Appendix E

Regression modeling

Regression analysis is a powerful and flexible framework that allows an analyst to model an outcome (the *response variable*) as a function of one or more *explanatory variables* (or predictors). Regression forms the basis of many important statistical models described in Chapters 7 and 8. This appendix provides a brief review of linear and logistic *regression models*, beginning with a single predictor, then extending to multiple predictors.

E.1 Simple linear regression

Linear regression can help us understand how values of a quantitative (numerical) outcome (or response) are associated with values of a quantitative explanatory (or predictor) variable. This technique is often applied in two ways: to generate predicted values or to make inferences regarding associations in the dataset.

In some disciplines the outcome is called the dependent variable and the predictor the independent variable. We avoid such usage since the words dependent and independent have many meanings in statistics.

A simple linear regression model for an outcome y as a function of a predictor x takes the form:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i\,, \text{ for } i = 1, \ldots, n\,,$$

where n represents the number of observations (rows) in the data set. For this model, β_0 is the population parameter corresponding to the *intercept* (i.e., the predicted value when $x = 0$) and β_1 is the true (population) *slope* coefficient (i.e., the predicted increase in y for a unit increase in x). The ϵ_i's are the *errors* (these are assumed to be random noise with mean 0).

We almost never know the true values of the population parameters β_0 and β_1, but we estimate them using data from our sample. The `lm()` function finds the "best" coefficients $\hat{\beta}_0$ and $\hat{\beta}_1$ where the the *fitted values* (or expected values) are given by $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$. What is left over is captured by the *residuals* ($\epsilon_i = y_i - \hat{y}_i$). The model almost never fits perfectly—if it did there would be no need for a model.

The best fitting regression line is usually determined by a *least squares* criteria that minimizes the sum of the squared residuals. The least squares regression line (defined by the values of $\hat{\beta}_0$ and $\hat{\beta}_1$) is unique.

E.1.1 Motivating example: Modeling usage of a rail trail

The Pioneer Valley Planning Commission (PVPC) collected data north of Chestnut Street in Florence, Massachusetts for a ninety day period. Data collectors set up a laser sensor that recorded when a rail-trail user passed the data collection station.

```
glimpse(RailTrail)

Observations: 90
Variables: 10
$ hightemp   <int> 83, 73, 74, 95, 44, 69, 66, 66, 80, 79, 78, 65, 41,...
$ lowtemp    <int> 50, 49, 52, 61, 52, 54, 39, 38, 55, 45, 55, 48, 49,...
$ avgtemp    <dbl> 66.5, 61.0, 63.0, 78.0, 48.0, 61.5, 52.5, 52.0, 67....
$ spring     <int> 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, ...
$ summer     <int> 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, ...
$ fall       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, ...
$ cloudcover <dbl> 7.6, 6.3, 7.5, 2.6, 10.0, 6.6, 2.4, 0.0, 3.8, 4.1, ...
$ precip     <dbl> 0.00, 0.29, 0.32, 0.00, 0.14, 0.02, 0.00, 0.00, 0.0...
$ volume     <int> 501, 419, 397, 385, 200, 375, 417, 629, 533, 547, 4...
$ weekday    <fctr> 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0,...
```

The PVPC wants to understand the relationship between daily ridership (i.e., the number of riders and walkers who use the bike path on any given day) and a collection of explanatory variables, including the temperature, rainfall, cloud cover, and day of the week.

In a simple linear regression model, there is a single quantitative explanatory variable. It seems reasonable that the high temperature for the day (`hightemp`, measured in degrees Fahrenheit) might be related to ridership, so we will explore that first. Figure E.1 shows a scatterplot between ridership (`volume`) and high temperature (`hightemp`), with the simple linear regression line overlaid. The fitted coefficients are shown below by providing a formula to the `lm()` function.

```
mod <- lm(volume ~ hightemp, data = RailTrail)
coef(mod)

(Intercept)    hightemp
    -17.079       5.702
```

The first coefficient is $\hat{\beta}_0$, the estimated y-intercept. The interpretation is that if the high temperature was 0 degrees Fahrenheit, then the estimated ridership would be about -17 riders. This is doubly non-sensical in this context, since it is impossible to have a negative number of riders and this represents a substantial extrapolation to far colder temperatures than are present in the data set (recall the *Challenger* discussion from Chapter 2). It turns out that the monitoring equipment didn't work when it got too cold, so values for those days are unavailable.

Pro Tip: In this case, it is not appropriate to simply multiply the average number of users on the observed days by the number of days in a year, since cold days that are likely to have fewer trail users are excluded due to instrumentation issues. Such missing data can lead to selection bias.

The second coefficient (the slope) is usually more interesting. This coefficient ($\hat{\beta}_1$) is interpreted as the predicted increase in trail users for each additional degree in temperature.

```
plotModel(mod, system = "ggplot2")
```

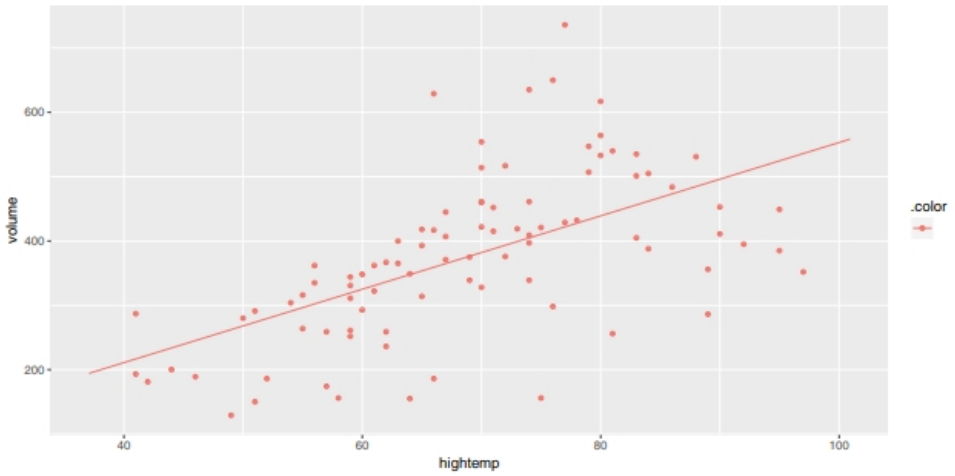

Figure E.1: Scatterplot of number of trail crossings as a function of highest daily temperature (in degrees Fahrenheit).

We expect to see about 5.7 additional riders use the rail trail on a day that is one degree warmer than another day.

E.1.2 Model visualization

Figure E.1 allows us to visualize our model in the data space. How does our model compare to a null model? That is, how do we know that our model is useful?

In Figure E.2, we compare the least squares regression line (right) with the null model that simply returns the average for every input (left). That is, on the left, the average temperature of the day is ignored. The model simply predicts an average ridership every day, regardless of the temperature. However, on the right, the model takes the average ridership into account, and accordingly makes a different prediction for each input value.

Obviously, the regression model works better than the null model (that forces the slope to be zero), since it is more flexible. But how much better?

E.1.3 Measuring the strength of fit

The correlation coefficient, r, is used to quantify the strength of the linear relationship between two variables. We can quantify the proportion of variation in the response variable (y) that is explained by the model in a similar fashion. This quantity is called the *coefficient of determination* and is denoted R^2. It is a common measure of goodness-of-fit for regression models. Like any proportion, R^2 is always between 0 and 1. For simple linear regression

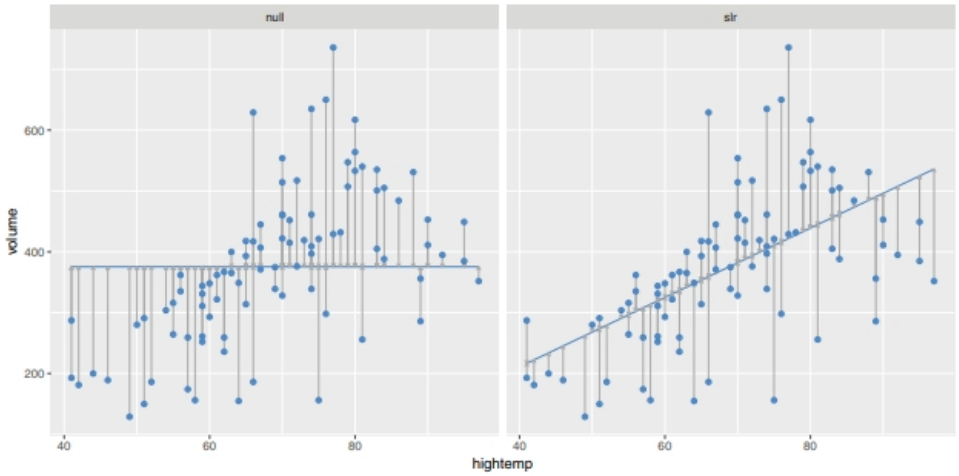

Figure E.2: At left, the model based on the overall average high temperature. At right, the simple linear regression model.

(one explanatory variable), $R^2 = r^2$. The definition of R^2 is given by:

$$\begin{aligned} R^2 &= 1 - \frac{SSE}{SST} = \frac{SSM}{SST} \\ &= 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \\ &= 1 - \frac{SSE}{(n-1)Var(y)}, \end{aligned}$$

where SSE is the sum of the squared residuals, SSM is the sum of the squares attributed to the model, and SST is the total sum of the squares. Let's calculate these values for the rail trail example.

```
n <- nrow(RailTrail)
SST <- var(~volume, data = RailTrail) * (n - 1)
SSE <- var(residuals(mod)) * (n - 1)
1 - SSE / SST
```

```
[1] 0.3394
```

```
rsquared(mod)
```

```
[1] 0.3394
```

In Figure E.2, the null model on the left has an R^2 of 0, because $\hat{y}_i = \bar{y}$ for all i, and so $SSE = SST$. On the other hand, the R^2 of the regression model on the right is 0.3394. We say that the regression model based on average daily temperature explained about 34% of the variation in daily ridership.

E.1.4 Categorical explanatory variables

Suppose that instead of using temperature as our explanatory variable for ridership on the rail trail, we only considered whether it was a weekday or not a weekday (e.g., weekend or holiday). The indicator variable `weekday` is *binary* (or dichotomous) in that it only takes on the values 0 and 1. (Such variables are sometimes called *indicator* variables or more pejoratively *dummy* variables.) This new linear regression model has the form:

$$\widehat{volume} = \hat{\beta}_0 + \hat{\beta}_1 \cdot weekday\,,$$

where the fitted coefficients are given below.

```
coef(lm(volume ~ weekday, data = RailTrail))

(Intercept)    weekday1
     430.71      -80.29
```

Note that these coefficients could have been calculated from the means of the two groups (since the regression model has only two possible predicted values). The average ridership on weekdays is 350.4 while the average on non-weekdays is 430.7.

```
mean(volume ~ weekday, data = RailTrail)

    0     1
430.7 350.4

diff(mean(volume ~ weekday, data = RailTrail))

     1
-80.29
```

In the coefficients listed above, the `weekday1` variable corresponds to rows in which the value of the `weekday` variable was `1` (i.e., weekdays). Because this value is negative, our interpretation is that 80 fewer riders are expected on a weekday as opposed to a weekend or holiday.

To improve the readability of the output we can create a new variable with more mnemonic values.

```
RailTrail <- RailTrail %>%
  mutate(day = ifelse(weekday == 1, "weekday", "weekend/holiday"))
```

Pro Tip: Care was needed to recode the `weekday` variable because it was a `factor`. Avoid the use of factors unless they are needed.

```
coef(lm(volume ~ day, data = RailTrail))

       (Intercept) dayweekend/holiday
            350.42              80.29
```

The model coefficients have changed (although they still provide the same interpretation). By default, the `lm()` function will pick the alphabetically lowest value of the categorical predictor as the *reference group* and create indicators for the other levels (in this

case dayweekend/holiday). As a result the intercept is now the predicted number of trail crossings on a weekday. In either formulation, the interpretation of the model remains the same: On a weekday, 80 fewer riders are expected than on a weekend or holiday.

E.2 Multiple regression

Multiple regression is a natural extension of simple linear regression that incorporates multiple explanatory (or predictor) variables. It has the general form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon, \text{ where } \epsilon \sim N(0, \sigma_\epsilon) .$$

The estimated coefficients (i.e., $\hat{\beta}_i$'s) are now interpreted as "conditional on" the other variables—each β_i reflects the *predicted* change in y associated with a one-unit increase in x_i, conditional upon the rest of the x_i's. This type of model can help to disentangle more complex relationships between three or more variables. The value of R^2 from a multiple regression model has the same interpretation as before: the proportion of variability explained by the model.

Pro Tip: Interpreting conditional regression parameters can be challenging. The analyst needs to ensure that comparisons that hold other factors constant do not involve extrapolations beyond the observed data.

E.2.1 Parallel slopes: Multiple regression with a categorical variable

Consider first the case where x_2 is an *indicator* variable that can only be 0 or 1 (e.g., weekday). Then,

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 .$$

In the case where x_1 is quantitative but x_2 is an indicator variable, we have:

$$\begin{aligned} \text{For weekends,} \quad & \hat{y}|_{x_1, x_2=0} = \hat{\beta}_0 + \hat{\beta}_1 x_1 \\ \text{For weekdays,} \quad & \hat{y}|_{x_1, x_2=1} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 \cdot 1 \\ & = \left(\hat{\beta}_0 + \hat{\beta}_2\right) + \hat{\beta}_1 x_1 . \end{aligned}$$

This is called a *parallel slopes* model (see Figure E.3), since the predicted values of the model take the geometric shape of two parallel lines with slope $\hat{\beta}_1$: one with y-intercept $\hat{\beta}_0$ for weekends, and another with y-intercept $\hat{\beta}_0 + \hat{\beta}_2$ for weekdays.

```
mod_parallel <- lm(volume ~ hightemp + weekday, data = RailTrail)
coef(mod_parallel)
```

```
(Intercept)    hightemp    weekday1
     42.807       5.348     -51.553
```

```
rsquared(mod_parallel)
```

```
[1] 0.3735
```

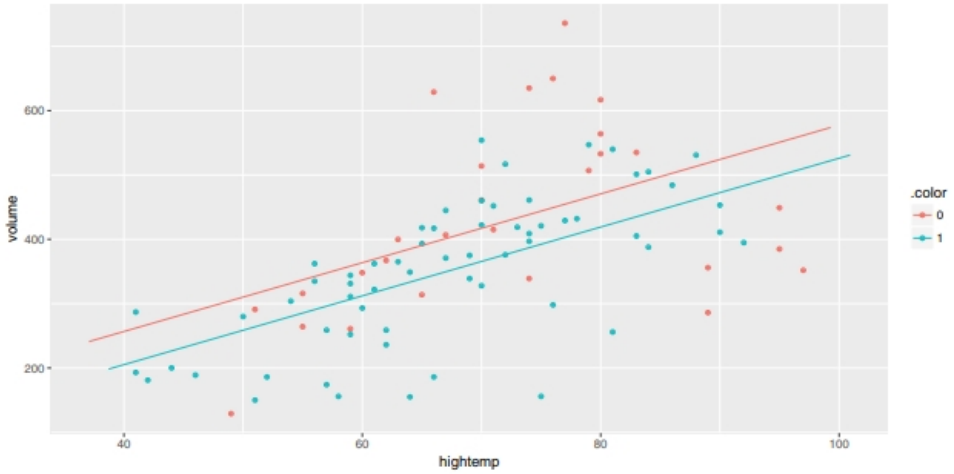

Figure E.3: Visualization of parallel slopes model for the rail trail data.

```
plotModel(mod_parallel, system = "ggplot2")
```

E.2.2 Parallel planes: Multiple regression with a second quantitative variable

If x_2 is a quantitative variable, then we have:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 \, .$$

Notice that our model is no longer a line, rather it is a *plane* that exists in three dimensions.

Now suppose that we want to improve our model for ridership by considering not only the average temperature, but also the amount of precipitation (rain or snow, measured in inches). We can do this in R by simply adding this variable to our regression model.

```
mod_planes <- lm(volume ~ hightemp + precip, data = RailTrail)
coef(mod_planes)
```

```
(Intercept)    hightemp      precip
    -31.520       6.118    -153.261
```

Note that the coefficient on `hightemp` (6.1 riders per degree) has changed from its value in the simple linear regression model (5.7 riders per degree). This is due to the moderating effect of precipitation. Our interpretation is that for each additional degree in temperature, we expect an additional 6.1 riders on the rail trail, after controlling for the amount of precipitation.

Pro Tip: Note that since the median precipitation on days when there was precipitation was only 0.15 inches, a predicted change for an additional inch may be misleading. It may be better to report a predicted difference of 0.15 additional inches or replace the continuous term in the model with a dichotomous indicator of any precipitation.
