Statistical linear regression models

Brian Caffo, Jeff Leek, Roger Peng

May 19, 2016

Basic regression model with additive Gaussian errors.

- Least squares is an estimation tool, how do we do inference?
- Consider developing a probabilistic model for linear regression

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

- Here the ϵ_i are assumed iid $N(0, \sigma^2)$.
- Note, $E[Y_i \mid X_i = x_i] = \mu_i = \beta_0 + \beta_1 x_i$
- Note, $Var(Y_i \mid X_i = x_i) = \sigma^2$.

Recap

- Model $Y_i = \mu_i + \epsilon_i = \beta_0 + \beta_1 X_i + \epsilon_i$ where ϵ_i are iid $N(0, \sigma^2)$
- ML estimates of β_0 and β_1 are the least squares estimates

$$\hat{\beta}_1 = Cor(Y, X)\frac{Sd(Y)}{Sd(X)} \quad \hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

- $E[Y \mid X = x] = \beta_0 + \beta_1 x$
- $Var(Y \mid X = x) = \sigma^2$

Interpretting regression coefficients, the itc

- β_0 is the expected value of the response when the predictor is 0

$$E[Y|X = 0] = \beta_0 + \beta_1 \times 0 = \beta_0$$

- Note, this isn't always of interest, for example when $X = 0$ is impossible or far outside of the range of data. (X is blood pressure, or height etc.)
- Consider that

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \beta_0 + a\beta_1 + \beta_1 (X_i - a) + \epsilon_i = \tilde{\beta}_0 + \beta_1 (X_i - a) + \epsilon_i$$

  So, shifting your X values by value a changes the intercept, but not the slope.
- Often a is set to $\bar{X}$ so that the intercept is interpretted as the expected response at the average X value.

Interpretting regression coefficients, the slope

- β_1 is the expected change in response for a 1 unit change in the predictor

$$E[Y \mid X = x+1] - E[Y \mid X = x] = \beta_0 + \beta_1 (x+1) - (\beta_0 + \beta_1 x) = \beta_1$$

- Consider the impact of changing the units of X.

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i = \beta_0 + \frac{\beta_1}{a}(X_i a) + \epsilon_i = \beta_0 + \tilde{\beta}_1 (X_i a) + \epsilon_i$$

- Therefore, multiplication of X by a factor a results in dividing the coefficient by a factor of a.
- Example: X is height in m and Y is weight in kg. Then β_1 is kg/m. Converting X to cm implies multiplying X by $100 cm/m$. To get β_1 in the right units, we have to divide by $100 cm/m$ to get it to have the right units.

$$Xm \times \frac{100cm}{m} = (100X)cm \text{ and } \beta_1 \frac{kg}{m} \times \frac{1m}{100cm} = \left(\frac{\beta_1}{100}\right)\frac{kg}{cm}$$

Using regression coeficients for prediction

- If we would like to guess the outcome at a particular value of the predictor, say X, the regression model guesses

$$\hat{\beta}_0 + \hat{\beta}_1 X$$

Example

`diamond` data set from `UsingR`

Data is diamond prices (Singapore dollars) and diamond weight in carats (standard measure of diamond mass, 0.2 g). To get the data use `library(UsingR); data(diamond)`

Plot of the data

```
## Loading required package: MASS

## Loading required package: HistData

## Loading required package: Hmisc

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## Loading required package: ggplot2

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
```

Fitting the linear regression model

```r
fit <- lm(price ~ carat, data = diamond)
coef(fit)
```

```
## (Intercept)       carat
##   -259.6259   3721.0249
```

- We estimate an expected 3721.02 (SIN) dollar increase in price for every carat increase in mass of diamond.
- The intercept -259.63 is the expected price of a 0 carat diamond.

Getting a more interpretable intercept

```r
fit2 <- lm(price ~ I(carat - mean(carat)), data = diamond)
coef(fit2)
```

```
##            (Intercept) I(carat - mean(carat))
##               500.0833              3721.0249
```

Thus \$500.1 is the expected price for the average sized diamond of the data (0.2041667 carats).

Changing scale

- A one carat increase in a diamond is pretty big, what about changing units to 1/10th of a carat?
- We can just do this by just dividing the coeficient by 10.
- We expect a 372.102 (SIN) dollar change in price for every 1/10th of a carat increase in mass of diamond.
- Showing that it's the same if we rescale the Xs and refit

```
fit3 <- lm(price ~ I(carat * 10), data = diamond)
coef(fit3)
```

```
##   (Intercept) I(carat * 10)
##     -259.6259      372.1025
```

Predicting the price of a diamond

```
newx <- c(0.16, 0.27, 0.34)
coef(fit)[1] + coef(fit)[2] * newx
```

```
## [1]  335.7381  745.0508 1005.5225
```

```
predict(fit, newdata = data.frame(carat = newx))
```

```
##         1         2         3
##  335.7381  745.0508 1005.5225
```

Predicted values at the observed Xs (red) and at the new Xs (lines)