

Recommended Exercise 7 in Statistical Linear Models, Spring 2021

alexao

13 mai, 2021

Problem 1 Inference about a new observation in multiple linear regression

a)

Since $\hat{\beta} = (X^T X)^{-1} X^T \mathbf{Y}$, we know that $\hat{\beta} \sim N(\beta, \sigma^2 (X^T X)^{-1})$. Since the Gaussian distribution is closed under linear transformations, this means that $\mathbf{x}_0^T \hat{\beta} \sim N(\mathbf{x}_0^T \beta, \sigma^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0)$, which is univariate. Now, since $E \mathbf{x}_0^T \hat{\beta} = \mathbf{x}_0^T \beta$, $\mathbf{x}_0^T \hat{\beta}$ is an unbiased estimator of $EY_0 = \mathbf{x}_0^T \beta$, with distribution as given above.

b)

Generally, we know that if $Z \sim N(0, 1)$ and $U \sim \chi_q^2$, where Z and U are independent, then $\frac{Z}{\sqrt{U/q}} \sim t_q$. In this case, we can set $Z = \frac{\mathbf{x}_0^T \hat{\beta} - \mathbf{x}_0^T \beta}{\sqrt{\sigma^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}}$ and $U = \frac{1}{\sigma^2} \text{SSE} = \frac{(n-p)\hat{\sigma}^2}{\sigma^2} \sim \chi_{n-p}^2$. Moreover, Z and U are independent, because $\hat{\beta}$ and $\hat{\sigma}^2$ are independent. Hence, the estimator

$$T = \frac{Z}{\sqrt{U/q}} = \frac{\frac{\mathbf{x}_0^T \hat{\beta} - \mathbf{x}_0^T \beta}{\sqrt{\sigma^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}}}{\sqrt{\frac{(n-p)\hat{\sigma}^2}{\sigma^2} / (n-p)}} = \frac{\mathbf{x}_0^T \hat{\beta} - \mathbf{x}_0^T \beta}{\sqrt{\hat{\sigma}^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}} \sim t_{n-p}.$$

This means that

$$\begin{aligned} -t &\leq \frac{\mathbf{x}_0^T \hat{\beta} - \mathbf{x}_0^T \beta}{\sqrt{\hat{\sigma}^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}} \leq t, \\ \iff \mathbf{x}_0^T \hat{\beta} - t\hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} &\leq \mathbf{x}_0^T \beta \leq \mathbf{x}_0^T \hat{\beta} + t\hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}, \\ \iff \mathbf{x}_0^T \hat{\beta} - t\hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} &\leq EY_0 \leq \mathbf{x}_0^T \hat{\beta} + t\hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}. \end{aligned}$$

Hence,

$$\begin{aligned} 1 - \alpha &= P \left(-t_{\alpha/2} \leq \frac{\mathbf{x}_0^T \hat{\beta} - \mathbf{x}_0^T \beta}{\sqrt{\hat{\sigma}^2 \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}} \leq t_{\alpha/2} \right) \\ &= P \left(\mathbf{x}_0^T \hat{\beta} - t_{\alpha/2} \hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} \leq EY_0 \leq \mathbf{x}_0^T \hat{\beta} + t_{\alpha/2} \hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} \right), \end{aligned}$$

which means that $\mathbf{x}_0^T \hat{\beta} \pm t_{\alpha/2} \hat{\sigma} \sqrt{\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}$ is a $100(1 - \alpha)\%$ confidence interval for EY_0 .

c)

A similar process to the one used in b) is used. For a future observation Y_0 we can derive that $Y_0 - \mathbf{x}_0^T \hat{\beta} \sim N(0, \sigma^2(1 + \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0))$, since $Y_0 = \mathbf{x}_0^T \beta + \varepsilon_0$ and $\mathbf{x}_0^T \hat{\beta}$ is an unbiased estimator of EY_0 . Set $Z = \frac{Y_0 - \mathbf{x}_0^T \hat{\beta}}{\sigma \sqrt{1 + \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}}$ and set U as in b). Solving the same inequalities as in b) then gives

$$1 - \alpha = P \left(\mathbf{x}_0^T \hat{\beta} - t_{\alpha/2} \hat{\sigma} \sqrt{1 + \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} \leq Y_0 \leq \mathbf{x}_0^T \hat{\beta} + t_{\alpha/2} \hat{\sigma} \sqrt{1 + \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0} \right),$$

which means that $\mathbf{x}_0^T \hat{\beta} \pm t_{\alpha/2} \hat{\sigma} \sqrt{1 + \mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0}$ is a $100(1 - \alpha)\%$ prediction interval for Y_0 . **LF: Why does Y_0 independent of ε give independence from $\mathbf{x}_0^T \hat{\beta}$?**

d)

```
acidrain <- read.table("https://www.math.ntnu.no/emner/TMA4267/2018v/acidrain.txt", header=TRUE)
fit <- lm(y~., data = acidrain)

# Confidence interval: Done manually via the equations found in previous tasks.
X <- model.matrix(fit)
x0 <- c(1, 3, 50, 1, 50, 2, 1, 0)
n <- dim(X)[1]
p <- dim(X)[2]
quantile <- qt(0.025, n-p, lower.tail = FALSE)
prediction <- t(x0) %*% fit$coefficients
upper.conf <- prediction + quantile * summary(fit)$sigma * sqrt(t(x0) %*% solve(t(X) %*% X) %*% x0)
lower.conf <- prediction - quantile * summary(fit)$sigma * sqrt(t(x0) %*% solve(t(X) %*% X) %*% x0)
prediction

#>           [,1]
#> [1,] 5.531684

lower.conf

#>           [,1]
#> [1,] 5.446329

upper.conf

#>           [,1]
#> [1,] 5.617039

# Confidence interval: Done in R.
newdata <- data.frame(x1=3, x2=50, x3=1, x4=50, x5=2, x6=1, x7=0)
predict(fit, newdata, level=.95, interval="confidence")

#>           fit      lwr      upr
#> 1 5.531684 5.446329 5.617039

# Prediction interval: Done manually via the equations found in previous tasks.
upper.pred <- prediction + quantile * summary(fit)$sigma * sqrt(1 + t(x0) %*% solve(t(X) %*% X) %*% x0)
lower.pred <- prediction - quantile * summary(fit)$sigma * sqrt(1 + t(x0) %*% solve(t(X) %*% X) %*% x0)
prediction

#>           [,1]
#> [1,] 5.531684
```

```

lower.pred

#>           [,1]
#> [1,] 5.272555

upper.pred

#>           [,1]
#> [1,] 5.790813

# Prediction interval: Done in R.
predict(fit, newdata, level=.95, interval="prediction")

#>           fit           lwr           upr
#> 1 5.531684 5.272555 5.790813

```

e)

Calculate $\mathbf{x}_0^T (X^T X)^{-1} \mathbf{x}_0$ with $\mathbf{x}_0^T = (1 \ x_0)$ and

$$X^T = \begin{pmatrix} 1 & 1 & 1 & \dots & \dots & 1 \\ x_1 & x_2 & x_3 & \dots & \dots & x_n \end{pmatrix}.$$

This, combined with the rest of the expression for the confidence interval from b), gives the desired expression for simple linear regression.

Problem 2 Plant stress

a)

- t-value for intercept is missing: This is calculated by the test statistic $\hat{\beta}_{\text{intercept}} / \hat{SE} \approx 16.15942 / 0.04140 \approx 390.3242$. This is the test statistic for testing $H_0 : \hat{\beta}_{\text{intercept}} = 0$ vs. $H_1 : \hat{\beta}_{\text{intercept}} \neq 0$.
- Std. Error for D:T is missing: This can be calculated from the same test statistic as in the last bullet point, since we know the t-value for D:T. Hence, $\hat{\beta}_{\text{D:T}} / \hat{SE} = t$ gives $\hat{SE} = \hat{\beta}_{\text{D:T}} / t \approx -0.00242 / -0.058 \approx 0.0417$. This is the estimated standard deviation of the estimation of the coefficient for D:T. Another way of calculating this value is by noting the design of the experiment. This is a two-level full factorial design, which is a case where we know that $\hat{\beta} \sim N(\beta, \sigma^2/n)$, which means that the estimated std. error for D:T can be found by **Residual standard error** divided by $\sqrt{32}$.
- p-value for D:F:T is missing: This can be calculated as $2P(T \geq t) = 2 * \text{pt}(2.198, 24, \text{lower.tail} = \text{F}) \approx 0.037836$. This is, loosely speaking, the probability that the t-statistic is as observed or more extreme. It is used to test $H_0 : \hat{\beta}_{\text{D:F:T}} = 0$ vs. $H_1 : \hat{\beta}_{\text{D:F:T}} \neq 0$.
- Multiple R-squared is missing: This can be calculated as $1 - \frac{\text{SSE}}{\text{SST}}$. SSE can be calculated from the formula $(n-p)\hat{\sigma}^2 = (32-8)\hat{\sigma}^2 = 24\hat{\sigma}^2 \approx 1.3163914$, where $\hat{\sigma}$ is found as **Residual Standard Error** in the summary-table. We also know the F-statistic, which is calculated as $\frac{\text{SSR}/(p-1)}{\text{SSE}/(n-p)} = \frac{(\text{SST}-\text{SSE})/(p-1)}{\text{SSE}/(n-p)} = \frac{(\text{SST}-\text{SSE})/(8-1)}{\text{SSE}/(32-8)} = \frac{(\text{SST}-\text{SSE})/7}{\text{SSE}/24} \approx 105.6$, which gives that $\text{SST} \approx 41.8612452$. Finally, this gives $R^2 \approx 0.9685535$.

b)

An estimator could be $\hat{\gamma} = 2^{\hat{\beta}_F - \hat{\beta}_D}$. The expected value and variance of this estimator can be calculated using a first order Taylor expansion. Setting $h(x, y) = 2^{x-y}$ and using a Taylor expansion in x and y of first order gives

$$\begin{aligned} h(\hat{\beta}_F, \hat{\beta}_D) &\approx h(\beta_F, \beta_D) + h_{\hat{\beta}_F}(\beta_F, \beta_D)(\beta_F - \hat{\beta}_F) + h_{\hat{\beta}_D}(\beta_F, \beta_D)(\beta_D - \hat{\beta}_D) \\ &= 2^{\beta_F - \beta_D} + 2^{\beta_F - \beta_D}(\beta_F - \hat{\beta}_F)\ln(2) + 2^{\beta_F - \beta_D}(\beta_D - \hat{\beta}_D)\ln(2), \end{aligned}$$

since $h_x(x, y) = \frac{\partial}{\partial x} h(x, y) = \frac{\partial}{\partial x} 2^{x-y} = \frac{\partial}{\partial x} \exp((x-y)\ln(2)) = 2^{x-y}\ln(2)$. This means that $E(\hat{\gamma}) = E(2^{\hat{\beta}_F - \hat{\beta}_D}) = E(h(\hat{\beta}_F, \hat{\beta}_D)) \approx h(\beta_F, \beta_D) = 2^{\beta_F - \beta_D}$, since $\hat{\beta}_F$ and $\hat{\beta}_D$ are unbiased estimators of β_F and β_D . Moreover, $\text{Var}(\hat{\gamma}) = \text{Var}(2^{\hat{\beta}_F - \hat{\beta}_D}) = \text{Var}(h(\hat{\beta}_F, \hat{\beta}_D)) \approx h_{\hat{\beta}_F}(\beta_F, \beta_D)^2 \text{Var}(\hat{\beta}_F) + h_{\hat{\beta}_D}(\beta_F, \beta_D)^2 \text{Var}(\hat{\beta}_D) = 2^{2(\beta_F - \beta_D)}(\ln(2))^2(\text{Var}(\hat{\beta}_F) - \text{Var}(\hat{\beta}_D))$.

From Figure 1, we can get the numerical estimates of the moments, given as

$$\begin{aligned} \widehat{E(\hat{\gamma})} &= 2^{\hat{\beta}_F - \hat{\beta}_D} \approx 0.636 \\ \widehat{\text{Var}(\hat{\gamma})} &= 2^{2(\hat{\beta}_F - \hat{\beta}_D)}(\ln(2))^2(\text{Var}(\hat{\beta}_F) - \text{Var}(\hat{\beta}_D)) \approx 6.67 \cdot 10^{-4} \end{aligned}$$

c)

A general test is $C\beta = d$. Let

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad d = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}.$$

The test statistic is

$$F = \frac{(C\hat{\beta})^T (C(X^T X)^{-1} C^T)^{-1} C\hat{\beta}}{r\hat{\sigma}^2} = 4.705 \sim F_{\text{rank}(C), n-p} = F_{3, 24}.$$

The critical values for $F_{3, 24}$ are 3.01 for $\alpha = 0.05$ and 3.72 for $\alpha = 0.025$. Thus, the null hypothesis is rejected at (least) level 0.025.

d)

The estimated regression coefficients are the same, since the columns in X are orthogonal (each estimate only depends on the response and its respective column in X). However, the estimated standard deviation changes because it is estimated from SSE and $(n - p)$, which are quantities that change when the model changes. A prediction is $\mathbf{x}_0^T \hat{\beta} = (1 \ 1 \ 1 \ -1 \ 1)^T \hat{\beta} \approx 17.82$, with a 95% prediction interval (calculated via the formula in c)) $[17.198, 18.425]$.

Problem 3 Multiple testing with plant stress

```
pvalues <- scan("https://www.math.ntnu.no/emner/TMA4267/2018v/damagePvalues.txt")
m <- length(pvalues)
```

a)

The family-wise error rate (FWER) is the probability of one or more false positive findings $P(V > 0)$, where V is the number of false positive findings among the m tests. The false discovery rate (FDR) gives the expected proportion of false positive findings among the rejected hypotheses. A false positive is the same as a type I error, a case where the null hypothesis is rejected despite it being true. In the case of multiple linear regression this means that it is concluded that there is a linear relationship between the predictors and the response, despite the relationship not existing.

b)

Control FWER at level $\alpha = 0.05$ with the Bonferroni method. This gives $\alpha_{\text{loc}} = \frac{0.05}{m} = 5 \times 10^{-6}$. In this case one would reject `length(pvalues[pvalues<5e-6]) = 19` null-hypotheses in the data. Bonferroni's method can always be used. It is often called conservative because controlling FWER is a very strict criterion, especially when m is large, as is the case here. Moreover, since Bonferroni's method does not account for dependency structures between the m hypotheses, the p -value cut-off might be estimated to a value that is much smaller than necessary, e.g. if the tests (here: genes) are highly correlated, which means that the effective number of tests is smaller than m . More elaborate methods take dependencies into account, which might yield more accurate cut-offs.

c)

```
pvaluesb <- ifelse(pvalues<5e-6, "Reject", "Keep")
type1c <- sum(pvaluesb[1:9000] == "Reject")
type2c <- sum(pvaluesb[9001:10000] == "Keep")
type1c # V. Hence, U = 9000 - V = 9000
```

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

```
type2c # T. Hence, S = 1000 - 981 = 19
```

```
#> [1] 981
```

```
# Hence, R = 19, m-R = 9981
```

Thus, there are zero false positives. In a true multiple hypothesis setting the only known values are m and R .

d)

```
pvaluesd <- ifelse(pvalues<0.05, "Reject", "Keep")
type1d <- sum(pvaluesd[1:9000] == "Reject")
type2d <- sum(pvaluesd[9001:10000] == "Keep")
type1d # V. Hence, U = 9000 - V = 8572
```

```
#> [1] 428
```

```
type2d # T. Hence, S = 1000 - 178 = 822
```

```
#> [1] 178
```

```
# Hence, R = V + S = 1250, m-R = 8750
```

Now we have 428 false positives.