Stat 570 HW6

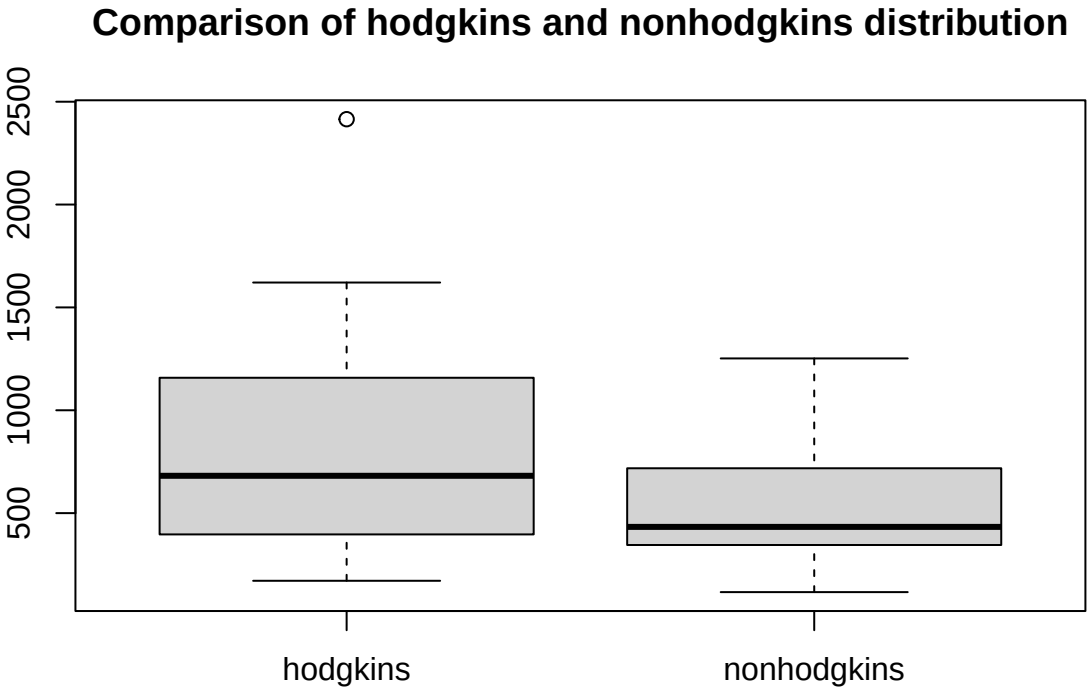
Dongyang Wang

2022-11-12

Q1

a

```
##             Min. 1st Qu. Median   Mean 3rd Qu. Max.
## hodgkins     171  396.75  681.5 823.20    1131 2415
## nonhodgkins  116  360.00  433.0 522.05     709 1252
```



As we can observe, T4 cells are more prevalent in Hodgkins remissions than in others, since it has a higher mean and greater variance. The data is right skewed.

b

```
## Warning: Setting row names on a tibble is deprecated.
```

```
## # A tibble: 3 x 3
##   estimate conf.low conf.high
## *    <dbl>    <dbl>     <dbl>
## 1 -301.    -542.     -60.8
## 2   -0.398   -0.756   -0.0405
## 3   -5.21    -9.53    -0.881
```

Here we have obtained a table showing the difference between the two categories, by using the beta 1 in the

linear model, where I input 0 for the hodgkin's and 1 for nonhodgkin's. In choosing a good mode, we can first consider how we want to interpret the model. That leaves out the third choice because it's hard to interpret. The first one is straightforward and the second one involves the exponential of beta1 for each unit increase in x (i.e, the difference in the two groups, changing from nonhodgkins to hodgkins). Also, considering the right skewness of the data, it's more preferable to use the log transformation for variance stabilization.

c

To study this we can want to test whether $\beta_1 = 0$, For the Poisson model, we use the canonical log link, i.e., $g(\mu) = \log \mu$. We go back to original scale by $\exp(\beta_0 + \beta_1 x)$ where x= 1 when it is nonhodgkins and 0 for hodgkins. For Gamma, the canonical is reciprocal. $g(\mu) = 1/\mu$. We go back to original scale by $1/(\beta_0 + \beta_1 x)$ where x= 1 when it is nonhodgkins and 0 for hodgkins. For Inverse Gaussian, the canonical is reciprocal. $g(\mu) = 1/\mu^2$. We go back to original scale by $(\beta_0 + \beta_1 x)^{-1/2}$ where x= 1 when it is nonhodgkins and 0 for hodgkins.

d

```
##                            b0            b1  b0 Lower CI   b1 Upper CI
## model_pois      6.713199e+00 -4.554358e-01 6.700380e+00 -4.760139e-01
## model_gamma     1.214772e-03  7.007537e-04 9.342666e-04  1.769618e-04
## model_inv_gauss 1.475670e-06  2.193567e-06 7.196904e-07  5.165490e-07
##                  b0 Lower CI   b1 Upper CI
## model_pois      6.726018e+00 -4.348577e-01
## model_gamma     1.495277e-03  1.224546e-03
## model_inv_gauss 2.231650e-06  3.870585e-06
```

Based on the result above, since 0 is not in the interval, we can conclude that the means for two groups are in fact different.

Q2

a

Since $\log y_i \sim N(\log(\frac{D}{V}\exp(-k_e x_i)), \sigma^2)$

The likelihood is $L = \Pi_{i=1}^n \frac{1}{\sqrt{2\pi}\sigma} \exp(\frac{(\log y_i - \log D + \log V + k_e x_i)^2}{2\sigma^2})$

The log likelihood is $l = \frac{n}{2}\log\sqrt{2\pi} - n\log\sigma + \Sigma_{i=1}^n \frac{(\log y_i - \log D + \log V + k_e x_i)^2}{2\sigma^2}$

So, we have the score functions.

$\frac{dl}{dk_e} = -\frac{1}{\sigma^2}\Sigma_{i=1}^n x_i(\log y_i - \log D + \log V + k_e x_i)$

$\frac{dl}{dV} = -\frac{1}{V\sigma^2}\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)$

$\frac{dl}{d\sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{\sigma^4}\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)^2$

We also have the information,

$I_{k_e k_e} = -E(\frac{dl}{dk_e dk_e}) = \frac{\Sigma_{i=1}^n x_i^2}{\sigma^2}$

$I_{VV} = -E(\frac{dl}{dVdV}) = -E(\frac{\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)}{V^2} - \frac{n}{V^2\sigma^2}) = \frac{n}{V^2\sigma^2}$

$I_{\sigma^2\sigma^2} = -E(\frac{dl}{d\sigma^2 d\sigma^2}) = -E(\frac{n}{2\sigma^4} - \frac{\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)}{\sigma^6}) = -E(\frac{n}{2\sigma^4} - \frac{n\sigma^2}{\sigma^6}) = \frac{n}{2\sigma^4}$

$I_{k_e V} = I_{V k_e} = -E(\frac{dl}{dV dk_e}) = E(-\frac{\Sigma_{i=1}^n x_i}{V\sigma^2}) = \frac{\Sigma_{i=1}^n x_i}{V\sigma^2}$

$I_{k_e \sigma^2} = I_{\sigma^2 k_e} = -E(\frac{dl}{d\sigma^2 dk_e}) = E(c\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)) = 0$ where c is constant.

$I_{V\sigma^2} = I_{\sigma^2 V} = -E(\frac{dl}{d\sigma^2 dV}) = E(c\Sigma_{i=1}^n (\log y_i - \log D + \log V + k_e x_i)) = 0$ where c is constant.

b

By solving the score functions,

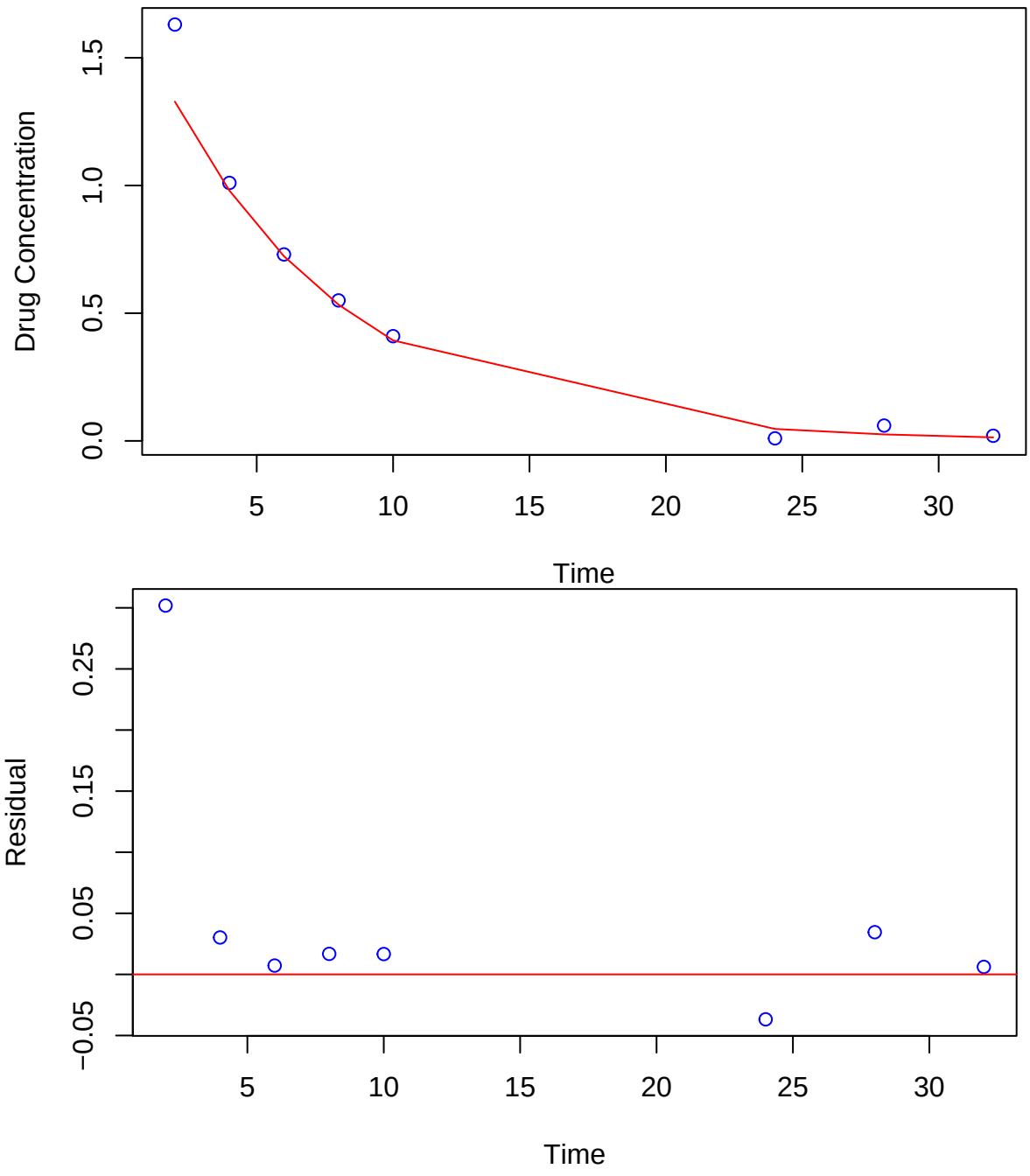
$\hat{k_e} = \frac{\Sigma_{i=1}^{n} \log y_i (x_i - \bar{x_i})}{n\bar{x}^2 - \Sigma_{i=1}^{n} x_i^2}$

$\hat{V} = \exp(\log D - \frac{1}{n}\Sigma_{i=1}^{n} \log y_i - \hat{k_e}\bar{x})$

$\hat{\sigma^2} = \frac{1}{n}\Sigma_{i=1}^{n}(\log y_i - \log D + \log \hat{V} + \hat{k_e} x_i)^2$

```
##                MLE    Lower CI   Upper CI
## Ke       0.1521064 0.111911096  0.1923016
## V       16.6633094 4.579285425 28.7473333
## Sigma^2  0.4119627 0.008246673  0.8156787
```

c



d

From the plots above, the assumptions are not entirely upheld, since there is one point being outlier, creating a large residual among all points. The other points seem to have good prediction though. The model is reasonable if we do not encounter extreme values.

e

By invariance of MLE and delta method on FIM, we can obtain the corresponding values of interest.

```
##          [,1]      [,2]      [,3]
```

```
## [1,] 2.534595 1.162621 3.90657
## [2,] 4.556990 3.352771 5.76121
```

Appendix

Q1

```
hodgkins <- c(396, 568, 1212, 171, 554, 1104, 257, 435, 295, 397, 288, 1004, 431,
              795, 1621, 1378, 902, 958, 1283, 2415)
nonhodgkins <- c(375, 375, 752, 208, 151, 116, 736, 192, 315, 1252, 675, 700, 440,
                 771, 688, 426, 410, 979, 377, 503)

res_1 <- rbind(summary(hodgkins),summary(nonhodgkins))
rownames(res_1) <- c("hodgkins", "nonhodgkins")
res_1

boxplot(hodgkins, nonhodgkins, names=c("hodgkins", "nonhodgkins"), main = "Comparison of hodgkins and no

library(broom)
x <- c(rep(0,length(hodgkins)), rep(1,length(nonhodgkins)))
y <- c(hodgkins, nonhodgkins)

lm1 <- lm(y ~ x)
lm2 <- lm(log(y) ~ x)
lm3 <- lm(sqrt(y) ~ x)
res1_b <- tidy(lm1, conf.int = T, conf.level = 0.90)[2,c(2,6,7)]
res2_b <- tidy(lm2, conf.int = T, conf.level = 0.90)[2,c(2,6,7)]
res3_b <- tidy(lm3, conf.int = T, conf.level = 0.90)[2,c(2,6,7)]

res_b <- rbind(res1_b, res2_b, res3_b)
rownames(res_b) <- c("Original", "Log", "Square Root")
res_b

model_pois <- glm(y ~ x, family = poisson(link = "log"))
model_gamma <- glm(y ~ x, family = Gamma(link = "inverse"))
model_inv_gauss <- glm(y ~ x, family=inverse.gaussian(link = "1/mu^2"))

model_pois <- c(model_pois$coefficients , model_pois$coefficients + qnorm(0.05)*sqrt(diag(vcov(model_poi
  model_pois$coefficients + qnorm(0.95)*sqrt(diag(vcov(model_pois))) )

model_gamma <- c(model_gamma$coefficients , model_gamma$coefficients + qnorm(0.05)*sqrt(diag(vcov(model_
  model_gamma$coefficients + qnorm(0.95)*sqrt(diag(vcov(model_gamma))) )

model_inv_gauss <- c(model_inv_gauss$coefficients , model_inv_gauss$coefficients + qnorm(0.05)*sqrt(diag
  model_inv_gauss$coefficients + qnorm(0.95)*sqrt(diag(vcov(model_inv_gauss))) )

res_d <- rbind(model_pois, model_gamma, model_inv_gauss)
colnames(res_d) <- c("b0", "b1", "b0 Lower CI", "b1 Upper CI", "b0 Lower CI", "b1 Upper CI")
res_d
```

Q2

```
y <- c(1.63,1.01,0.73,0.55,0.41,0.01,0.06,0.02)
log_y <- log(y)
```

```r
n <- length(y)
x <- c(2,4,6,8,10,24,28,32)
D <- 30

mle_ke <- sum(log_y*(x-mean(x))) / (n*mean(x)^2-sum(x^2))
mle_V <- exp(log(D)-mean(log_y)-mle_ke*mean(x))
mle_sig2 <- sum((log_y-log(D)+log(mle_V)+mle_ke*x)^2)/n


FIM <- matrix(c(sum(x^2)/mle_sig2, sum(x)/(mle_V*mle_sig2), 0,
                sum(x)/(mle_V*mle_sig2), n/(mle_V^2*mle_sig2), 0,
                0, 0, n/(2*mle_sig2^2)),nrow = 3, ncol = 3)

mle_se <- sqrt(diag(solve(FIM)))

res_b <- c(mle_ke, mle_V, mle_sig2) + as.vector(mle_se) %o% c(0,-1.96,1.96)
colnames(res_b) <- c("MLE", "Lower CI", "Upper CI")
rownames(res_b) <- c("Ke", "V", "Sigma^2")
res_b

yhat = exp(log(D/mle_V*exp(-mle_ke*x)))
plot(y ~ x, col="blue", xlab = "Time", ylab = "Drug Concentration")
lines(yhat ~ x, col="red" )

plot(y-yhat ~ x, col="blue", xlab = "Time", ylab = "Residual")
abline(h = 0, col="red")

Sigma <- solve(FIM[1:2,1:2])
grad_1 <- c(mle_V, mle_ke)
grad_2 <- c(log(2)/mle_ke^2,0)

ci1 <- (mle_V * mle_ke) + sqrt(t(grad_1)%*%Sigma%*%grad_1)[1] %o% c(0,qnorm(0.025),qnorm(0.975))
ci2 <- log(2)/mle_ke + sqrt(t(grad_2)%*%Sigma%*%grad_2)[1] %o% c(0,qnorm(0.025),qnorm(0.975))
res_e <- rbind(ci1, ci2)
res_e
```