Notes 11

Maryclare Griffin

4/6/2023

These notes are based on Chapters 2 and 6 of KNNL.

We will continue to assume the **normal error linear regression model** for a dependent variable or response Y and independent variables, predictors, or covariates $X_1, \dots X_{p-1}$ is defined as:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where:

- The elements of $\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{pmatrix}$ are parameters
- The elements of the $n \times p$ matrix $\boldsymbol{X} = \begin{pmatrix} 1 & X_{11} & X_{12} & \dots & X_{1,p-1} \\ 1 & X_{21} & X_{22} & \dots & X_{2,p-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & X_{n2} & \dots & X_{n,p-1} \end{pmatrix}$ are known constants
- $\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$ is a random error term elements that are ϵ_i that are independent and normally distributed with mean $E\left\{\epsilon_i\right\} = 0$ and variance $\sigma^2\left\{\epsilon_i\right\} = \sigma^2$.

Suppose that instead, we want to talk about the distribution of our point estimate of the mean response at when the predictors are equal to $X_{h1}, \dots, X_{h,p-1}$? Let $\boldsymbol{X}_h = \begin{pmatrix} 1 & X_{h1} & \dots & X_{h,p-1} \end{pmatrix}$ refer to the $1 \times p$ vector of predictor values. The point estimate of the mean response is $\hat{Y}_h = \boldsymbol{X}_h\boldsymbol{b}$. Using the same logic we used to describe the sampling distribution of b_k and $\frac{b_k - \beta_k}{s\left\{b_k\right\}}$, we can describe the sampling distribution of $\hat{Y}_h$ and $\frac{\hat{Y}_h - E\left\{\hat{Y}_h\right\}}{s\left\{\hat{Y}_h\right\}}$.

Under the normal errors regression model, $\hat{Y}_h$ is a normal random variable with mean $E\left\{\hat{Y}_h\right\} = \boldsymbol{X}_h\boldsymbol{\beta}$ and variance $\sigma^2\left\{\hat{Y}_h\right\} = \sigma^2\boldsymbol{X}_h\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}_h'$.

Under the normal errors regression model,

$$\frac{\hat{Y}_h - E\left\{\hat{Y}_h\right\}}{s\left\{\hat{Y}_h\right\}}.$$

is t distributed with $n - p$ degrees of freedom.

This allows us to construct $(1-\alpha)\times 100\%$ confidence intervals for the mean response for any value of the predictor variables. The limits of a $(1-\alpha)\times 100\%$ confidence interval for $E\left\{\hat{Y}_h\right\}$ are $\hat{Y}_h \pm t\left(\alpha/2; n-p\right) s\left\{\hat{Y}_h\right\}$.

> ***Example 1:*** Again, consider data from a company that manufactures refrigeration equipment, called the Toluca company. They produce refrigerator parts in lots of different sizes, and the amount of time it takes to produce a lot of refrigerator parts depends on the number of parts in the lot and several other variable factors. Let X be the number of refrigerator plots in a lot, and let Y refer to the amount of time it takes to produce a size of lot X. Suppose a cost analyst in the Toluca Company is interested finding a 90% confidence interval for $E\left\{Y_h\right\}$ when the lot size $X_h = 65$ units.

```
load("~/Dropbox/Teaching/STAT525/Spring2023/bookdata/toluca.RData")
n <- nrow(data) # Extract number of observations
Y <- data$Y # Extract response
X <- data$X # Extract predictor
linmod <- lm(Y~X) # Fit linear model
pred <- predict(linmod, newdata = data.frame("X" = 65),
                se.fit = TRUE)
Y.hat.h <- pred$fit
s.Y.hat.h <- pred$se.fit
alpha <- 0.1
lower <- Y.hat.h + qt(alpha/2, n - 2)*s.Y.hat.h
upper <- Y.hat.h - qt(alpha/2, n - 2)*s.Y.hat.h
```

> We obtain a 90% confidence interval of $(277.432, 311.426)$ for $E\left\{\hat{Y}_h\right\}$.

Now, suppose that we want to want to talk about of a new observation $Y_{h(new)}$ when the predictors are equal to $X_{h1}, \dots, X_{h,p-1}$. Again, let $\boldsymbol{X}_h = \begin{pmatrix} 1 & X_{h1} & \dots & X_{h,p-1} \end{pmatrix}$ refer to the $1\times p$ vector of predictor values. If we knew $\boldsymbol{\beta}$ and σ^2, this would be simple! The distribution of a new observation $Y_{h(new)}$ would be normal, with mean $E\left\{Y_{h(new)}\right\} = \boldsymbol{X}_h\boldsymbol{\beta}$ and variance $\sigma^2\left\{Y_{h(new)}\right\} = \sigma^2$. In practice, we don't know $\boldsymbol{\beta}$ or σ^2.

To address this, we consider the distribution of $Y_{h(new)} - \hat{Y}_h$, our new observation minus the estimated mean $\hat{Y}_h = \boldsymbol{X}_h\boldsymbol{b}$. Because $Y_{h(new)}$ is a normal random variable and $\hat{Y}_h$ is a linear combination of normal random variables, $Y_{h(new)} - \hat{Y}_h$ is also normal with mean

$$
\begin{aligned}
E\left\{Y_{h(new)} - \hat{Y}_h\right\} &= E\left\{Y_{h(new)}\right\} - E\left\{\hat{Y}_h\right\} \\
&= \boldsymbol{X}_h\boldsymbol{\beta} - \boldsymbol{X}_h\boldsymbol{\beta} \\
&= 0
\end{aligned}
$$

and variance

$$
\begin{aligned}
\sigma^2\left\{Y_{h(new)} - \hat{Y}_h\right\} &= \sigma^2\left\{Y_{h(new)}\right\} - 2\sigma\left\{Y_{h(new)}, \hat{Y}_h\right\} + \sigma^2\left\{\hat{Y}_h\right\} \\
&= \sigma^2 - 2\sigma\left\{Y_{h(new)}, \hat{Y}_h\right\} + \sigma^2\boldsymbol{X}_h\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}_h \\
&= \sigma^2 + \sigma^2\boldsymbol{X}_h\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}_h \\
&= \sigma^2\left(1 + \boldsymbol{X}_h\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}_h\right).
\end{aligned}
$$

Note that the variance of $Y_{h(new)} - \hat{Y}_h$ has two distinct components:

- The variability of a new observation $Y_{h(new)}$ if the mean $E\left\{Y_{h(new)}\right\}$ were known
- The variability of our estimate $\hat{Y}_h$ of the mean $E\left\{Y_{h(new)}\right\}$

The corresponding unbiased estimator of $\sigma^2\left\{Y_{h(new)} - \hat{Y}_h\right\}$ is, $s^2\left\{Y_{h(new)} - \hat{Y}_h\right\} = s^2\left(1 + \boldsymbol{X}_h\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}_h\right)$

which is obtained by plugging s^2 in for σ^2. We will refer to $\sigma^2\left\{Y_{h(new)} - \hat{Y}_h\right\}$ and $s^2\left\{Y_{h(new)} - \hat{Y}_h\right\}$ as $\sigma^2\{\text{pred}\}$ and $s^2\{\text{pred}\}$, respectively.

It follows that the studentized statistic $\frac{Y_{h(new)} - \hat{Y}_h}{s\{\text{pred}\}} \sim t_{n-2}$ under the normal error linear regression model. By the same logic we have used to obtain $(1-\alpha) \times 100\%$ confidence intervals for $\boldsymbol{b}$ and $\hat{Y}_h$, we can obtain a $(1-\alpha) \times 100\%$ **prediction interval** for $Y_{h(new)}$. The limits of a $(1-\alpha) \times 100\%$ prediction interval for $\hat{Y}_{h(new)}$ are $\hat{Y}_h \pm t(\alpha/2; n-p)\, s\{pred\}$. It is called a **prediction interval** because it will contain a single future value $Y_{h(new)}$ with probability $(1-\alpha) \times 100\%$ under the normal error linear regression model.

Example 2: Again, consider data from a company that manufactures refrigeration equipment, called the Toluca company. Suppose a cost analyst in the Toluca Company is interested finding a 90% prediction interval for $E\{Y_h\}$ when the lot size $X_h = 100$ units.

```
pred <- predict(linmod, newdata = data.frame("X" = 100),
                se.fit = TRUE)
Y.hat.h <- pred$fit
s <- pred$residual.scale
s.pred <- sqrt(pred$se.fit^2 + s^2)
alpha <- 0.1
lower <- Y.hat.h + qt(alpha/2, n - 2)*s.pred
upper <- Y.hat.h - qt(alpha/2, n - 2)*s.pred
```

We obtain a 90% confidence interval of $(332.207, 506.565)$ for $\hat{Y}_{h(new)}$.

Confidence and prediction intervals can be computed for many predictor values simultaneously. This allows us to construct very useful visualizations of model fit by plotting the fitted regression line with $1-\alpha$ confidence and prediction intervals at each possible predictor value.

Example 3: Again, consider data from a company that manufactures refrigeration equipment, called the Toluca company. Suppose a cost analyst wants to visualize the confidence intervals and prediction intervals for a range of possible lot sizes.

```
plot(X, Y, pch = 16, xlab = "Lot Size",
     ylab = "Work Hours")
Xvals <- seq(10, 130, length.out = 1000)
conf <- predict(linmod, newdata = data.frame("X" = Xvals),
                interval = "confidence", level = 1 - alpha)
pred <- predict(linmod, newdata = data.frame("X" = Xvals),
                interval = "prediction", level = 1 - alpha)
fitted <- conf[, "fit"]
lower.conf <- conf[, "lwr"]
upper.conf <- conf[, "upr"]
lower.pred <- pred[, "lwr"]
upper.pred <- pred[, "upr"]
lines(Xvals, fitted)
polygon(c(Xvals, rev(Xvals)),
        c(lower.conf, rev(upper.conf)),
        col = rgb(0, 0, 1, 0.25),
        border = NA)
polygon(c(Xvals, rev(Xvals)),
        c(lower.pred, rev(upper.pred)),
        col = rgb(1, 0, 0, 0.25),
        border = NA)
legend("topleft", fill = c(rgb(0, 0, 1, 0.25),
                           rgb(1, 0, 0, 0.25)),
       legend = c("90% Confidence", "90% Prediction"),
```

```
      title = "Interval Type", border = NA)
```

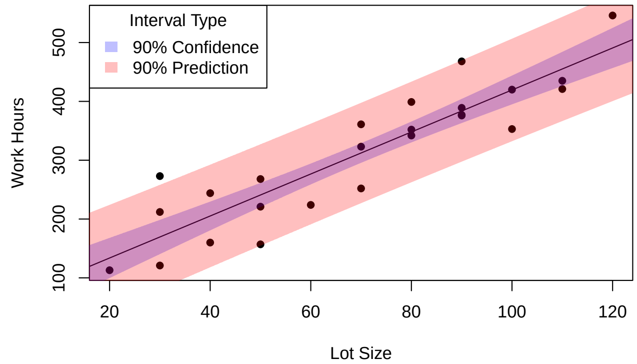

Figure 1: Example 3