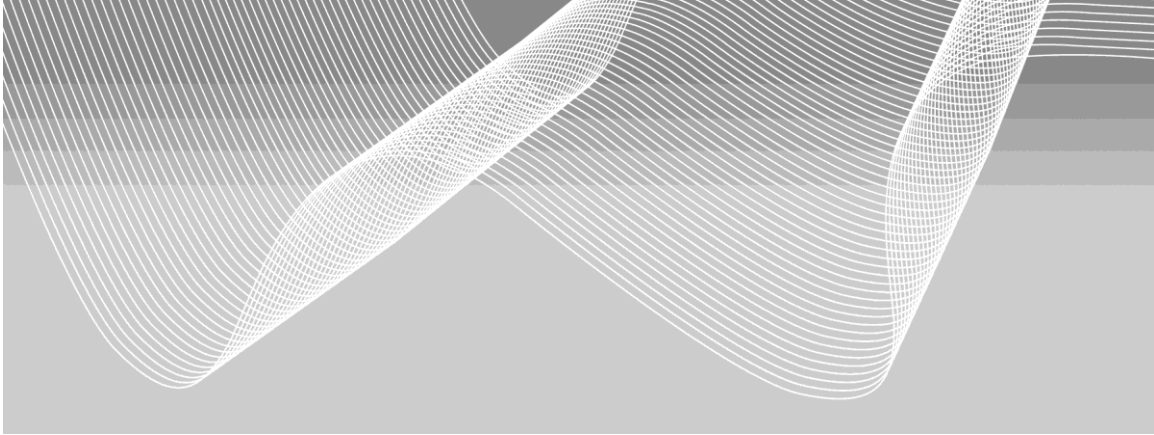


Fundamentals of Predictive Analytics with JMP®

Ron Klimberg and B. D. McCullough

Contents

- Chapter 1 Introduction
- Chapter 2 Statistics Review
- Chapter 3 Introduction to Multivariate Data
- Chapter 4 Regression and ANOVA Review
- Chapter 5 Logistic Regression
- Chapter 6 Principal Component Analysis
- Chapter 7 Cluster Analysis
- Chapter 8 Decision Trees
- Chapter 9 Neural Networks
- Chapter 10 Model Comparison
- Chapter 11 Overview of Predictive Analytics and the Modeling Process
- Appendix Data Sets

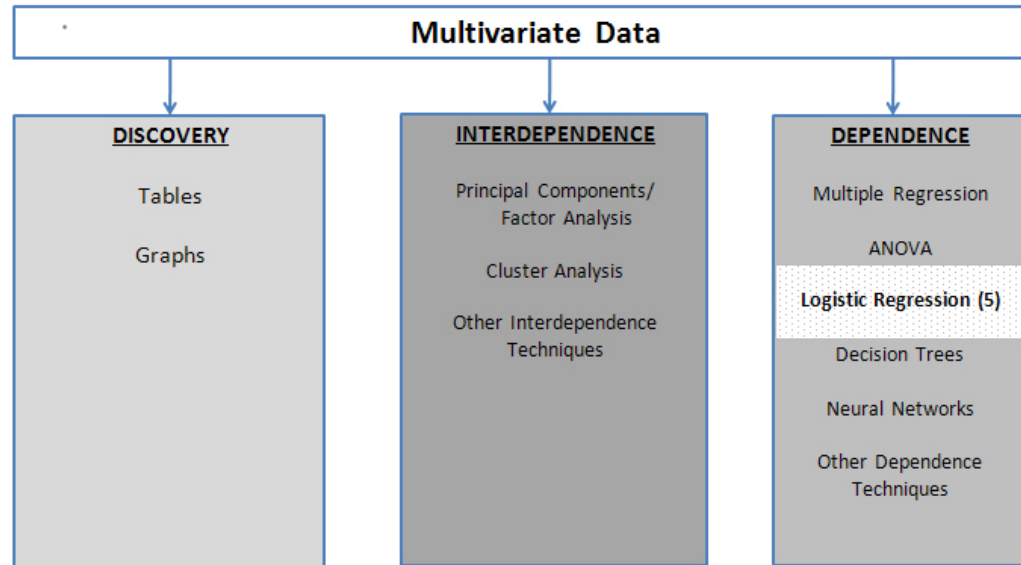


Chapter 5

Logistic Regression

A Logistic Regression Statistical Study	19
References	30
Exercises	30

Figure 5.1 A Framework for Multivariate Analysis



Logistic regression as shown in our multivariate analysis framework in Figure 5.1 is one of the dependence techniques in which the dependent variable is discrete and, more specifically, binary: taking on only two possible values. Some examples: Will a credit card applicant pay off his bill or not? Will a mortgage applicant default? Will someone who receives a direct mail solicitation respond to the solicitation? In each of these cases the answer is either “yes” or “no”. Such a categorical variable cannot directly be used as a dependent variable in a regression, but a simple transformation solves the problem: let the dependent variable Y take on the value 1 for “yes” and 0 for “no”.

Since Y takes on only the values 0 and 1, we know $E[Y_i] = 1 \cdot P[Y_i=1] + 0 \cdot P[Y_i=0] = P[Y_i=1]$ but from the theory of regression we also know that $E[Y_i] = a + b \cdot X_i$ (here we use simple regression but the same holds true for multiple regression). Combining these two results we have $P[Y_i=1] = a + b \cdot X_i$ and we can see that, in the case of a binary dependent variable, the regression may be interpreted as a probability. We then seek to use this regression to estimate the probability that Y takes on the value 1. If the estimated probability is high enough, say, above 0.5, then we predict 1; conversely, if the estimated probability of a 1 is low enough, say, below 0.5, then we predict 0.

When linear regression is applied to a binary dependent variable, it is called commonly the Linear Probability Model (LPM). Traditional linear regression is designed for a continuous dependent variable, and is not well-suited to handling a binary dependent variable. Three primary difficulties arise in the LPM. First, the predictions from a linear regression do not necessarily fall between zero and one; what are we to make of a

predicted probability greater than one? How do we interpret a negative probability? A model that is capable of producing such nonsensical results does not inspire confidence.

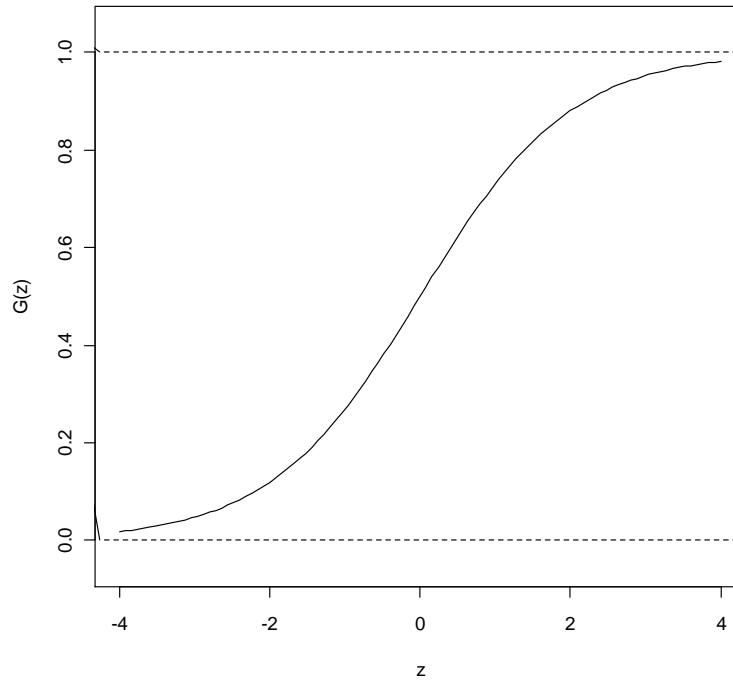
Second, for any given predicted value of y (denoted \hat{y}), the residual ($\text{resid} = y - \hat{y}$) can take only two values. For example, if $\hat{y} = 0.37$, then the only possible values for the residual are $\text{resid} = -0.37$ or $\text{resid} = 0.63 (= 1 - 0.37)$, since it has to be the case that $\hat{y} + \text{resid}$ equals zero or one. Clearly the residuals will not be normal, and plotting a graph of \hat{y} vs. resid will produce not a nice scatter of points, but two parallel lines. The reader should verify this assertion by running such a regression and making the requisite scatterplot. A further implication of the fact that residual can take on only two values for any \hat{y} is that the residuals are heteroscedastic; this violates the linear regression assumption of homoscedasticity (constant variance). The estimates of the standard errors of the regression coefficients will not be stable and inference will be unreliable.

Third, the linearity assumption is likely to be invalid, especially at the extremes of the independent variable. Suppose we are modeling the probability that a consumer will pay back a \$10,000 loan as a function of his income. The dependent variable is binary, 1 = he pays back the loan, 0 = he does not pay back the loan. The independent variable is income, measured in dollars. If the person's income is \$50,000, he might have a probability of 0.5 of paying back the loan. If his income is increased by \$5,000 then his probability of paying back the loan might increase to 0.55, so that every \$1000 increase in income increases the probability of paying back the loan by 1%. A person with an income of \$150,000 (who can pay the loan back very easily) might have a probability of 0.99 of paying back the loan. What happens to this probability when his income is increased by \$5000? Probability cannot increase by 5%, because then it would exceed 100%, yet according to the linearity assumption of linear regression, it must do so.

A better way to model $P[Y_i=1]$ would be to use a function that is not linear, one that increases slowly when $P[Y_i=1]$ is close to zero or one, and that increases more rapidly in between; it would have an "S" shape. One such function is the logistic function

$$G(z) = \frac{1}{1 + e^{-z}} = \frac{e^z}{1 + e^z}$$

whose cumulative distribution function is shown in Figure 5.2.

Figure 5.2 The logistic function

Another useful representation of the logistic function is

$$1 - G(z) = \frac{e^{-z}}{1 + e^{-z}}$$

Recognize that the y-axis, $G(z)$, is a probability and let $G(z) = \pi$, the probability of the event occurring. We can form the odds ratio (the probability of the event occurring divided by the probability of the event not occurring) and do some simplifying:

$$\frac{\pi}{1 - \pi} = \frac{G(z)}{1 - G(z)} = \frac{1}{\frac{e^{-z}}{1 + e^{-z}}} = \frac{1}{e^{-z}} = e^z$$

Consider taking the natural logarithm of both sides. The left side will become $\log[\pi / (1 - \pi)]$ and the log of the odds ratio is called the logit. The right hand side will become z (since $\log(e^z) = z$) so that we have the relation

$$\log \left[\frac{\pi}{1 - \pi} \right] = z$$

and this is called the logit transformation.

If we model the logit as a linear function of X , *i.e.*, let $z = \beta_0 + \beta_1 X$, then we have

$$\log \left[\frac{\pi}{1 - \pi} \right] = \beta_0 + \beta_1 X$$

We could estimate this model by linear regression and obtain estimates b_0 of β_0 and b_1 of β_1 if only we knew the log of the odds ratio for each observation. Since we do not know the log of the odds ratio for each observation we will use a form of nonlinear regression called logistic regression to estimate the below model:

$$E[Y_i] = \pi_i = G(\beta_0 + \beta_1 X_i) = \frac{1}{1 + e^{-\beta_0 - \beta_1 X_i}}$$

and in so doing obtain the desired estimates b_0 of β_0 and b_1 of β_1 . The estimated probability for an observation X_i will be

$$P[Y_i = 1] = \hat{\pi}_i = \frac{1}{1 + e^{-b_0 - b_1 X_i}}$$

and the corresponding estimated logit will be

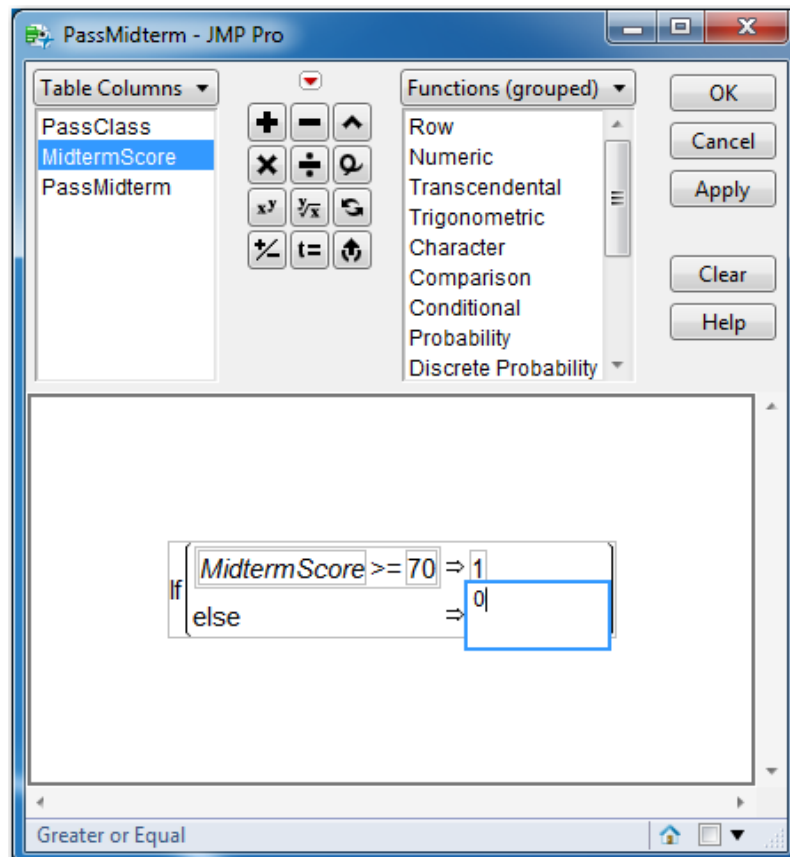
$$\log \left[\frac{\hat{\pi}}{1 - \hat{\pi}} \right] = b_0 + b_1 X$$

which leads to a natural interpretation of the estimated coefficient in a logistic regression: b_1 is the estimated change in the logit (log odds) for a one unit change in X .

To make these ideas concrete, suppose we open a small dataset `toylogistic.jmp`, containing students' midterm exam scores (`MidtermScore`) and whether or not the student passed the class (`PassClass=1` if pass, `PassClass=0` if fail). A passing grade for the midterm is 70. The first thing to do is create a dummy variable to indicate whether or not the student passed the midterm: `PassMidterm = 1` if `MidtermScore` \geq 70 and `PassMidterm = 0` otherwise:

Click on **Cols**→**New Column** produces the New Column dialog box. In the Column Name text box, type in for our new dummy variable PassMidterm. Click on the drop box for modeling type and change it to Nominal. Click the drop box for Column Properties and select Formula. The Formula dialog box appears. Under Functions click Conditional → If. Under Table Columns click on MidtermScore so that it appears in the top box to the right of the If. Under Functions click Comparison > “a>=b”. In the formula box to the right of >= enter 70. Click tab. In the box to the right of the ⇒ click it on and enter the number **1** and similarly enter 0 for the else clause. The Formula dialog box should look like Figure 5.3.

Figure 5.3 Formula dialog box



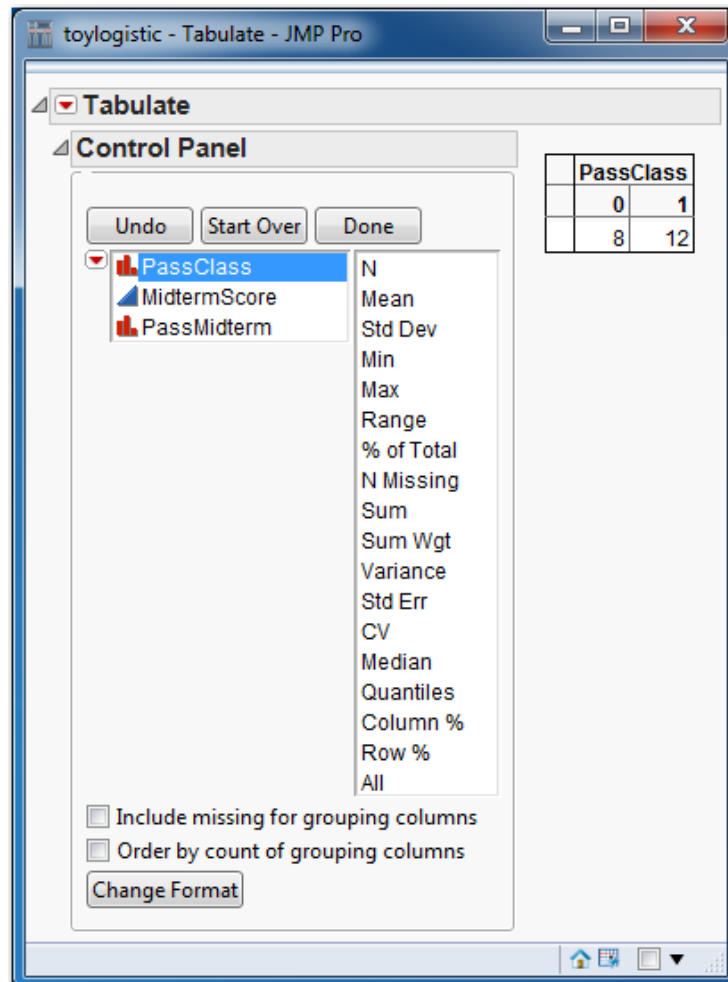
Click **OK**→**OK**.

First let us use a traditional contingency table analysis to determine the odds ratio. Make sure that both PassClass and PassMidterm are classified as nominal variables. Right-

click on the column PassClass in the data grid and select **Column Info...**. Beside Modeling Type, click on the black triangle and select Nominal, then **OK**. Do the same for PassMidterm.

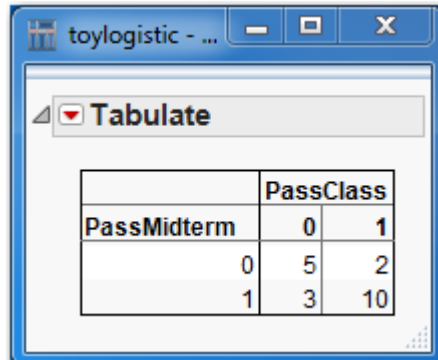
Select **Tables**→**Tabulate** and the **Control Panel** will appear; it shows the general layout for a table. Select PassClass, drag and drop it into “Drop zone for columns” and select “Add Grouping Columns”. Now that data have been added, the words “Drop zone for rows” no longer will be visible, but the “Drop zone for rows” will still be the lower left panel of the table. See Figure 5.4.

Figure 5.4 Control Panel for Tabulate



Select **PassMidterm**, drag and drop it to the panel immediately to the left of the “8” in the table; select “Add Grouping Columns”. Click on “Done”. A contingency table identical to Figure 5.5 will appear.

Figure 5.5 Contingency table from toydataset.jmp.



		PassClass	
PassMidterm		0	1
0		5	2
1		3	10

The probability of passing the class when you did not pass the midterm is:

$$P(\text{PassClass}(1) | P(\text{PassMidterm}(0))) = \frac{2}{7} \text{ and the probability of not passing the class}$$

when you did not pass the midterm: $P(\text{PassClass}(0) | P(\text{PassMidterm}(0))) = \frac{5}{7}$ (similar to row percentages). The odds of passing the class given that you have failed the midterm is:

$$\frac{P(\text{PassClass}(1) | P(\text{PassMidterm}(0)))}{P(\text{PassClass}(0) | P(\text{PassMidterm}(0)))} = \frac{2/7}{5/7} = \frac{2}{5}$$

Simply considering only the students that did not pass the midterm, the odds the number of students that pass the class divided by the number of students that did not pass the class.

Similarly, we calculate the odds of passing the class given that you have passed the midterm as:

$$\frac{P(\text{PassClass}(1) | P(\text{PassMidterm}(1)))}{P(\text{PassClass}(0) | P(\text{PassMidterm}(1)))} = \frac{10/13}{3/13} = \frac{10}{3}$$

Of the students that did pass the midterm, the odds is the number of students that pass the class divided by the number of students that did not pass the class.

In the above paragraphs we spoke only of “odds”. Now let us calculate an “odds ratio”. It is important to note that this can be done in two equivalent ways. Suppose we want to know the odds ratio of passing the class by comparing those who pass the midterm (PassMidterm=1 in the numerator) to those who fail the midterm (PassMidterm=0 in the denominator). The usual calculation leads to:

$$\frac{\text{Odds of passing the class; given passed the Midterm}}{\text{Odds of passing the class; given failed the Midterm}} = \frac{10/3}{2/5} = \frac{50}{6} = 8.33.$$

which has the following interpretation: the odds of passing the class are 8.33 times the odds of failing the course if you pass the midterm. This odds ratio can be converted into a probability. We know that $P(Y=1)/P(Y=0)=8.33$, and by definition $P(Y=1)+P(Y=0)=1$, so solving two equations in two unknowns yields $P(Y=0) = (1/(1+8.33)) = (1/9.33)= 0.1072$ and $P(Y=1) = 0.8928$. As a quick check, observe that $0.8928/0.1072=8.33$. Note that the log-odds is $\ln(8.33) = 2.120$. Of course, the user doesn't have to perform all these calculations by hand; JMP will do them automatically. When a logistic regression has been run, simply clicking on the red triangle and selecting “Odds Ratios” will do the trick.

Equivalently, we could compare those who fail the midterm (PassMidterm=0 in the numerator) to those who pass the midterm (PassMidterm=1 in the denominator) and calculate:

$$\frac{\text{Odds of passing the class; given failed the Midterm}}{\text{Odds of passing the class; given passed the Midterm}} = \frac{2/5}{10/3} = \frac{6}{50} = \frac{1}{8.33} = 0.12.$$

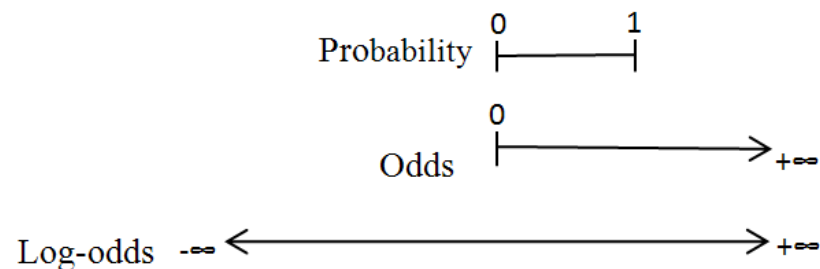
which tells us that the odds of failing the class are 0.12 times the odds of passing the class for a student who passes the midterm. Since $P(Y = 0) = 1 - \pi$ (the probability of failing the midterm) is in the numerator of this odds ratio, we must interpret it in terms of the event failing the midterm. It is easier to interpret the odds ratio when it is less than 1 by using the following transformation: $(OR - 1)*100\%$. Compared to a person who passes the midterm, a person who fails the midterm is 12% as likely to pass the class, or equivalently, a person who fails the midterm is 88% less likely, $(OR - 1)*100\% = (0.12 - 1)*100\% = -88\%$, to pass the class than someone who passed the midterm. Note that the log-odds is $\ln(0.12) = -2.12$.

The relationships between probabilities, odds (ratios), and log-odds (ratios) are straightforward. An event with a small probability has small odds, and also has small log-odds. An event with a large probability has large odds and also large log-odds. Probabilities are always between zero and unity; odds are bounded below by zero but can be arbitrarily large; log-odds can be positive or negative and are not bounded, as shown

in Figure 5.6. In particular, if the odds ratio is 1 (so the probability of either event is 0.50), then the log-odds equals zero. Suppose $\pi = 0.55$ so the odds ratio $0.55/0.45 = 1.222$. Then we say that the event in the numerator is $(1.222-1) = 22.2\%$ more likely to occur than the event in the denominator.

Different software packages adopt different conventions for handling the expression of odds ratios in logistic regression. By default JMP has uses the “log odds of 0/1” convention which puts the “0” in the numerator and the “1” in the denominator. This is a consequence of the sort order of the columns, which we will address shortly.

Figure 5.6 Ranges of Probabilities, Odds and Log-odds



To see the practical importance of this, rather than compute a table and perform the above calculations, we can simply run a logistic regression. It is important to make sure that **PassClass** is nominal and that **PassMidterm** is continuous. If **PassMidterm** is nominal, JMP will fit a different but mathematically equivalent model that will give different (but mathematically equivalent) results. The scope of the reason for this is beyond this book, but interested readers can consult **Help**→**Books**→**Modeling and Multivariate Methods** and refer to Appendix A.

If you have been following along with the book, both variables ought to be classified as nominal, so **PassMidterm** needs to be changed to continuous. Right-click on the column **PassMidterm** in the data grid and select **Column Info...** . Beside Modeling Type, click on the black triangle and select Nominal, then **OK**.

Now that the dependent and independent variables are correctly classified as Nominal and Continuous, respectively, let's run the logistic regression:

Click from the top menu **Analyze**→**Fit Model**. Click on **PassClass** and click on **Y**. Click **PassMidterm** and click on **Add**. The Fit Model dialog box should now look like Figure 5.7. Click **Run**.

Figure 5.7 Fit Model dialog box

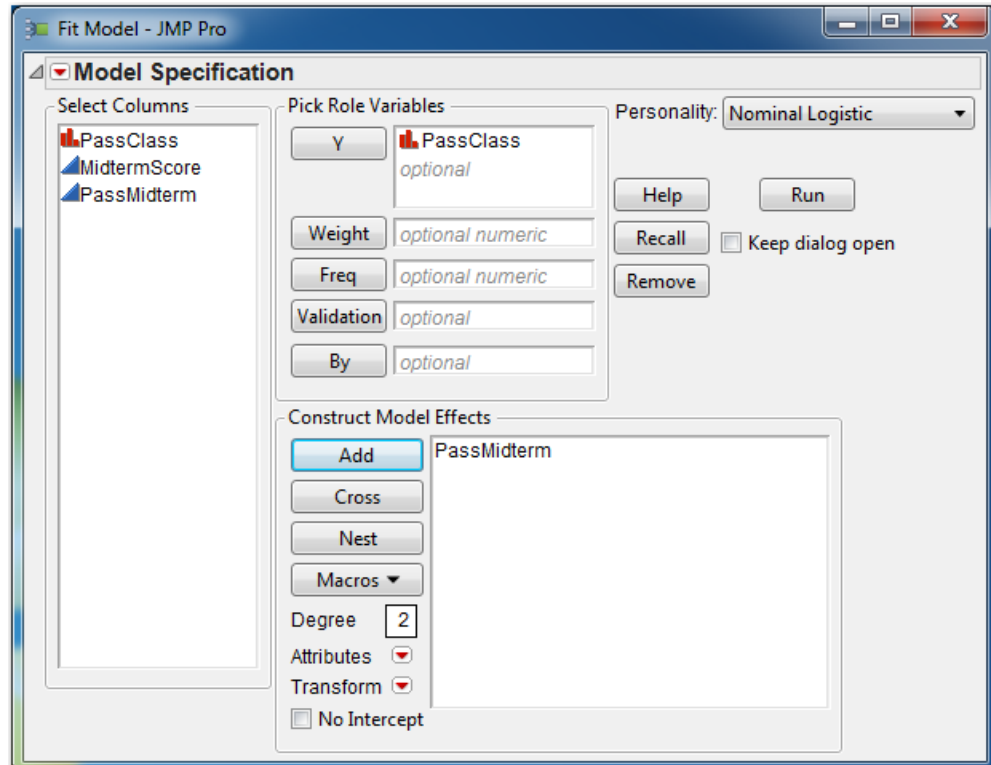
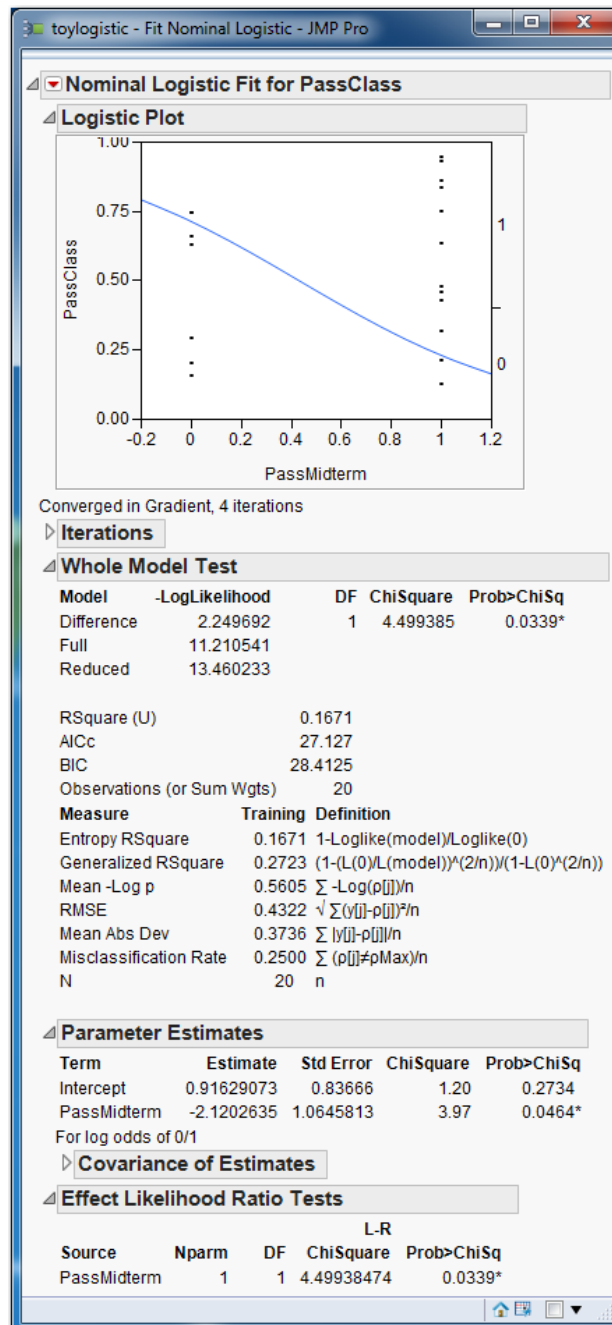


Figure 5.8 displays the logistic regression results.

Figure 5.8 Logistic Regression Results for toylogistic.jmp



Examine the parameter estimates in Figure 5.8. The intercept is 0.91629073 and the slope is -2.1202635. The slope gives the expected change in the logit for a one unit change in the independent variable, *i.e.*, the expected change on the log of the odds ratio. However, if we simply exponentiate the slope, *i.e.*, compute $e^{-2.1202635} = 0.12$, then we get the 0/1 odds ratio.

There is no need for us to exponentiate the coefficient manually. JMP will do this for us:

Click on the red triangle and select Odds Ratios. The Odds Ratios tables are added to the JMP output as shown in Figure 5.9.

Figure 5.9 Odds Ratios tables using the nominal independent variable PassMidterm

▲ Odds Ratios
For PassClass odds of 0 versus 1
Tests and confidence intervals on odds ratios are likelihood ratio based.

▲ Unit Odds Ratios
Per unit change in regressor

Term	Odds Ratio	Lower 95%	Upper 95%	Reciprocal
PassMidterm	0.12	0.011664	0.855944	8.3333333

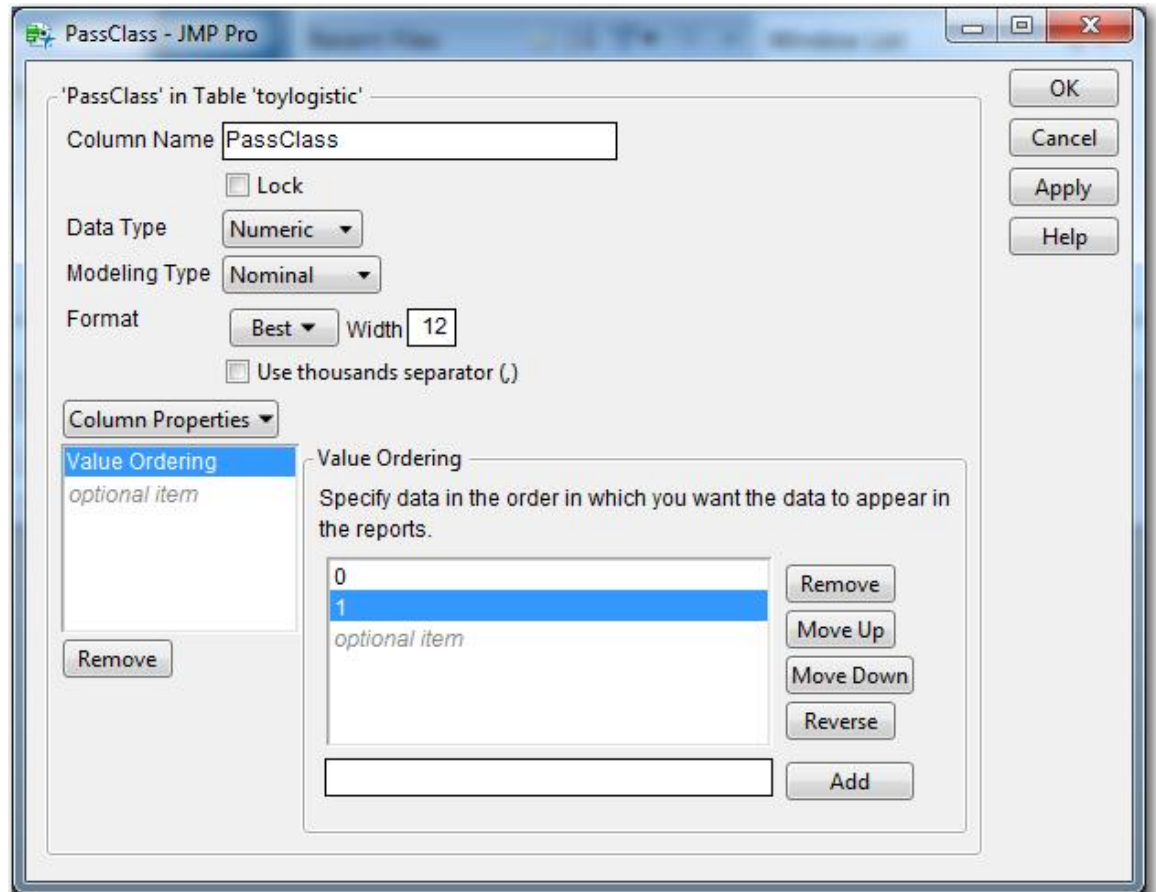
▲ Range Odds Ratios
Per change in regressor over entire range

Term	Odds Ratio	Lower 95%	Upper 95%	Reciprocal
PassMidterm	0.12	0.011664	0.855944	8.3333333

Unit Odds Ratios refers to the expected change in the odds ratio for a one-unit change in the independent variable. Range Odds Ratios refers to the expected change in the odds ratio when the independent variable changes from its minimum to its maximum. Since the present independent variable is a binary 0-1 variable, these two definitions are the same. We get not only the odds ratio, but a confidence interval, too. Notice the right-skewed confidence interval; this typical of confidence intervals for odds ratios.

To change from the default “log odds of 0/1” convention which puts the “0” in the numerator and the “1” in the denominator, in the data table rightclick on the name of the PassClass column. Under “Column Properties” select “Value Ordering”. Click on the value “1” and click “Move Up” as in Figure 5.10.

Figure 5.10 Changing the Value Order



Then, when you re-run the logistic regression, while the parameter estimates will not change, the odds ratios will change to reflect the fact that the “1” is now in the numerator and the “0” is in the denominator.

The independent variable is not limited to being only a nominal (or ordinal) dependent variable, it can be continuous. In particular, let’s examine the results using the actual score on the midterm, **MidtermScore** as an independent variable:

Click **Analyze**→**Fit Model**. Click PassClass and click on **Y** and then click MidtermScore and click on **Add**. Click **Run**.

This time the intercept is 25.6018754 and the slope is -0.3637609, so we expect the log-odds to decrease by 0.3637609 for every additional point scored on the midterm, as shown in Figure 5.11.

Figure 5.11 Parameter Estimates

Parameter Estimates				
Term	Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept	25.6018754	11.184069	5.24	0.0221*
MidtermScore	-0.3637609	0.1581661	5.29	0.0215*

For log odds of 0/1

To view the effect on the odds ratio itself, as before click on the red triangle and click Odds Ratios. Figure 5.12 displays the Odds Ratios tables.

Figure 5.12 Odds Ratios tables using the continuous independent variable MidtermScore

	Pass Class	Midterm Score	Lin[0]	Prob[0]	Prob[1]	Most Likely PassClass
1	0	62	3.048697664	0.9547262676	0.0452737324	0
2	0	63	2.6849367335	0.936131922	0.063868078	0
3	0	64	2.3211758029	0.9106156911	0.0893843089	0
4	0	65	1.9574148724	0.8762529099	0.1237470901	0
5	0	66	1.5936539419	0.8311295662	0.1688704338	0
6	0	70	0.1386102197	0.5345971803	0.4654028197	0
7	0	72	-0.588911641	0.3568846133	0.6431153867	1
8	0	74	-1.316433502	0.2114122777	0.7885877223	1
9	1	68	0.8661320808	0.7039402276	0.2960597724	0
10	1	69	0.5023711503	0.6230163983	0.3769836017	0
11	1	71	-0.225150711	0.4439489049	0.5560510951	1
12	1	73	-0.952672572	0.2783476639	0.7216523361	1

For a one unit increase in the midterm score, the new odds ratio will be 69.51% of the old odds ratio or, equivalently, we expect to see a 30.5% reduction in the odds ratio $(0.695057 - 1) * 100\% = -30.5\%$). For example, suppose a hypothetical student has a midterm score of 75%. His log odds of failing the class would be $25.6018754 - 0.3637609 * 75 = -1.680192$, so his odds of failing the class would be $\exp(-1.680192) =$

0.1863382; that is, he is much more likely to pass than fail. Converting odds to probabilities ($0.1863328/(1+0.1863328) = 0.157066212786159$), we see that his probability of failing the class is 0.15707, and his probability of passing the class is 0.84293. Now, if he increased his score by one point to 76, then his log odds of failing the class would be $25.6018754 - 0.3637609*76 = -2.043953$. Thus, his odds of failing the class becomes $\exp(-2.043953) = 0.1295157$. So, his probability of passing the class would rise to 0.885334, and his probability of failing the class would fall to 0.114666. With respect to the Unit Odds Ratio, which equals 0.695057, we see that a one unit increase in the test score changes the odds ratio from 0.1863382 to 0.1295157. In accordance with the estimated coefficient for the logistic regression, the new odds ratio is 69.5% of the old odds ratio because $0.1295157/0.1863382 = 0.695057$.

Finally, we can use the logistic regression to compute probabilities for each observation. As noted, the logistic regression will produce an estimated logit for each observation. These can be used, in the obvious way, to compute probabilities for each observation. Consider a student whose midterm score is 70. His estimated logit is $25.6018754 - 0.3637609(70) = 0.1386124$. Since $\exp(0.1386129) = 1.148679 = \pi/(1-\pi)$, we can solve for π (the probability of failing) = 0.534597.

We can obtain the estimated logits and probabilities by clicking the red triangle on “Normal Logistic Fit” and selecting **Save Probability Formula**. Four columns will be added to the worksheet: Lin[0], Prob[0] and Prob[1]. These give for each observation the estimated logit, the probability of failing the class, and the probability of passing the class, respectively. Observe that the sixth student has a midterm score of 70. Look up his estimated probability of failing (Prob[0]); it is very close to what we just calculated above. See Figure 5.13. The difference being the computer carries 16 digits through its calculations, while we carried only six.

Figure 5.13 Verifying calculation of probability of failing

	Pass Class	Midterm Score	Lin[0]	Prob[0]	Prob[1]	Most Likely PassClass
1	0	62	3.048697664	0.9547262676	0.0452737324	0
2	0	63	2.6849367335	0.936131922	0.063868078	0
3	0	64	2.3211758029	0.9106156911	0.0893843089	0
4	0	65	1.9574148724	0.8762529099	0.1237470901	0
5	0	66	1.5936539419	0.8311295662	0.1688704338	0
6	0	70	0.1386102197	0.5345971803	0.4654028197	0
7	0	72	-0.588911641	0.3568846133	0.6431153867	1
8	0	74	-1.316433502	0.2114122777	0.7885877223	1
9	1	68	0.8661320808	0.7039402276	0.2960597724	0
10	1	69	0.5023711503	0.6230163983	0.3769836017	0
11	1	71	-0.225150711	0.4439489049	0.5560510951	1
12	1	73	-0.952672572	0.2783476639	0.7216523361	1

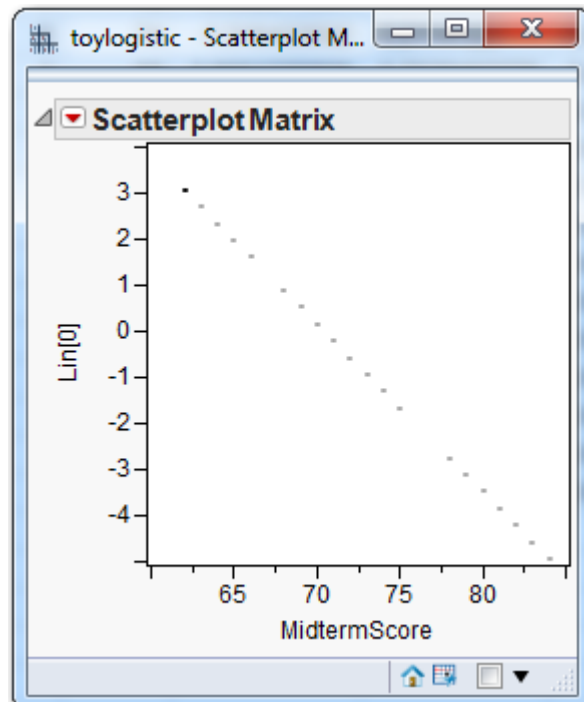
The fourth column (Most Likely PassClass) classifies the observation either 1 or 0 depending upon whether the probability is greater than or less than 50%. We can observe how well our model classifies all the observations (using this cut off point of 50%) by producing a confusion matrix: Click on the red triangle and select Confusion matrix. Figure 5.14 displays the confusion matrix for our example. The rows of the confusion are the actual classification, that is, whether PassClass is 0 or 1. The columns are the predicted classification from the model, that is, the predicted 0/1 values from that last fourth column using our logistic model and a cut point of .50. Correct classifications are along the main diagonal from upper left to lower right. We see the model classified 6 students as not passing the class, and actually they did not pass the class. The model also classifies 10 students as passing the class when they actually did. The values on the other diagonal, both equal to 2, are misclassifications. The results of the confusion matrix will be examined in more detail when we discuss model comparison in Chapter 9.

Figure 5.14 Confusion matrix

Confusion Matrix		
Actual	Predicted	
Training	0	1
0	6	2
1	2	10

Of course, before we can use the model we have to check the model's assumptions, etc. The first step is to verify the linearity of the logit; this can be done by plotting the estimated logit against PassClass: Click on Graph→Scatterplot Matrix. Click on Lin[0] and click **Y, columns** and click MidtermScore and click **X**. Click **OK**. As shown in Figure 5.15, the linearity assumption appears to be perfectly satisfied.

Figure 5.15 Scatterplot of Lin[0] and MidtermScore



The analog to the ANOVA F Test for linear regression is found under the Whole Model Test, shown in Figure 5.16, in which the Full and Reduced models are compared. The null hypothesis for this test is that all the slope parameters are equal to zero. Since Prob→ChiSq is 0.0004, this null hypothesis is soundly rejected. For a discussion of other statistics found here, such as BIC and Entropy RSquare, see the **JMP Help**.

Figure 5.16 Whole Model Test for the Toylogistic Dataset

Whole Model Test				
Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	6.264486	1	12.52897	0.0004*
Full	7.195748			
Reduced	13.460233			
RSquare (U)		0.4654		
AICc		19.0974		
BIC		20.383		
Observations (or Sum Wgts)		20		

The next important part of model checking is the Lack of Fit test. See Figure 5.17. It compares the model actually fitted to the saturated model. The saturated model is a model generated by JMP that contains as many parameters as there are observations, and so fits the data very well. The null hypothesis for this test is that there is no difference between the estimated model and the saturated model. If this hypothesis is rejected, then more variables (such as cross-product or squared terms) need to be added to the model. In the present case, as can be seen, Prob>ChiSq=0.7032. We can therefore conclude that we do not need to add more terms to the model.

Figure 5.17 Lack of Fit Test for current model

Lack Of Fit			
Source	DF	-LogLikelihood	ChiSquare
Lack Of Fit	18	7.1957477	14.3915
Saturated	19	0.0000000	Prob>ChiSq
Fitted	1	7.1957477	0.7032

A Logistic Regression Statistical Study

Let's turn now to a more realistic dataset with several independent variables. During this discussion we will also present briefly some of the issues that should be addressed and some of the thought processes during a statistical study.

Cellphone companies are very interested in determining which customers might switch to another company; this is called "churning". Predicting which customers might be about to churn enables the company to make special offers to these customers, possibly stemming their defection. Churn.jmp contains data on 3333 cellphone customers,

including the variable Churn (0 means the customer stayed with the company, 1 means the customer left the company). Before we can begin constructing a model for customer churn, we need to discuss model building for logistic regression. Statistics and econometrics texts devote entire chapters to this concept; in several pages we can only sketch the broad outline. The first thing to do is make sure that the data are loaded correctly. Observe that Churn is classified as Continuous; be sure to change it to Nominal. One way is to right-click on the Churn column in the data table, select “Column Info...” and under “Modeling Type” choose “Nominal”. Another way is to look at the list of variables on the left side of the data table, find Churn, click on the blue triangle (which denotes a continuous variable) and change it to nominal (the blue triangle then becomes a red histogram). Check to make sure that all binary variables are classified as Nominal. This includes Intl_Plan, VMail_Plan, E_VMAIL_PLAN, and D_VMAIL_PLAN. Should Area_Code be classified as Continuous or Nominal? (Nominal is the correct answer!) CustServ_Call, the number of calls to customer service, could be treated as either continuous or nominal/ordinal; we treat it as continuous.

When building a linear regression model and the number of variables is not so large that this cannot be done manually, one place to begin is by examining histograms and scatterplots of the continuous variables, and crosstabs of the categorical variables as discussed in Chapter 3. Another very useful device as discussed in Chapter 3 is the scatterplot/correlation matrix which can, at a glance, suggest potentially *useful* independent variables that are correlated with the dependent variable. The scatterplot/correlation matrix approach cannot be used with logistic regression, which is nonlinear, but a method similar in spirit can be applied.

We are now faced with a similar situation that was discussed in Chapter 4 in which our goal is to build a model that follows the principle of parsimony, that is, a model which explains as much as possible of the variation in Y while using as few significant independent variables as possible. However, now with multiple logistic regression, we are in a nonlinear situation. We have four approaches we could take. We briefly list and discuss each of these approaches and some of their advantages and disadvantages:

- **Include all the variables.** In this approach you just input all the independent variables into the model. An obvious advantage of this approach is that it is fast and easy. However, depending on the dataset, most likely several independent variables will be insignificantly related to the dependent variable. Including variables that are not significant can cause severe problems—weaken the interpretation of the coefficients and lessen the prediction accuracy of the model. This approach definitely does not follow the principle of parsimony, and it can cause numerical problems for the nonlinear solver that may lead to a failure to obtain an answer.
- **Bivariate method.** In this approach you search for independent variables that may have predictive value for the dependent variable by running a series of bivariate logistic regressions, *i.e.*, we run a logistic regression for each of the

independent variables, searching for "significant" relationships. A major advantage of this approach is that it is the one most agreed upon by statisticians [1]. On the other hand, this approach is not automated, very tedious and is limited by the analyst's ability to run the regressions; that is, it is not practical with very large data sets. Further, it misses interaction terms which, as we shall see, can be very important.

- **Stepwise.** In this approach you would use the Fit Model platform, change the Personality to Stepwise and Direction to Mixed. The Mixed option is like Forward Stepwise, but variables can be dropped after they have been added. An advantage of this approach is that it is automated, so, it is fast and easy. The disadvantage of stepwise is that it could lead to possible interpretation and prediction errors depending on the data set. However, using the Mixed option, as opposed to the Forward or Backward Direction option, tends to lessen the magnitude and likelihood of these problems.
- **Decision Trees.** A Decision Tree is a data mining technique that can be used for variable selection and will be discussed in Chapter 8. The advantage of using decision trees is that it is automated, and it is fast and easy to run. Further, it is a popular variable reduction approach taken by many data mining analysts, [2]. However, somewhat like the stepwise approach, the decision tree approach could lead to some statistical issues. In this case, significant variables identified by a decision tree are very sample dependent. These issues will be discussed further in Chapter 8.

No one approach is a clear cut winner. Nevertheless, we do not recommend using the "Include all the variables" approach. If the data set is too large and/or you do not have the time, we recommend that you run both the stepwise and decision trees models and compare the results. The dataset churn.jmp is not too large, so we will apply the bivariate approach.

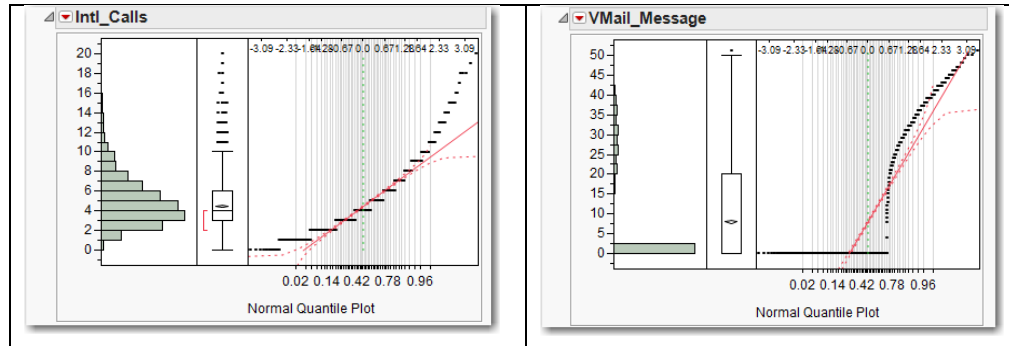
While it is traditional to choose $\alpha = 0.05$, in this preliminary stage we adopt a more lax standard, $\alpha = 0.25$. The reason for this is we want to include, if possible, a group of variables that individually are not significant but together are significant. Having identified an appropriate set of candidate variables, run a logistic regression including all of them. Compare the coefficient estimates from the multiple logistic regression with the estimates from the bivariate logistic regressions. Look for coefficients that have changed in sign or have dramatically changed in magnitude, as well as changes in significance; such changes indicate the inadequacy of the simple bivariate models, and confirm the necessity of adding more variables to the model.

Three important ways to improve a model are as follows:

- If the logit appears to be nonlinear when plotted against some continuous variable, one resolution is to convert the continuous variable to a few dummies, say, three, that cut the variable at its 25th, 50th and 75th percentiles.
- If a histogram shows that a continuous variable has an excess of observations at zero (which can lead to nonlinearity in the logit), adding a dummy variable that equals one if the variable equals zero and equals zero otherwise.
- Finally, a seemingly-continuous variable that is actually discrete can be broken up into a handful of dummy variables, *e.g.*, zip codes.

Before we can begin modeling, we must first explore the data. With our churn data set, creating and examining the histograms of the continuous variables reveals nothing much of interest, except VMail_Message, which has an excess of zeros (see the second point in the previous paragraph). Figure 5.18 shows plots for Intl_Calls and VMail_Message. To produce such plots, select **Analyze**→**Distribution**, click on Intl_Calls and then “**Y, Columns**” and “**OK**”. To add the Normal Quantile Plot, click on the red arrow next to Intl_Calls and select “Normal Quantile Plot”. Here it is obvious that Intl_Calls is skewed right. We note that a logarithmic transformation of this variable might be in order, but we will not pursue the idea.

Figure 5.18 Distribution of Intl_Calls and VMail_Message



A correlation matrix of the continuous variables (**Graph**→**Scatterplot Matrix**, put the desired variables in **Y, Columns**) turns up a curious pattern: Day_Charge and Day_Mins, Eve_Charge and Eve_Mins, Night_Charge and Night_Mins, and Intl_Charge and Intl_Mins all are perfectly correlated. The charge is obviously a linear function of the number of minutes. Therefore we can drop the “Charge” variables from our analysis (we could also drop the “Mins” variables instead; it doesn’t matter which one we drop). If our dataset had a very large number of variables, the scatterplot matrix would be too big to comprehend. In such a situation, we would choose groups of variables for which to make scatterplot matrices, and examine those.

A scatterplot matrix for the four binary variables turns up an interesting association. E_VMAIL_PLAN and D_VMAIL_PLAN are perfectly correlated; both have common 1s and where the former has -1 the latter has zero. It would be a mistake to include both of these variables in the same regression (try it and see what happens). Let's delete E_VMAIL_PLAN from the data set and also delete VMail_Plan as it agrees perfectly with E_VMAIL_PLAN: when the former has a "no" the latter has a "-1", and similarly for "yes" and "+1".

Phone is more or less unique to each observation (we ignore the possibility that two phone numbers are the same but have different area codes), and therefore should not be included in the analysis. So, we will drop Phone from the analysis.

A scatterplot matrix between the remaining continuous and binary variables turns up a curious pattern: D_VMAIL_PLAN and VMailMessage have a correlation of 0.96. They have zeros in common, and where the former has 1s the latter has numbers (see again point two in the above paragraph, we won't have to create a dummy variable to solve the problem as D_VMAIL_PLAN will do the job nicely).

To summarize, we have dropped 7 of the original 23 variables from the dataset: Phone, Day_Charge, Eve_Charge, Night_Charge, Intl_Charge, E_VMAIL_PLAN, VMail_Plan, so there are now 16 variables left, one of which is the dependent variable, Churn. We have 15 possible independent variables to consider.

Next comes the time-consuming task of running several bivariate (two-variable, one dependent and one independent) analyses, some of which will be logistic regressions (when the independent variable is continuous) and some of which will be contingency tables (when the independent variable is categorical). In total we have 15 bivariate analyses to run. What about Area Code? JMP reads it as a continuous variable, but it's really nominal, so be sure to change it from continuous to nominal. Similarly, be sure that D_VMAIL_PLAN is set as a nominal variable, not continuous.

Do *not* try to keep track of the results in your head, or by referring to the 15 bivariate analyses that would fill your computer screen. Make a list of all 15 variables that need to be tested, and write down the test result (*e.g.*, the relevant p-value) and your conclusion (*e.g.*, "include" or "exclude"). This not only prevents simple errors, it is a useful record of your work should you have to come back to it later. There are few things more pointless than conducting an analysis that concludes with a 13 variable logistic regression, only to have some reason to rerun the analysis and now wind up with a 12 variable logistic regression. Unless you have documented your work, you will have no idea why the discrepancy exists or which is the correct regression.

Below we briefly show how to conduct both types of bivariate analyses, one for a nominal independent variable and one for a continuous independent variable. We leave the other 14 to the reader.

Examining the results of a contingency table of Churn vs. State (**Analyze**→**Fit Y by X**, click Churn (which is nominal) and then click **Y, Response**, click State and then click **X, Factor**; and click **OK**) at the bottom of the table of results are the Likelihood Ratio and Pearson tests, both of which test the null hypothesis that State does not affect Churn, and both of which reject the null. State matters. On the other hand, performing a logistic regression of Churn on VMail_Message (**Analyze**→**Fit Y by X**, click Churn and click **Y, Response** and click VMail_Message and click **X, Factor**; and click **OK**), under “Whole Model Test” that $\text{Prob} > \text{ChiSq}$, the p-value of less than 0.0001, so we conclude that VMail_message affects Churn. Remember that for all these tests, we are setting α (probability of Type I error) = 0.25.

In the end, we have 10 candidate variables for possible inclusion in our multiple logistic regression model:

State	Intl_Plan	D_VMAIL_PLAN
VMail_Message	Day_Mins	Eve_Mins
Night_Mins	Intl_Mins	Intl_Calls
CustServ_Call		

Remember that the first three of these variables (the first row) should be set to nominal, and the rest to continuous (of course, leave the dependent variable Churn as nominal!).

Let’s run our initial multiple logistic regression with Churn as the dependent variable and the above 10 variables as independent variables:

Click **Analyze**→**Fit Model**, click Churn and click **Y**, and select the above 10 variables (to select variables that are not consecutive, click on each variable while holding down the Ctrl key), and click **Add**. Check the box next to Keep dialog open. Click **Run**.

The Whole Model Test lets us know that our included variables have an effect on the Churn and with a p-value less than .0001, as shown in Figure 5.19.

Figure 5.19 Whole Model Test and Lack of Fit for the churn data set.

Nominal Logistic Fit for Churn				
Converged in Gradient, 6 iterations				
Iterations				
Whole Model Test				
Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	341.6995	59	683.3991	<.0001*
Full	1037.4471			
Reduced	1379.1467			
RSquare (U)		0.2478		
AICc		2197.13		
BIC		2561.59		
Observations (or Sum Wgts)		3333		
Measure	Training	Definition		
Entropy RSquare	0.2478	$1 - \text{Loglike}(\text{model}) / \text{Loglike}(0)$		
Generalized RSquare	0.3293	$(1 - (L(0)/L(\text{model}))^{2/n}) / (1 - L(0)^{2/n})$		
Mean -Log p	0.3113	$\sum -\text{Log}(p_{[j]})/n$		
RMSE	0.3070	$\sqrt{\sum (y_{[j]} - p_{[j]})^2/n}$		
Mean Abs Dev	0.1883	$\sum y_{[j]} - p_{[j]})/n$		
Misclassification Rate	0.1308	$\sum (p_{[j]} \neq p_{\text{Max}})/n$		
N		3333	n	
Lack Of Fit				
Source	DF	-LogLikelihood	ChiSquare	Prob>ChiSq
Lack Of Fit	3273	1037.4471	2074.894	
Saturated	3332	0.0000		Prob>ChiSq
Fitted	59	1037.4471		1.0000

The Lack Of Fit test tells us that we have done a good job explaining Churn. From the Lack of Fit we see that $-\text{LogLikelihood}$ for the Full model is 1037.4471. Now, linear regression minimizes the sum of squared residuals, so when comparing two linear regressions the preferred one has the smaller sum of squared residuals. In the same way, the nonlinear optimization of the logistic regression minimizes the $-\text{LogLikelihood}$ (which is equivalent to maximizing the LogLikelihood), so the model with the smaller $-\text{LogLikelihood}$ is preferred to a model with a larger $-\text{LogLikelihood}$.

Examining the p-values of the independent variables in the Parameter Estimates, a variable for which Prob>ChiSq is less than 0.05 is said to be significant, otherwise it is said to be insignificant, similar to what is practiced in linear regression. The regression output gives two sets of tests, one for the “Parameter Estimates” and another for “Effect Likelihood Ratio Tests”. We shall focus on the latter. To see why, consider the State

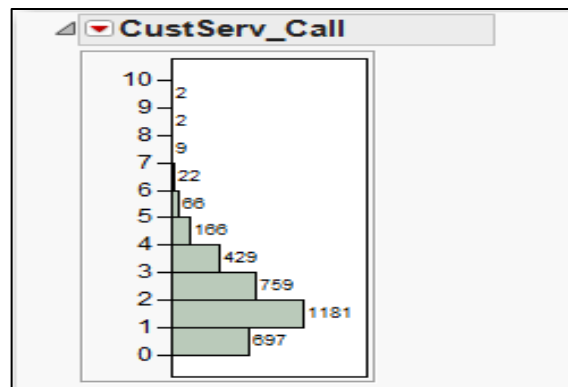
variable, which is really not one variable but many dummy variables. We are not so much interested in whether any particular state is significant or not (which is what the Parameter Estimates tell us) but whether, overall, the collection of state dummy variables is significant. This is what the Effect Likelihood Ratio Tests tells us; the effect of all the state dummies is significant with a “Prob>ChiSq” of 0.0010. True, many of the State dummies are insignificant, but overall State is significant; we will keep this variable as it is. It may prove worthwhile to reduce the number of state dummies into a handful of significant states and small clusters of “other” states that are not significant, but we will not pursue this line of inquiry here.

We can see that all the variables in the model are significant. We may be able to derive some new variables that help improve the model. We will provide two examples of deriving new variables—(1) Converting a continuous variable into discrete variables; (2) Producing interaction variables.

Let us try to break up a continuous variable into a handful of discrete variables. An obvious candidate is CustServ_Call. Look at its distribution in Figure 5.20.

Analyze→**Distribution**, select CustServ_Call, select “**Y, Columns**” and click “**OK**”. Click the red arrow next to CustServ_Call and uncheck “**Outlier Box Plot**”, then choose “**Histogram Options**”→“**Show Counts**”.

Figure 5.20 Histogram of CustServ_Call

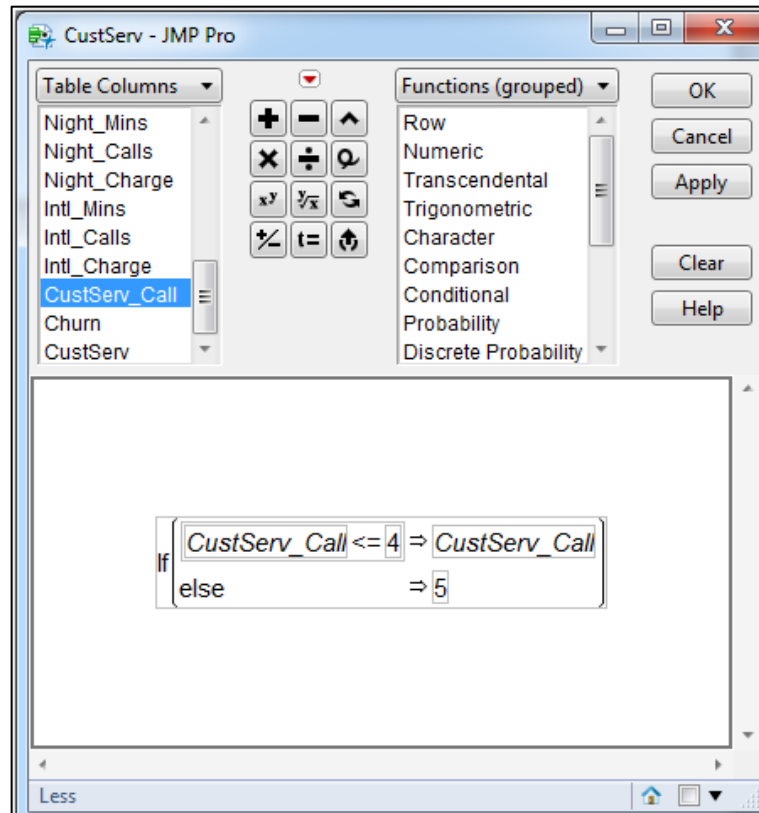


Let’s create a new nominal variable called CustServ, so that all the counts for 5 and greater are collapsed into a single cell:

Click **Cols**→**New Columns**. For column name type CustServ, for Modeling Type change it to **Nominal** and then click the drop arrow for Column Properties and click **Formula**. In Formula dialog box, click on **Conditional**→**If**. Then, in the top expr click on CustServ_Call and type ≤ 4 and in the top then clause

click on CustServ_Call. For the else clause type in 5. See Figure 5.21. Click **OK** and click **OK**.

Figure 5.21 Creating the CustServ variable



Now drop the CustServ_Call variable from the Logistic Regression and add the new CustServ nominal variable, which is equivalent to adding some dummy variables. Our new value of -LogLikelihood is 970.6171, which constitutes a very substantial improvement in the model.

Another possible important way to improve a model is to introduce interactions terms, that is, the product of two or more variables. Best practice would be to consult with subject-matter experts and seek their advice. Some thought is necessary to determine meaningful interactions, but it can pay off in substantially improved models. Thinking about what might make a cell phone customer want to switch to another carrier, we have all heard a friend complain about getting charged an outrageous amount for making an international call. Based on this observation, we could conjecture that customers who

make international calls and who are not on the international calling plan might be more irritated and more likely to churn. A quick bivariate analysis shows that there are more than a few such persons in the dataset. **Tables**→**Tabulate**, drag and drop Intl_Plan to “Drop zone for columns”, drag and drop Intl_Calls to “Drop zone for rows” and choose “Add Grouping Columns”. Observe that almost all customers make international calls, but most of them are not on the international plan (which gives cheaper rates for international calls). For example, for the customers who made no international call, all 18 of them were not on the international calling plan. For the customers who made 8 international calls, 106 were not on the international calling plan, and only 10 of them were. There is quite the potential for irritated customers here! This is confirmed by examining the output from the previous logistic regression. The parameter estimate for “Intl_Plan[no]” is positive and significant. This means that when a customer does not have an international plan, the probability that he churns increases.

Customers who make international calls and don’t get the cheap rates are perhaps more likely to churn than customers who make international calls and get cheap rates. Hence the interaction term Intl_Plan*Intl_Mins might be important. To create this interaction term, we have to create a new dummy variable for Intl_Plan, because the present variable is not numeric and cannot be multiplied by Intl_Mins:

First, click on the Intl_Plan column in the data table to select it. Then select **Cols**→**Recode**. Under **New Value**, where it has **No**, type in **0** and right below that where it has **Yes**, type **1**. For the “**In Place**” pull down menu, select “**New Column**” and click “OK”. The new variable Intl_Plan2 is created. However, it is still nominal. Rightclick on this column, and under “**Column Info...**” change the Data Type to “Numeric” and the Modeling Type to “Continuous”. Click “OK”. (This variable has to be continuous so that we can use it in the interaction term, which is created by multiplication; nominal variables cannot be multiplied.)

To create the interaction term:

Click **Cols**→**New Column** and call the new variable IntlPlanMins. Under Column Properties select Formula. Click on Intl_Plan2, click on the times sign (“x”) in the middle of the dialog box, click on Intl_Mins and click **OK**. Click **OK** again.

Now add the variable IntlPlanMins as the 11th independent variable in multiple logistic regression that includes CustServ and run it. The variable IntlPlanMins is significant, and the -LogLikelihood has dropped to 947.1450, as shown in Figure 5.22. This is a substantial drop for adding one variable. Doubtless other useful interaction terms could be added to this model, but we will not further pursue this line of inquiry.

Figure 5.22 Logistic Regression Results with Interaction Term Added

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	432.0016	65	864.0033	<.0001*
Full	947.1450			
Reduced	1379.1467			

Now that we have built an acceptable model, it is time to validate the model. We have already checked the Lack Of Fit, but now we have to check linearity of the logit. From the red arrow, select **Save Probability Formula** which adds four variables to the dataset: Lin[0] (which is the logit), Prob[0], Prob[1] and the predicted value of Churn, Most Likely Churn". Now we have to plot the logit against each of the continuous independent variables – the categorical independent variables do not offer much opportunity to reveal nonlinearity (plot some and see this for yourself). All the relationships of the continuous variables can be quickly viewed by generating a scatterplot matrix and then clicking the red triangle and Fit Line. Nearly all the red fitted lines are horizontal or near horizontal. For all of the logit vs. independent variable plots, there is no evidence of nonlinearity.

We can also see how well our model is predicting by examining the confusion matrix which is shown in Figure 5.23.

Figure 5.23 Confusion matrix

	Actual	Predicted
Training	0	1
0	2749	101
1	326	157

The actual number of churners in the dataset is $326+157 = 483$. The model predicted a total of 258 ($=101+157$) churners. The number of bad predictions made by the model is $326+101 = 427$, which comprises 326 predicted not to churn that actually did churn, and 101 predicted to churn that did not churn. Further, observe in the Prob[1] column of the data table that we have the probability that any customer will churn. Right click on this column and select sort; this will sort all the variables in the dataset according to the probability of churning. Scroll to the top of the dataset. Look at the Churn column. It has

mostly ones and some zeroes here at the top, where the probabilities are all above 0.85. Scroll all the way to the bottom and see that the probabilities now are all below 0.01, and the values of Churn are all zero. We really have modeled the probability of churning.

Now that we have built a model for predicting churn, how might we use it? We could take the next month's data (when we do not yet know who has churned) and predict who is likely to churn. Then these customers can be offered special deals to keep them with the company, so that they do not churn.

References

- [1] Hosmer, D. W. and S. Lemeshow. *Applied Logistic Regression*. New York ; Chichester ; Brisbane : J. Wiley and Sons, 2001.
- [2] Pollack, R. *Data Mining Methods and Applications (Discrete Mathematics & Its Applications)*, Lawrence, K., Kudyba, S. and R. Klimberg, (eds.), Taylor and Francis Publishers, Dec 2007.

Exercises

1. Consider the logistic regression for the toy dataset, where π is the probability of passing the class:

$$\log \left[\frac{\hat{\pi}}{1 - \hat{\pi}} \right] = 25.60188 - 0.363761 \text{MidtermScore}$$

Consider two students, one who scores 67% on the midterm and one who scores 73% on the midterm. What are the odds that each fails the class? What is the probability that each fails the class?

2. Consider the first logistic regression for the Churn dataset, the one with 10 independent variables. Consider two customers, one with an international plan and one without. What are the odds that each churns? What is the probability that each churns?
3. We have already found that the interaction term IntlPlanMins significantly improves the model. Find another interaction term that does so.

4. Without deriving new variables such as CustServ or creating interaction terms such as IntlPlanMins, use a stepwise method to select variables for the Churn dataset. Compare your results to the bivariate method used in the chapter; pay particular attention to the fit of the model and the confusion matrix.
5. Use the Freshmen1.jmp dataset and build a logistic regression model to predict whether or not a student returns. Perhaps the continuous variables Miles from Home and Part Time Work Hours do not seem to have an effect. See whether turning them into discrete variables makes a difference (*e.g.*, turn Miles from Home into some dummy variables, 0-20 miles, 21-100 miles, more than 100 miles).

About These Authors:



Ron Klimberg is professor at the Haub School of Business at Saint Joseph's University in Philadelphia, PA. Before joining the faculty in 1997, he was professor at Boston University, an operations research analyst for the Food and Drug Administration, and a consultant. His primary research interests lie in the areas of multiple criteria decision making, DEA, facility location, data visualization and data mining. Ron was the 2007 recipient of the Tengelmann Award for his excellence in scholarship, teaching, and research. He received his PhD from The Johns Hopkins University..



B.D. McCullough is professor at the LeBow College of Business at Drexel University in Philadelphia, PA. Prior to joining Drexel, he was a senior economist at the Federal Communications Commission and an assistant professor at Fordham University. His research fields include applied econometrics and time series, accuracy of statistical and econometrics software, replicability of research, and data mining. He received his PhD from the University of Texas at Austin.

Learn more about these authors by visiting their author pages, where you can download free chapters, access example code and data, read the latest reviews, get updates, and more:

<http://support.sas.com/klimberg>

<http://support.sas.com/mccullough>

ACCELERATE YOUR SAS[®] KNOWLEDGE WITH SAS BOOKS.

Visit the [SAS[®] Press author pages](#) to learn about our authors and their books, download free chapters, access example code and data, and more.

Browse our [full catalog](#) to find additional books that are just right for you.

Subscribe to our [monthly e-newsletter](#) to get the latest on new books, documentation, and tips—delivered to you.

Browse and search [free SAS documentation](#) sorted by release and by product.

Email us: sasbook@sas.com
Call: 800-727-3228

