Chapter 14: Linear Regression and Transformations

The previous chapter introduced us to ordinary least squares and its assumptions. Bounded data will, by definition, violate the assumption that the residuals are Normally distributed. So, how do we deal with bounded data? We transform it using a bijective function (one-to-one and onto). This chapter deals with the two types of boundedness and the appropriate bijective functions to use.

❧❧❧

The voters of Maine are being sent to the polls to vote on a constitutional referendum that proposes to limit the definition of marriage to the union of one man and one woman. This was not the first time that Americans were sent to the polls to vote on this or a closely related issue. Given the information from previous votes, what is the estimated probability that this ballot measure will pass in Maine?

In the previous chapter, we re-introduced linear modeling. In this chapter, we continue our treatment of linear modeling, but we begin to deal with some violations of the assumptions, thus extending the usefulness of this method.

The Ordinary Least Squares method (OLS) assumes that the error terms are Normally distributed. But, what happens when this is not true? Depending on the type and the severity of the violation, there are essentially three ways of handling it: First, you can ignore it. Ignoring this violation is usually not *too* bad when you are dealing with interpolation as the increase in bias and the loss of efficiency are usually minor. However, if the predictions are important, you definitely should not ignore this violation.

Second, we can use other methods (and modeling paradigms) for performing regression. Two popular alternatives to the Classical Linear Model paradigm are Generalized Linear Models (GLMs) and Generalized Additive Models (GAMs). The former paradigm will be covered in Chapters 15 through 19; the latter, well examined in Wood (2006). The strength of these models is that they extend the CLM to include (for instance) discrete dependent variables and non-linear relationships (Nelder and Wedderburn 1972; Wood 2006). These unified paradigms allow the computer to estimate the effect coefficients using a single method (called Maximum Likelihood Estimation). The drawback is that not all problems lend themselves to fitting using Maximum Likelihood Estimation (MLE). Luckily, most do.

Finally, you can transform the dependent variable into something more Normal-esque than before. These transformations are very flexible. Once you get used to working in two different systems of units, you can easily use transformation methods to 'Normalize' many restricted dependent variable. Unfortunately, one cannot transform an arbitrary dependent variable; there are types that cannot be fit using this technique, such as discrete variables. To handle these types of dependent variables, we will need to introduce a new modeling paradigm (Chapter 15).

14.1: The Issue of Boundedness

We finished Chapter 12 with a model of vote proportions for ballot measures concerning single-sex marriage. We applied that model to an upcoming vote in Maine to predict the outcome. Finally, we used Monte Carlo methods to predict the probability that the ballot measure would pass. In the end, we

	Estimate	Std. Error	t-value	p-value
Constant Term	0.1512	0.0659	2.293	0.0295
Year Passed (post 2000)	-0.0201	0.0036	-5.618	$\ll 0.0001$
Banned Civil Unions	-0.0373	0.0200	-1.868	0.0723
Percent Religious	0.0095	0.0011	8.801	$\ll 0.0001$

Table 14.1: *Results table for the regression of percent support of a generic ballot outlawing same-sex marriage against the three included variables. This is a replication of Table 12.4.*

predicted that the ballot measure had a 20% chance of passing, with a point-prediction of 42% of the voters in favor of the bill.

Results from extension problem EP12.7, however, suggest that there may be something inherently wrong with this model. To see this more clearly, let us predict the proportion of voters in support of a hypothetical 1994 ballot measure in Mississippi (religious precent = 85) that also banned civil unions (the results table from our SSM Vote model is replicated in Table 14.1).

From the results summarized in the table, the point-prediction for this 1994 Mississippian ballot measure is

$$\begin{aligned}\hat{p} &= 0.1512 + -0.0201(\texttt{yearPassed}) + -0.0373(\texttt{civilBan}) + 0.0095(\texttt{religPct})\\ &= 0.1512 + -0.0201(-6) + -0.0373(1) + 0.0095(85)\\ &= 1.0379\end{aligned}$$

Thus, this model predicts that the ballot measure will pass with over 103% of the vote — a physically impossible outcome. What went wrong? How can we fix this model so that this cannot happen?

First, nothing “went wrong,” *per se*. The model did *exactly* what it was supposed to do. The prediction, however, is based on a *linear* model. With linear models, we can always find large enough (or small enough) values for the independent variables to make the prediction arbitrarily large or small. When we are predicting a dependent variable that is bounded in theory, this will lead to an impossible prediction. Thus, the issue is with the linear aspect of the prediction equation *and* with the bounded nature of the dependent variable (bounded below by 0 and above by 1).

Thus, to improve the model, we only need to eliminate its boundedness; that is, we need to change the dependent variable so that all values make physical sense. This is done through the process of variable transformation. There are three steps: First, transform the dependent variable from

$$\begin{tikzcd}
\text{Level Units:} & Y \arrow[d, swap, "\text{Transform: } \tilde{y} = f(y)"] & & P \\
\text{Transformed Units:} & \widetilde{Y} \arrow[rr, swap, "\text{regression}"] & & \widetilde{P} \arrow[u, swap, "\text{Back-transform: } p = f^{-1}(\tilde{p})"]
\end{tikzcd}$$

Figure 14.1: *Schematic of a variable transformation procedure, such as described in the text. Here, Y is the original values of the dependent variable, $\widetilde{Y}$ is the transformed values of the dependent variable, $\widetilde{P}$ is the result from the regression in transformed units, and P is the result in the original (level) units.*

a restricted range to an unrestricted range. Second, perform the analysis on this transformed variable. Finally, back-transform the results into the original units.

The overview of this plan is shown in Figure 14.1. The key is the transformation. It must change the range of Y from its current limits to an unlimited version, denoted $\widetilde{Y}$. Luckily, there are two transformations that take care of most of our needs: the logit (*loh'-jit*) and the logarithm transformations.

14.2: Data Bounded by 0 and 1

One type of data you may come across in your research is proportion data, data where the values are bounded below and above (by 0 and 1, respectively); that is, if Y is the dependent variable, then $0 < Y < 1$. The (arguably) best function that transforms this bounded domain into an unbounded range is the logit function:

$$\tilde{y} = \text{logit}(y) := \log\left(\frac{y}{1-y}\right) \tag{14.1}$$

The logit function transforms variables bounded by 0 and 1 into unbounded variables; in symbols,

$$\text{logit} : (0,1) \mapsto \mathbb{R}$$

The logit's inverse, which transforms it from logit units back into level units is called the logistic function:

$$y = \text{logistic}(\tilde{y}) := \frac{1}{1+\exp(-\tilde{y})} = \frac{\exp(\tilde{y})}{1+\exp(\tilde{y})} \tag{14.2}$$

The logistic function transforms unbounded variables into variables bounded by 0 and 1:

$$\text{logistic} : \mathbb{R} \mapsto (0, 1)$$

Other transforms are available, but the logit is frequently used for the following three reasons:

1. The transformation *and* its inverse are both functions (the transform is a bijective function). This means that the results are always commensurate to the original problem.
2. The transformation is symmetric. This means that the 'stretching' is the same for values near 0 as they are for values near 1.
3. The function is exact, as opposed to the probit transform which requires numerical approximations. This increases the speed and accuracy of your predictions.

A careful reader will note that the domain of Y includes neither 0 nor 1. This is because there is no way of transforming a (semi-) closed interval into an open interval such as $\mathbb{R}$ while ensuring that the inverse is also a function. This is a provable fact of mathematics (Strichartz 2000).

But, what do we do if there are y-values that are zero (one)? One solution is to add (subtract) an extremely small number, ϵ, to the zero (one). A second solution is to completely drop those data from the analysis. A third solution is to change the proportion into a count and use a different paradigm (Chapter 17).

Note: None of these solutions is perfect. *If* you insist on using the CLM, then you should do all three and see how much your answer changes. A general rule of thumb is that if your underlying research model is correct then the results should not vary wildly based on similar models. That is, if we know Y depends on X_1 and X_2, then all appropriate modeling techniques should give *approximately* the same results. If they do not, then there is something seriously wrong with our assumptions about the underlying relationships — the model.

	Estimate	Std. Error	t-value	p-value
Constant Term	-1.7429	0.3553	-4.91	$\ll 0.0000$
Year Passed (after 2000)	-0.0914	0.0165	-5.56	$\ll 0.0000$
Contains a Civil Union Ban	-0.2007	0.1020	-1.97	0.0590
Percent Religious in State	0.0450	0.0060	7.49	$\ll 0.0000$

Table 14.2: *Results table of the results of regression on the dependent variable, using a logit link in the GLM. In this model, the residual deviance is 0.064987, the null deviance is 0.286802, the* $R^2 = 0.7734$, *and the* $AIC = -97.6$.

In reality, a final option (and the best) is to admit that this method is *not* appropriate and to use a more appropriate method, such as logistic regression (Chapter 16).

Example 14.1: The voters of Maine are being sent to the polls to vote on a constitutional referendum that proposes to limit the definition of marriage to the union of one man and one woman. This was not the first time that Americans were sent to the polls to vote on this or a closely related issue. Given the information from previous votes, what is the estimated probability that this ballot measure will pass in Maine?

Let us now answer this question more correctly. Recall that without performing a transformation of the dependent variable, there existed predictions which fell outside reality. To fix this, let us transform the dependent variable using the logit function, repeat the analysis, back-transform these transformed results to the original units, and compare results.

The first step is to transform the dependent variable. As the dependent variable is a proportion, let us use the `logit` transform (from the `RFS` package). If we decide to call the new variable `logitWin`, then the command will be

```
logitWin <- logit(propWin)
```

Now, this is our new dependent variable. As such, we perform the same analysis as in Chapter 12:

```
model.lgt <- lm(logitWin ~ yearPassed + civilBan + religPct)
```

The `summary(model.lgt)` command provides the results summarized in Table 14.2. Note that all three independent variables are more statistically significant than in the non-transformed model, Table 14.1. Also note that the effect directions are the same as before.

$$\begin{tikzcd}
\text{Level Units:} & \texttt{propWin} \arrow[d, "\text{logit()}"'] & \texttt{vote.me} \\
\text{Transformed Units:} & \texttt{logitWin} \arrow[r, "\text{lm()}"'] & \texttt{model.lgt} \arrow[u, "\text{logistic()}"']
\end{tikzcd}$$

Figure 14.2: *Schematic of the variable transformation procedure used in Example 14.1. Note that the results table, Table 14.2, displays the coefficients of* `model.lgt`*, which is in the transformed units, not the original units. As such you cannot compare these magnitudes with the magnitudes in Table 14.1.*

Note: You *cannot* directly compare the magnitudes of these coefficients with the magnitudes of the previous coefficients; these effect estimates are in different units. The coefficients seen in Table 14.1 predict in the original units (proportions). The coefficients in Table 14.2 predict in logit (of proportions) units. Furthermore, merely taking the logistic of the coefficients will not put them in level units; the transform is non-linear, as we designed, thus the effect of any depends on the values of all. In order to compare the two models, we need to perform predictions (remembering to back-transform them). Refer to Figure 14.2 for the steps we use in this example.

Predicting the proportion of the vote for the Maine ballot measure is almost as easy as it was before. The only additional step is that we need to back-transform the prediction to get it in proportion units.

So, according to this transformed model, what is the expected vote in Maine? To answer this, we need the Maine information: `yearPassed` = 9, `civilBan` = 0, `religPct` = 48. With this information, and under the assumption that the model is correct, we have our prediction of -0.4091 logits. Back-transforming this value gives a prediction of logistic(-0.4091) = 40% of the population will vote in favor of this ballot measure — slightly different from our original prediction of 42%.

However, remember that the original question was not this point estimate, it was a probability of the ballot measure passing. To determine this probability, we just need to repeat the same steps as we did answering this question before (Section 12.4.5), but remembering to back-transform the results.

The Monte Carlo results of the transformed model indicate that there is a 15% chance that the ballot measure will pass in Maine. The histogram of all million predictions is presented as Figure 14.3. From this information,

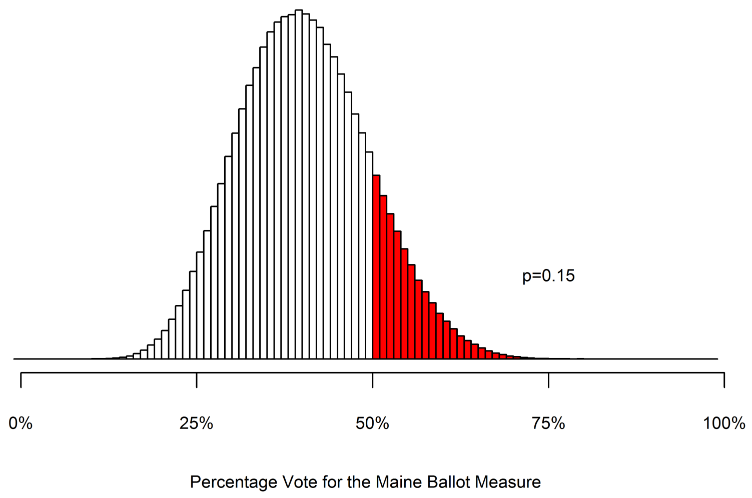

Figure 14.3: *Histogram of the results of the Monte Carlo experiment described in the text. Note that the distribution has a slight right-skew as a result of the transformation process. Also note that there are no predicted vote outcomes less than 0 or greater than 1, as compared to the original untransformed model of Chapter 12. In fact, the lowest prediction is 9.03%, while the largest is 81.63%.*

we can conclude that there is a slight chance that the SSM ballot measure will pass in Maine (15%), with a predicted 40% vote in favor. If we were into betting, we could also conclude that this model predicts that the odds of this ballot measure passing is $\frac{1.00-0.15}{0.15}$, 5.67-to-1 *against*.

14.3: Data Bounded Below by 0

When the data is proportion data (bounded by 0 and 1), we can use the logit function to transform it into an unbounded variable, perform the usual analysis, and back-transform those results into level units. However, not all bounded variables fit this description, e.g., age, height, income. Such variables are bounded below by 0, but have no theoretical upper bound. For such variables, we use the log transform.

The logarithm function transforms variables bounded below by 0 into unbounded variables; in symbols, $\log : (0, \infty) \mapsto \mathbb{R}$. Its inverse is the exponential function, $\exp : \mathbb{R} \mapsto (0, \infty)$. Both functions are bijections and so are appropriate functions for transforming our variables.

Example 14.2: The gross domestic product (GDP) per capita is one of many measures of average wealth in countries. If extant theory is correct, then the wealth in the country is directly affected by the level of honesty in the government — countries with high levels of honesty (low levels of corruption) should be wealthier than those with low levels of honesty (high levels of corruption). Furthermore, if theory is correct, the level of democracy in a country should *also* influence the country's level of wealth — countries with higher levels of democracy should be wealthier than countries with low levels of democracy.

Let us determine if reality (in the form of the data in `gdpcap`) supports the current theory or if current theory needs to explain the severe discrepancies. Furthermore, let us predict the GDP per capita for Papua New Guinea and provide a 95% confidence interval for that prediction.

I leave it as an exercise for you to model the data *without* transforming the dependent variable and discovering the predicted GDP per capita for Papua New Guinea is -\$2337, which is not physically possible (EP14.7). If nothing else, this prediction should tell you that the data needs transformation before being modeled.

The process to estimate the GDP per capita in Papua New Guinea is formulaic for us by now: transform the dependent variable by applying the logarithm function, model the transformed variable, predict in the transformed units, back-transformed into level units — here, dollars.

One feature of `R` that is shared by few other statistical packages is that you do not have to actually create a new variable; you can perform the transformation within the modeling command; *e.g.*,

```
model.log <- lm(log(gdpcap) ~ democracy + hig)
```

The results table for this model is provided in Table 14.3. Again, as we have transformed the dependent variable, the coefficients are not in units of dollars. As such, their magnitudes cannot be directly compared to those in the untransformed model, EP14.7. Their *directions*, however, can be directly compared. Thus, this model tells us that higher levels of honesty in government correspond to countries with higher GDPs per capita. Additionally, countries with higher democracy scores correspond to countries with *lower*

	Estimate	Std. Error	t-value	p-value
Constant term	6.9333	0.1479	46.89	$\ll 0.0001$
Level of Democracy	-0.0028	0.0113	-0.25	0.8055
Honesty in Government	0.4702	0.0359	13.11	$\ll 0.0001$

Table 14.3: *Results table for the GDP per capita modeling exercise. As the model is a transformed model, these effects estimates are not in units of dollars.*

GDPs per capita. This last finding, which conflicts with current theory, is not statistically significant at the usual $\alpha = 0.05$ level.

With this model, we can estimate the GDP per capita in Papua New Guinea using the standard method, but remembering that we must back-transform the final estimate. That is, if we used the commands

```
PNG <- data.frame(hig=2.1,democracy=10)
est <- predict(model.log, newdata=PNG)
```

then we would report our estimate of Papua New Guinea's GDP per capita as `exp(est)`, which is $2678.

The question asked us to calculate the prediction, but to also provide a 95% confidence interval for that prediction. To do this, we must again use Monte Carlo methods. The steps are all the same, with the additional step of back-transforming the predictions (Line 18).

```
1   set.seed(3)
2   outcome <- numeric()
3   trials  <- 1e6
4
5   b.int   <-  6.933298
6   b.dem   <- -0.002776
7   b.hig   <-  0.470225
8
9   s.int   <- 0.147873
10  s.dem   <- 0.011253
11  s.hig   <- 0.035855
12
13  e.int   <- rnorm(trials, m=b.int, s=s.int)
14  e.dem   <- rnorm(trials, m=b.dem, s=s.dem)
15  e.hig   <- rnorm(trials, m=b.hig, s=s.hig)
16
17  outcome <- e.int + e.dem*10 + e.hig*2.1
18  pred    <- exp(outcome)
```

The assignments in Lines 5–11 are the coefficient estimates and standard errors from the model (Table 14.3). The histogram of these results are provided

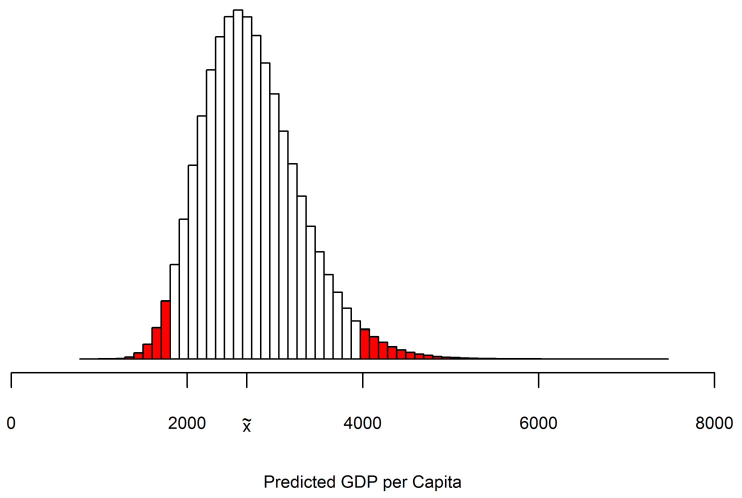

Figure 14.4: *Results of the Monte Carlo experiment predicting the GDP per capita for Papua New Guinea and its 95% confidence interval. Note that 5% of the predictions fall in the red region, 2.5% above and 2.5% below. The median of this distribution is designated by $\tilde{x}$.*

in Figure 14.4. To calculate a 95% confidence interval for our prediction, we merely find the values of `pred` for which 2.5% and 97.5% of the data are less.

To determine these bounds by hand, we merely list out all of the million predictions, sort them from lowest to highest, then pick out the prediction at $0.025 \times 1,000,000 = 25,000$ and at $0.975 \times 1,000,000 = 975,000$:

`sort(pred)[25000]` and `sort(pred)[975000]`

Alternatively, we can use the `quantiles` function:

```
quantiles(pred c(0.025,0.975))
```

From this, we can conclude that our model predicts the GDP per capita for Papua New Guinea is \$2678, with a 95% confidence interval being \$1807 to \$3973. It is interesting to note that the actual GDP per capita in Papua New Guinea is \$2400, which is well within our confidence interval.

Note: Here, I use the prediction as the point estimate for the GDP per capita of Papua New Guinea. It would have also been appropriate to use the mean of the Monte Carlo trials (\$2733) *or* the median of the Monte Carlo trials (\$2680). All three are acceptable measures of the center. It is usual, however, to use the original prediction.

14.4: Additional Bounds

Thus far, we have looked at transformation of a dependent variable when it is bounded above and below by 1 and 0, and when it is only bounded below by 0. Other bounds are possible. In this section, we figure out how to handle all types of bounds. The basic steps are to determine if the variable is bounded on one side or two. If one, then perform an *algebraic* transformation so that the new variable is bounded below by 0, then use the log transform. If two, then perform an algebraic transformation so that the new variable is bounded by 0 and 1, then use the logit transform. In either case, you will need to remember to back-transform the predictions with this algebraic transformation.

Note: The only bounds I have come across in my own research are those bounded by 0 and 1, bounded by 0 and 100 (percentages), and bounded below by 0. The quick solution for percentages is to divide them by 100 to make them proportions, then multiply the predictions by 100 to turn the predictions back into percentages.

14.4.1 Bounded by L and U

What if our data has a theoretic lower bound L and a theoretic upper bound U? As it is bounded above *and* below, we will change it into a proportion and using the logit transform as in Section 14.2, remembering to back-transform with the additional transformation. The algebraic transformation is

$$a(y) = p = \frac{y - L}{U - L}$$

The back-transform is

$$a^{-1}(p) = y = p(U - L) + L$$

Example 14.3: The scores on the quantitative portion of the Graduate Record Examination (GRE) range from $L = 200$ to $U = 800$. If we wished to properly model a person's GRE quantitative score, we would first subtract 200 from each score, then divide by $800 - 200 = 600$. The new variable would range from 0 to 1, a proportion.

Example 14.4: The grade point averages (GPAs) are bounded below by $L = 0$ and above by $U = 4$. To appropriately model GPAs, we would have to subtract 0, then divide by 4. This new variable would now be a proportion.

14.4.2 Bounded Below by L

It may be that your dependent variables is bounded below by a specific value, L, but not bounded above. As it is bounded on only *one* side, we will transform it into a variable bounded below by 0 and then apply the logarithm transform as in Section 14.3, remembering to back-transform with the additional transformation. The algebraic transformation is

$$a(y) = p = y - L$$

The back-transform is

$$a^{-1}(p) = y = p + L$$

Example 14.5: Hourly workers make at least \$7.25 per hour. To model

hourly wage, we would subtract off $L = 7.25$ from each hourly wage. This new variable is bounded below by 0, so we can apply the log transformation to it.

14.4.3 Bounded Above by U

It may be that your dependent variable is theoretically bounded *above* by U. As there is only *one* bound, we will perform an algebraic transformation so that it is bounded below by 0 and then apply the log transform as in Section 14.3, remembering to back-transform with the additional transformation. The algebraic transformation is

$$a(y) = p = U - y$$

The back-transform is

$$a^{-1}(p) = y = U - p$$

Example 14.6: In the ocean, different species live at different depths. In fact,

we can predict the depth based solely on the species observed. Ocean depth is bounded above by 0 and has no theoretic lower bound (although it certainly has a genuine lower bound at the Challenger Deep in the Mariana Trench, which has a depth of -35,994 ft). To transform the depths into a variable upon which we can perform a log transform, we subtract each value from $U = 0$. After we predict, we will have to back-transform by again subtracting each prediction from $U = 0$.

Of course, the transformation in this last example is equivalent to measuring depth in terms of 'distance below the surface', which is a positive number requiring no additional transformation.

Example 14.7: Free and fair elections are one of the requirements for a legitimate democratic system; furthermore, being a legitimate democratic State is necessary for some forms of external assistance. As such, many not-so-democratic States wish to appear democratic. They hold elections, but the elections are either fraudulent or the electoral system (rules governing the elections) is unfair.

There are many definitions for fairness in an election, but they all contain the same requirement that a person's vote has the same probability of being counted as anyone else's. In other words, the probability of a vote being invalidated is independent of the characteristics of the person casting the vote — including who the vote was for. This aspect of fairness can actually be tested in elections where the number of invalidated votes is counted: If the proportion of the vote for a specific candidate or position is not independent of the proportion of the vote invalidated in the electoral division, then there is evidence against the assumption of fairness.

Does the 2011 independence referendum in southern Sudan indicate an issue with fairness?

As one of the conditions to the 2005 Naivasha Agreement, which ended the civil war in Sudan, the South was allowed to vote on independence from the North. That referendum was held between January 9–15, 2011. Official results stated that 98.83% of the South Sudanese voted against unity and in favor of independence.

The `xsd2011referendum` data contains the number of votes in favor of independence (`Secession`), the number of votes declared invalid (`Invalid`), and the total number of votes cast (`Votes`). Because we need to determine if there is a (linear) relationship between the proportion of the vote for a specific side and the proportion of the vote invalidated in the electoral division, and because we just have vote counts, we need to create the

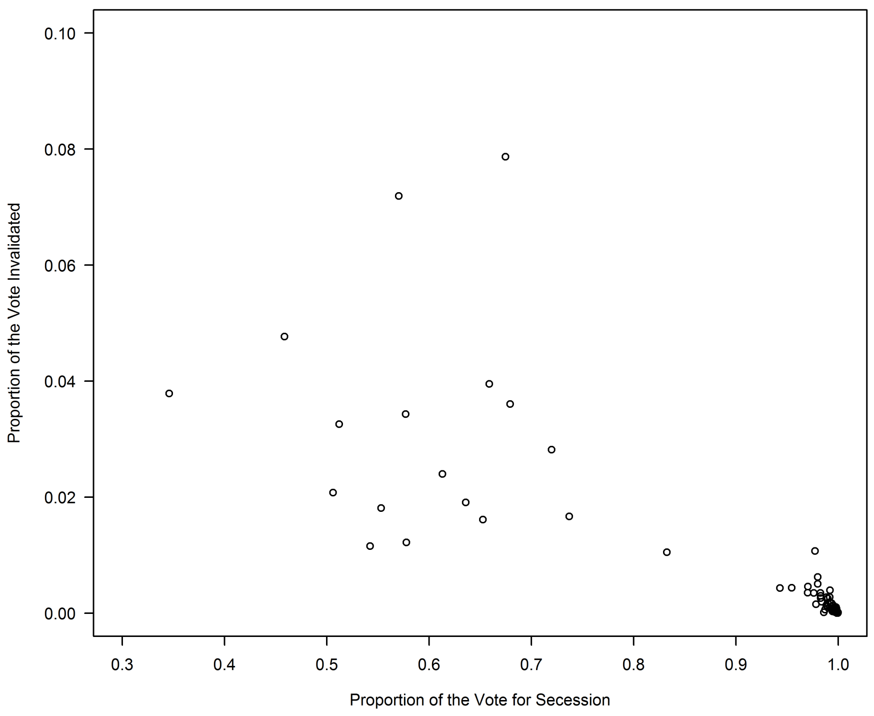

Figure 14.5: *A scatterplot of the results of the 2011 referendum on independence for South Sudan. Note the apparent presence of a relationship between these two variables. As such, there appears to be evidence that the election was not fair for those voting against independence.*

proportions by dividing the independence votes and the invalid votes by the total number of votes (*cf* Section 14.4.1).

Once that is done, we need to transform these proportions using the logit transformation, perform linear regression, and check for a relationship. If one exists in the transformed variables, then one exists in the untransformed variables. First, however, it is always a good idea to plot the variables to see if there is an obvious answer to the question. Figure 14.5 a the plot of proportion of the vote invalidated against the proportion of the vote in favor of independence.

Suggested by the plot, there appears to be a strong relationship between the two variables, evidence of an election that is not fair. Because of the direction of the slope, it appears as though those areas voting most

	Estimate	Std. Error	t-value	p-value
Constant term	1.3924	0.6613	2.11	0.0386
Proportion of Vote for Independence	-8.5511	0.7212	-11.86	$\ll 0.0001$

Table 14.4: *Results table for the South Sudan referendum. The results are in logit units. note the high level of statistical significance in the effect of the proportion of the vote in favor of independence. This is very indicative of a lack of fairness in the election.*

strongly in favor of independence had a much lower probability of having their votes rejected.

Note: As we are using the logit transform, we must drop any electoral division (here, county) which has zero invalid votes or zero votes in favor of secession. To easily do this in `R`, we can use the `which` function, which determines which entries have the provided condition. Thus,

```
which(xsd$Invalid==0)
```

returns a vector of values $\{15, 19, 23, 24, 28, 46, 47, 49, 50, 57, 72, 73\}$. These numbers correspond to the counties that had zero invalid votes cast. Storing this vector in the variable `dr` allows us to remove those counties from any subsequent calculations. As such, our proportion calculations are:

```
p.ind <- xsd$Secession[-dr]/xsd$Votes[-dr]
p.inv <- xsd$Invalid[-dr]/xsd$Votes[-dr]
```

The negative signs tells `R` to return values in the vector *other than* these entries.

The results of the linear regression on the transformed dependent variable are given in Table 14.4. There is a very strong relationship between the proportion of the vote invalidated in the county and the proportion of the vote in favor of secession: Those counties with a greater proportion of people voting for independence also had a lower proportion of the vote invalidated. That there is a strong relationship between these two variables is troubling.

To make this relationship more obvious, and to make our point stronger, we can plot the data, the prediction curve, and the 95% confidence bands on the same plot.

Note: What confidence intervals are for univariate data, so are confidence bands for bivariate data.

❧❧❧

In R, the philosophy behind graphing is to start with a fresh plot and paint successive layers on top of it. This allows us to create graphs that tell the story and to do so easily. To make the graph described above, we need to

1. Plot the points (displayed in proportion units),
2. Plot the prediction curve (displayed in proportion units, but calculated in logit units),
3. Plot the 95% confidence bands (displayed in proportion units, but calculated in logit units).

The first step has been done already (Figure 14.5):

```
plot(p.ind,p.inv)
```

The second step requires the repeated use of the `predict` function. First, to make things easier, let us define `indnew` as a series of "proportion of vote in favor of independence" values for which we want to make predictions: `indnew <- 0:100/100`. This creates a vector containing the values $0/100, 1/100, 2/100, \ldots,$ and 100/100. With this, our predict statement will be

```
l.pred <-predict(
          model.xsd,
          newdata=data.frame(p.ind=indnew),
          se.fit=TRUE
          )
```

Note: The `se.fit=TRUE` parameter will be important for calculating the confidence bands.

Remember that these predictions are in logit units. To get them into level units, we just apply the logistic function to these point predictions:

```
p.pred <- logistic(l.pred$fit)
```

Note: The `$fit` selects only the fitted predictions from the `l.pred` variable.

This is necessary as we are also using the `se.fit=TRUE` parameter.

Now that we have the predictions in the original units, we merely paint it on the current plot (from Step 1):

```
lines(indnew, p.pred)
```

The third step requires us to calculate the 95% confidence bands and paint them on the plot as well. The formula to calculate the upper 95% confidence bands is

```
ucb.l <- l.pred$fit+1.96*l.pred$se.fit
```

the lower,

```
lcb.l <- l.pred$fit-1.96*l.pred$se.fit
```

Note: These formulas should look vaguely familiar. They are the same formulas as when we calculated the upper and lower limits for Normal confidence intervals,

$$u = \overline{x} + 1.96 s_x \qquad \text{and} \qquad l = \overline{x} - 1.96 s_x$$

The 1.96 comes from the fact that we are using a Normal distribution and a 95% confidence level. A 90% confidence level would use 1.645.

Once again, we must back-transform these two variables using the logistic function. So, our final confidence bands are

`ucb <- logistic(ucb.l)` and `lcb <- logistic(lcb.l)`

Finally, we paint this on the current plot with

`lines(indnew,ucb, col=2)` and `lines(indnew,lcb, col=2)`

Putting all this together gives us Figure 14.6. Note that the predictions are curved in these units; they are straight in logit units. Also note the confidence bands are wider where the data is sparse. This is due to the same reasons confidence intervals are wider when the sample size is smaller. Lastly, note that no horizontal line can fit between the confidence bands. This indicates that there is a statistically significant relationship between the two variables at the $\alpha = 0.05$ level. It says the same thing as Table 14.4, but in a graphical manner. Graphs often makes the points more manifest.

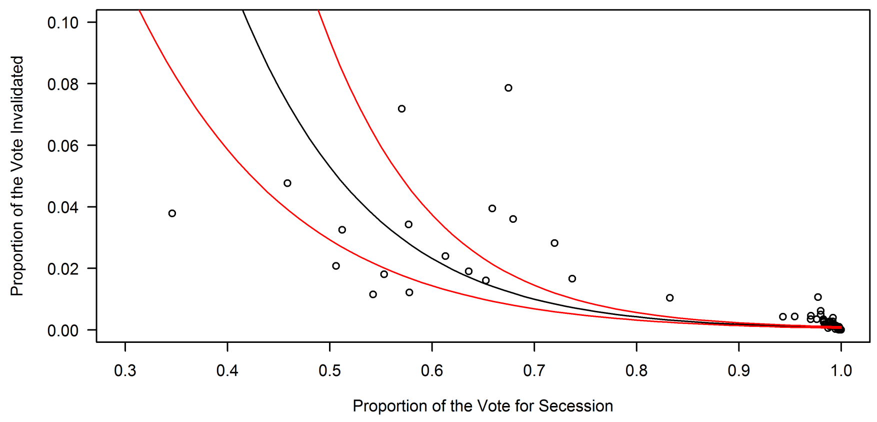

Figure 14.6: *A plot of the results of the South Sudan referendum. Included are the prediction line (in black) and the 95% confidence bands (in red). Note that a horizontal line cannot fit between the confidence bands. This indicates a statistically significant relationship between the proportion of the votes invalidated and the proportion of the votes in favor of independence. This, in turn, supports the conclusion of an unfair election.*

14.5: Conclusion

In this chapter, we focused on transforming bounded variables so that they did not violate the Normality assumptions as strongly as they did without the transformation. To accomplish this, we noted that there are three basic types of continuous variables: unbounded, bounded on one side, and bounded on two sides. If the dependent variable is unbounded, we do not necessarily need to transform it (although some transforms may reduce the non-Normality of the residuals). If the variable is bounded on one side, we performed an algebraic transformation so that it is bounded below by zero, then applied a log transformation. If the variable is bounded on two sides, we performed an algebraic transformation so that it was bounded by 0 and 1, then applied a logit transformation.

In either case, we needed to ensure that we back-transformed to the original units, first using an exponential or a logistic back-transform, then the inverse of our algebraic transform — order matters.

While this chapter does not exactly mark the end of continuous dependent variables, it does end our view of them in terms of the Classical Linear Model (CLM). This chapter already shows why the CLM needs to be replaced. Here, we were able to stay within the framework, but we had to perform variable transformations to make it work. Once we stray from continuous data, the CLM cannot work; there is no way of transforming a discrete dependent variable into a Normally distributed random variable. As such, we need an new paradigm — Generalized Linear Models (GLMs). The next chapter introduces GLMs, while still using a continuous dependent variable. This is done to show that GLMs can do anything CLMs can do. In fact, if you had used the `glm()` function in this and the previous chapter, in lieu of the `lm()` function, the results would be *exactly* the same, only the table layout would be different.

14.6: R Functions

In this chapter, we were introduced to several R functions that will be useful in the future. These are listed here.

14.6.0 Packages

RFS

14.6.0 Statistics

lm(formula) This function performs linear modeling on the data, with the supplied formula. As there is much information contained in this function, you will want to save the results in a variable, to retrieve the information through the `summary()` and `names()` functions.

predict(model) The `predict` function calculates the value of the dependent variable in the `model` given the independent variables used to create the model. If new predictions are required, the `newdata=` parameter must be used. This parameter takes a new set of data as its argument. Make sure that all independent variables used in the model are defined in the `newdata=` parameter. If not, an error message will results. Finally, the `se.fit=TRUE` parameter calculates the standard error at each prediction point.

14.6.0 Probability

pnorm(x) This function is the cumulative distribution function (CDF) for the Normal distribution. It returns a *p*robability that a Normally-distributed variable will be less than or equal to `x`. This function has two additional parameters that remove the requirement that `x` has undergone the z-transformation, `m` and `s`.

rnorm(n, m, s) This function returns `n` draws from a Normal distribution centered at `m` and with a standard deviation `s`. This function is the cornerstone of much Monte Carlo analysis.

14.6.0 Graphics

lines(x,y) This is an extremely handy line-generating function, painting a line on the current plot (or returns an error if no plot exists). It first invisibly plots the pairs of points (`x`,`y`) then connects the points with drawn line segments.

If the `col` parameter is not set, then the line will be black. Otherwise, the line will be the color specified. There are three ways of stating the color: using the Windows 1-16 values, using names, and using the rgb values. The following all refer to 'red': `col=2`, `col="red"`, and `col="#ff0000"`.

plot() This function produces a scatterplot of the two-dimensional data. The call can be either `plot(x,y)` or `plot(y~x)`; both give identical results. This function can produce graphs that are very customized. The `R` help file for `par` is invaluable. Some important parameters include `xlab=""` (label for the x-axis), `ylab=""` (label for the y-axis), `xlim=c(min,max)` and `ylim=c(min,max)` (axis limits, min and max, for the x- and y-axis), and `las=1` (makes axis values painted horizontal).

14.6.0 Mathematics

log(x, b) This returns the logarithm of `x`, with a base of `b`. If you omit the `b`, this function returns the natural logarithm of `x`. To calculate the common logarithm, set `b=10`. The logarithm function is used to transform variables bounded on one side into variables bounded on neither side.

exp(x) This function returns the exponential of the argument, `x`; that is, it returns e^x. The exponential function is the inverse of the logarithm function.

logit(x) This function returns the logit of the provided number. This number must be between 0 and 1, not including either 0 or 1. The logit function is frequently used to transform proportions into unbounded data. It is available through the `RFS` package.

logistic(x) This function returns the logistic of a given number. The range of ths logistic function is 0 to 1, exclusive. it is the inverse of the logit function. As such, it is often used to transform predictions from logit units to proportion units. It is available through the `RFS` package.

cloglog(x) The complementary log-log function is a second appropriate transformation for proportion data. It is, however, not a symmetric function. It is available through the `RFS` package.

cloglog.inv(x) This function is the inverse of the complementary log-log function. It is available through the `RFS` package.

14.6.0 Programming

which(condition) This function returns a vector of indices corresponding to the original vector's values meeting the criteria. Thus, `which(x==4)` returns the indices of all elements in vector `x` that equal 4. Note that equality is checked with a *double* equals, ==. Other comparisons include: >, <, >=, <=, `!=`, `&`, `|`, and `!`. The last four are 'not equal to', 'and', 'or', and 'not'.

14.7: Extensions

This section offers suggestions on things you can practice from this chapter. Save the scripts in your Chapter 14 folder. For each of the following problems, please save the associated `R` script in the chapter folder as `ext0x.R`, where `x` is the problem number.

1. Predict the Mississippi 1994 SSM ballot measure vote using the transformed SSM Vote model. Is the prediction physically possible?

2. Determine a 95% confidence interval, with the *un*transformed SSM Vote model, for predicting Maine's vote. Is the actual outcome within the 95% confidence interval?

3. Determine a 95% confidence interval, with the *transformed* SSM Vote model, for predicting Maine's vote. Is the actual outcome within the 95% confidence interval?

4. Determine if the assumptions of OLS are violated in the transformed SSM Vote model.

5. The actual vote share for Maine was 52.8%. Explain why both models failed in predicting the actual vote outcome. How bad was the error? What can be done to improve the predictions?

6. The logit transformation is not the only possible choice. There is also the asymmetric complementary log-log transformation (`cloglog` in the `RFS` package). Use this function as the transformation to predict Maine's vote, its 95% confidence interval, and the probability of the SSM ballot measure passing. The inverse of the complementary log-log transform has no name, but the `R` function is `cloglog.inv`, also in the `RFS` package.

7. Estimate the GDP per capita for Papua New Guinea using the *un*transformed model, as well as the 95% confidence interval. How close is this estimate to the real answer, and it the real answer within the predicted confidence interval?

14.8: Applications

- James M. Avery. (2009) “Political Mistrust among African Americans and Support for the Political System.” *Political Research Quarterly* 62(1): 132–45.
- Mark Andreas Kayser. (2009) “Partisan Waves: International Business Cycles and Electoral Choice.” *American Journal of Political Science* 53(4): 950–70.
- Pamela A. Morris. (2008) “Welfare Program Implementation and Parents’ Depression.” *The Social Service Review* 82(4): 579–614.
- Kar Tean Tan, Christopher C. White, and Donald L. Hunston. (2011) “An adhesion test method for spray-applied fire-resistive materials.” *Fire and Materials* 35(4): 245–59.

14.9: References and Additional Readings

- George Casella and Roger L. Berger. (2001) *Statistical Inference*. New York: Duxbury Press.
- Annette J. Dobson and Adrian Barnett. (2008) *An Introduction to Generalized Linear Models*, Third Edition. New York: Chapman & Hall.
- Julian J. Faraway. (2004) *Linear Models with R*. New York: Chapman & Hall.
- Julian J. Faraway. (2005) *Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models*. New York: Chapman & Hall.
- John A. Nelder and Robert W. M. Wedderburn. (1972) "Generalized Linear Models." *Journal of the Royal Statistical Society. Series A (General)* 135(3): 370–84.
- Shayle R. Searle. (1997) *Linear Models*. New York: Wiley-Interscience.
- James H. Stapleton. (2009) *Linear Statistical Models*. New York: John Wiley and Sons.
- Robert S. Stritchartz. (2000) *The Way of Analysis*, Revised Edition. Boston: Jones and Bartlett Mathematics.
- Simon N. Wood. (2006) *Generalized Additive Models: An Introduction with R*. New York: Chapman & Hall.