod2.f), ylab="cooks distance")
```
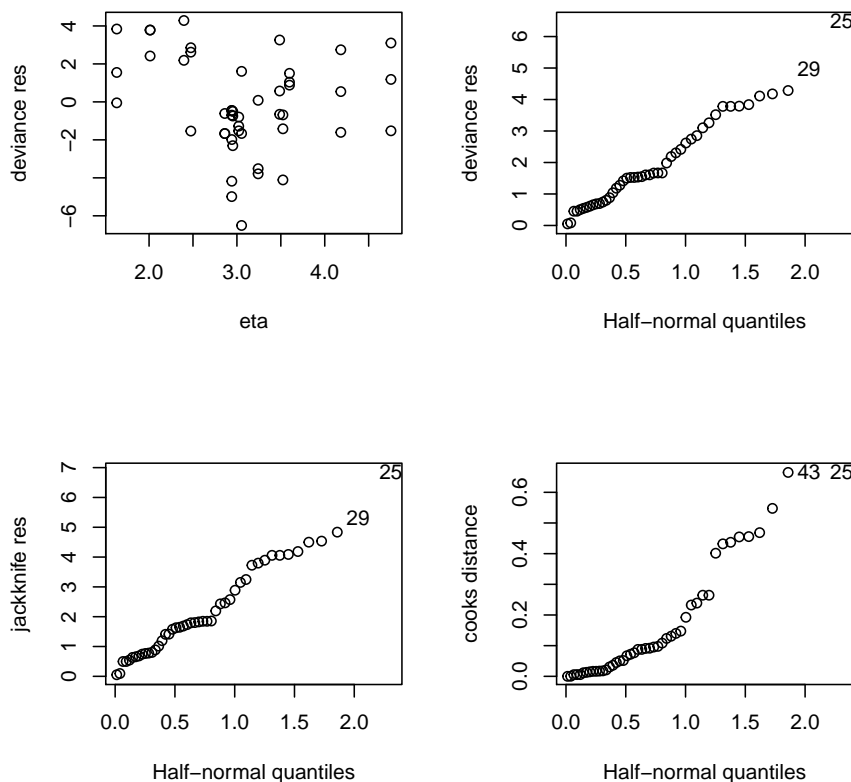


The diagnostic plots look fine now. However, we still observe that p-value of model adequacy test is low. Let's check overdispersion.

```
> (phi <- sum(residuals(pmod2.f, type="pearson")^2/pmod2.f$df.residual))

[1] 7.355161
```

The large value of $\hat{\phi}$ suggests that overdispersion occurs. Thus, we fit the data by quasi-Poisson model.

```
> pmod.q <- glm(numDefects ~ prebake + flux + cooling + temp + prebake:cooling +
+               flux:cooling + flux:temp + cooling:temp,
+               family=quasipoisson, subset=-27, data=data)
> summary(pmod.q)

Call:
glm(formula = numDefects ~ prebake + flux + cooling + temp +
    prebake:cooling + flux:cooling + flux:temp + cooling:temp,
    family = quasipoisson, data = data, subset = -27)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-6.5090  -1.5709  -0.5039   1.5787   4.2854

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        3.52381    0.21141  16.668  < 2e-16 ***
prebake2           1.23003    0.21966   5.600 2.02e-06 ***
flux2             -0.57083    0.20268  -2.816  0.00766 **
cooling2           0.07465    0.29790   0.251  0.80349
temp2             -1.51196    0.27148  -5.569 2.22e-06 ***
prebake2:cooling2 -1.77525    0.33519  -5.296 5.25e-06 ***
flux2:cooling2     0.45881    0.31261   1.468  0.15042
flux2:temp2        0.19085    0.34833   0.548  0.58696
cooling2:temp2     0.85535    0.34822   2.456  0.01872 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 7.355163)

    Null deviance: 1137.32  on 46  degrees of freedom
Residual deviance:  289.02  on 38  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

From the p-values in the summary table, `flux2:cooling2` and `flux2:temp2` are not significant. So we will consider a model without these two terms and compare the two models using F-test.

```
> pmod.q.r <- glm(numDefects ~ prebake + flux + cooling + temp + prebake:cooling +
+                 cooling:temp, family=quasipoisson, subset=-27, data=data)
> anova(pmod.q.r, pmod.q, test="F")

Analysis of Deviance Table

Model 1: numDefects ~ prebake + flux + cooling + temp + prebake:cooling +
    cooling:temp
Model 2: numDefects ~ prebake + flux + cooling + temp + prebake:cooling +
    flux:cooling + flux:temp + cooling:temp
  Resid. Df Resid. Dev Df Deviance    F Pr(>F)
1        40     309.76
2        38     289.02  2   20.741 1.41 0.2566
```

The test shows that we should remove these two terms.

```
> summary(pmod.q.r)

Call:
glm(formula = numDefects ~ prebake + flux + cooling + temp +
    prebake:cooling + cooling:temp, family = quasipoisson, data = data,
    subset = -27)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-6.893  -1.676  -0.090   1.432   4.104

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)        3.4384     0.2091  16.448  < 2e-16 ***
prebake2           1.2300     0.2212   5.561 1.95e-06 ***
flux2             -0.3502     0.1494  -2.344   0.0241 *
cooling2           0.2601     0.2675   0.972   0.3367
temp2             -1.4388     0.2351  -6.121 3.19e-07 ***
prebake2:cooling2 -1.7608     0.3375  -5.218 5.90e-06 ***
cooling2:temp2     0.8914     0.3472   2.567   0.0141 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 7.458044)

    Null deviance: 1137.32  on 46  degrees of freedom
Residual deviance:  309.76  on 40  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

Now the `summary` looks fine. Note the main effect `cooling` is not significant. However, the interaction terms contains `cooling`. Hence, we will keep the main effect.

```
> par(mfrow=c(2,2))
> plot(predict(pmod.q.r), residuals(pmod.q.r), xlab="eta", ylab="deviance res")
> halfnorm(residuals(pmod.q.r), ylab="deviance res")
> halfnorm(rstudent(pmod.q.r), ylab="jackknife res")
> halfnorm(cooks.distance(pmod.q.r), ylab="cooks distance")
```
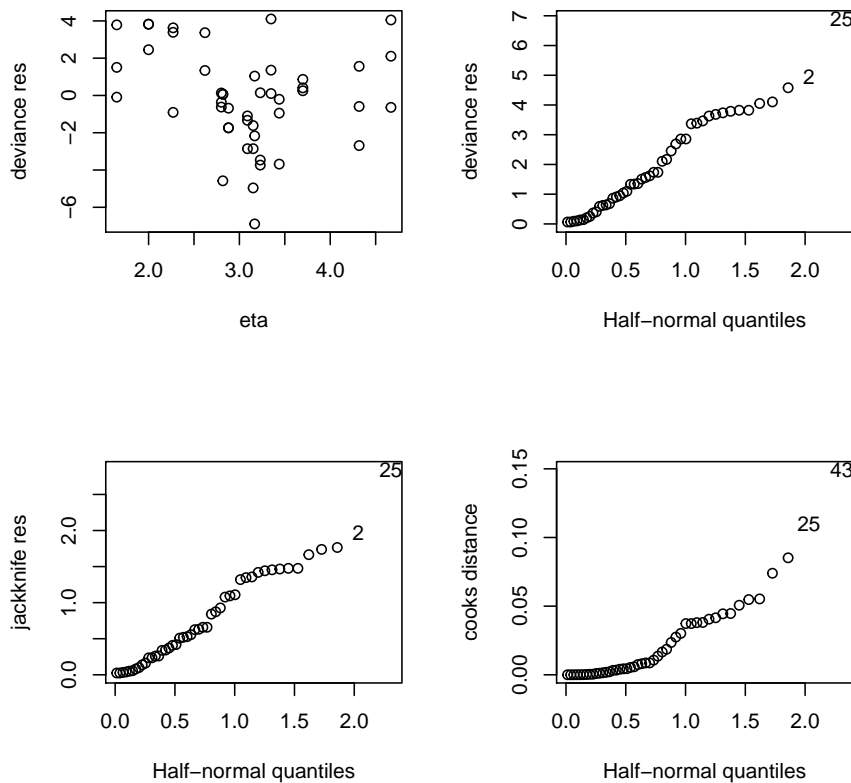
The diagnostic plots of the resulting model also look fine.

`flux` is the only main effect which does not interact with any other factors. It has a negative effect on the number of defects since its coefficient is negative. `prebake` has a positive main effect on the number of defects. However, the coefficient of the interaction between `prebake` and `cooling` is negative and the effect is larger than the main effect of `prebake`. This means that the effect of `prebake` is negative when `cooling = 2`. (Other interpretations are acceptable as long as they are well explained.)