Modeling Count Data using Stata

Poisson and Negative Binomial Regression Techniques

Najib A. Mozahem

Contents

Chapter 1

Modeling Count Data - The Theory

1.1 Introduction

Perhaps the most known and used regression technique is linear regression, where the dependent variable is continuous in nature. For example, if the dependent variable was salaries, then we can use linear regression, because a salary can take on any value in a certain interval. Linear regression can also be used when the dependent variable is student grades where the grade can be any value between zero and 100, including decimal values. As you know, any regression model makes certain mathematical assumptions. If these assumptions are violated, then the use of the regression model is questioned. This is why we cannot use linear regression when the dependent variable is not continuous in nature. When the dependent variable can take on only two values for example (i.e. a student either passes or fails a course) then

linear regression cannot be used. In such a case we can use logistic regression. What about when the dependent variable represents a count? For example, what if the dependent variable is the number of courses that a student has failed in? In this case the dependent variable cannot be negative since the minimum value is zero. In addition, the dependent variable cannot take on decimal values since students cannot fail in 2.5 courses for example. Another issue is that the maximum number of failed courses has a ceiling. A student cannot fail in 100 courses since the entire program does not have 100 courses. When the dependent variable is created by counting the times that a certain event has happened, we use regression techniques that are referred to as count models. In order to understand how these techniques work, let us first consider a simple count table. As an example, consider the data displayed in Table 1.1. The table contains the records of twenty students. The data includes the total number of courses in which each student has failed, the total number of courses in which the student has passed, and the college in which the student is enrolled. What we want to see is whether students in engineering fail in more courses than business students. This means that the variable that we intend to study is the number of courses in which the student has failed. This is an example of a count variable. To help us answer our question we can create a count table that summarizes the data.

1.2 Count Tables

Consider Table 1.2, which summarizes the data shown in Table 1.1. We see that out of a total of 243 courses taken by business students, there are 35 failing grades and 208 passing grade. For engineers, out of a total of 189

Table 1.1: Records of students.

Number of failed courses	**Number of passed courses**	**College**
3	20	Business
0	9	Business
5	25	Business
7	21	Business
1	26	Engineering
2	15	Business
2	13	Business
0	18	Business
4	23	Engineering
2	8	Engineering
5	23	Engineering
3	14	Engineering
8	19	Business
0	24	Business
1	5	Engineering
2	13	Engineering
3	17	Business
4	21	Engineering
5	27	Business
9	25	Engineering

courses, there are 31 failing grades and 158 passing grades. Therefore, the count of failed courses for engineers is actually smaller than the count for business (31 for engineers and 35 for business). Does this mean that the failure rate in business is higher? The answer is no because we did not take

Table 1.2: Count table showing number of failed and not failed courses for students in the business and engineering schools.

	College		
Failed	**Business**	**Engineering**	**Total**
No	208	158	366
Yes	35	31	66
Total	243	189	432

into account the total number of courses. In other words, we need to calculate the proportion of courses which resulted in a failed grade. This means that we need to look at the risk.

1.2.1 Risk

Now that we have seen Table 1.2, we would like to calculate the risk that a student in business would fail a course and to compare it to the risk that a student in engineering would fail a course. Risk here indicates the probability of the event happening. For business students for example, out of a total of 243 courses, 35 resulted in a failed grade. This means that the probability of failure, or the risk of failure, is $35/243 = 0.144$. For an engineering student, the risk is $31/189 = 0.164$. Therefore, we see that this risk of failure for engineers is greater than the risk of failure of business students.

1.2.2 Incidence-rate Ratio

We now know that the risk of failure for business students is larger than the risk of failure of engineering students. To directly compare the two risks we can calculate the incidence-rate ratio, which is simply the ratio of the two numbers:

$$Riskratio = 0.144/0.164 = 0.878$$

What does this value mean? Simply that the likelihood of a business student failing a course is 0.878 times the likelihood of an engineering student failing the course. We could have calculated the incidence-rate ratio by dividing the risk for engineers by the risk for business students:

$$Riskratio = 0.164/0.144 = 1.139$$

This means that the likelihood of an engineering student failing a course is 1.139 times the likelihood of a business student failing a course.

1.2.3 2x3 Tables

The above logic is maintained even when we have more than two groups. Consider Table 1.3 for example. This table contains the same information as Table 1.2 with the addition of a new college which is the life sciences college. We already know that the risk of failure for business and engineering students are 0.144 and 0.164 respectively. The risk for students in the life sciences school is $20/202 = 0.099$, which is smaller than the other two risks. Using these risks, we can also calculate the risk ratios. What is different is this case is that we can calculate two different risk ratios:

$$Riskratio_1 = 0.144/0.164 = 0.878$$

Table 1.3: Count table showing number of failed and not failed courses for students in the business, engineering, and life sciences schools.

	College			
Failed	**Business**	**Engineering**	**Life Scences**	**Total**
No	208	158	182	548
Yes	35	31	20	86
Total	243	189	202	634

$$Riskratio_2 = 0.099/0.164 = 0.604$$

The first ratio is comparing the risk of business students to the risk of engineering students, while the second ratio is comparing the risk of life sciences students to the risk of engineering students. In both cases, the referent group is engineering students. It is up to you to pick and choose the referent category that suits your goals. In our case, the two risk ratios above are less than one, which indicates that the risk for both business and life sciences students is smaller than that of engineering students.

The above exercise is useful when we want to compare the risk across certain groups. This type of analysis however will not take us very far. The reason is that usually, we are interested in studying the effect that several variables have on the probability of the outcome. What if we wanted to see whether the risk of failure was affected by the college, gender, and the GPA, all at the same time? In this case, we need to use regression models.

1.3 Poisson Regression

First, you need to understand that there are several count models to choose from. The choice of the model depends on the data that we are analyzing. Usually, we start the analysis by assuming a Poisson model since this is considered to be the basic count model. In this type of regression, we are interested in the number of occurrences of a certain event, i.e. how many times a student will fail for example. As such, the dependent variable μ refers to the rate of occurrence or the expected number of times an event will occur. In order to visualize the distribution of a variable that follows the Poisson distribution, consider Figure 1.1 which shows the probability distribution functions for different average rates. The y-axis represents the probability that a certain event will happen a certain number of times. For example, looking at the graph for mean = 1, we see that the probability of the event not happening at all (zero) or happening once is high, while the probability of the event happening four times is very low.

If the expected average increases, i.e. the mean increases, then the probability of events happening more frequently also increases. This is why as the mean increases the graph starts to rise on the right side while dropping at the left side. The purpose of Poisson regression is to look at the factors that would increase the probability of an event happening more frequently.

In linear regression, the relationship between the dependent variable and the independent variable is formulated as:

$$y = ax + b$$

In the above equation, y is the dependent variable, x is the independent variable, a is the slope, and b is the y-intercept. One of the nice things about linear regression is how easy it is to interpret the relationship between the dependent variable and the independent variable. As an example, assume that we have the following linear equation:

$$y = 3x + 2$$

If x is equal to 2, y will be equal to 8, and if x is equal to 3, y will be equal to 11. Note that for every one unit increase in x, the value of y increases by 3, which is the value of the slope. This is the definition of the slope. It is the amount by which the dependent variable changes when the independent variable increases by one. The slope is important for two reasons. The first reason relates to the sign. If the slope is positive, then any increase in the independent variable will lead to an increase in the dependent variable. The more I eat, the heavier I get. If the slope is negative, then an increase in the independent variable will lead to a decrease in the dependent variable. The

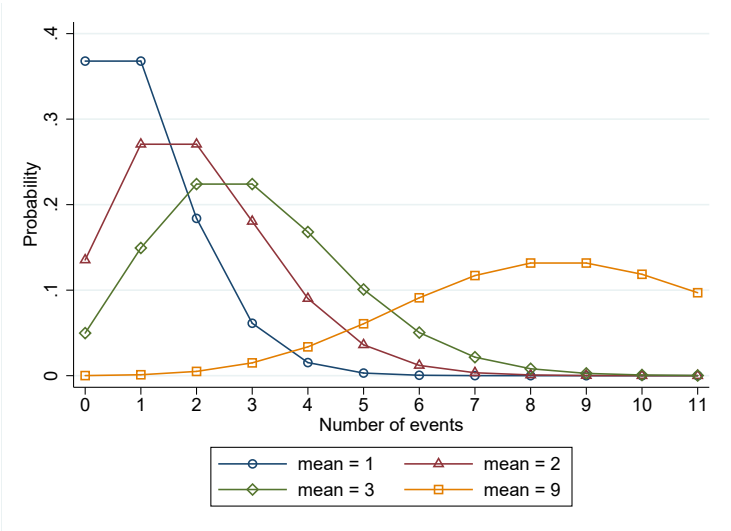

Figure 1.1: The Poisson distribution for different means.

more I buy food, the less money I have.

The second reason relates to the magnitude of the slope. The larger the magnitude of the slope, the greater the effect that the independent variable has on the dependent variable. If the slope is 2, then a one unit increase in the independent variable will result in an increase of 2 in the dependent variable. If, however, the slope is 10, then a one unit increase in the independent variable will result in an increase of 10 in the dependent variable. So the sign of the slope tells us about the direction of the relation and the magnitude tells us about the magnitude of the effect that one variable might have on the other.

Unfortunately, in Poisson regression things are not that simple. The reason is that the Poisson regression model has the following form:

$$\ln(\mu) = ax + b$$

As already mentioned, μ is the rate of occurrences, which is the dependent variable. The above equation is linear, but instead of having the dependent variable on the left hand side we have the natural logarithm of the dependent variable. This means that the slope a represents the amount by which $\ln \mu$ increases when x increases by one unit. As you can see, this is not a natural way of interpreting things. Fortunately, there is something that we can do to make the interpretation more intuitive. All we need to do is to take the exponential of both sides:

$$e^{\ln(\mu)} = e^{ax+b}$$

$$\mu = e^{ax+b}$$

There is nothing complicated in what we did. We know from algebra that an equality is maintained when we perform the same operation to both sides. In our case, we first took the exponent of both sides. We then took advantage of the rule $e^{\ln(k)} = k$.

This new form is better because now the dependent variable, which is the rate μ, is on the left side. For example, if a is positive, when x increases the term e^{ax+b} will increase. Since this term is equal to the rate of occurrence of the event, this means that the number of times that the event is expected to occur will also increase. On the other hand, when a is negative, when x increases the expected number of occurrences will decrease.

1.3.1 Continuous Variables

Let us take an example. Assume that we perform Poisson regression where the dependent variable is the number of courses in which a student has failed and the independent variable is the GPA of the student. Basically, we want to see if having a higher GPA predicts fewer course failures. Assume that once the model was fit that we get the following equation:

$$\ln(\mu) = -0.099(GPA) + 7.091$$

What this means is that when the GPA of the student increases by one, the function $\ln(\mu)$ decreases by -0.099. Since, as we said, this is hard to

understand, let's consider the other more intuitive form:

$$\mu = e^{-0.099(GPA)+7.091}$$

Now consider two students, one with a GPA of 77 and the other with a GPA of 78. According to our model, the expected number of withdrawals for each is:

Student with a GPA of 77: $\mu = e^{-0.099(77)+7.091} = 0.587$

Student with a GPA of 78: $\mu = e^{-0.099(78)+7.091} = 0.532$

This means that the expected number of failed courses for a student with a GPA of 77 is 0.587, and the expected number of failure courses for a student with a GPA of 77 is 0.532. To compare these two numbers, we can divide them in order to find the incidence-rate ratio:

Incidence-rate ratio: $0.532/0.587 = 0.906$

What this means is that the expected count for a student with a GPA of 78 is 0.906 times the expected count of a student with a GPA of 77. The great news is that 0.906 is actually $e^{-0.099}$. We now have a very intuitive interpretation of the slope a. When we fit a Poisson model and obtain a value for the coefficient associated with an independent variable, we know that when the independent variable x increases by one unit, the expected number of occurrences is multiplied by e^a. When a is positive, $e^a > 1$, which means that the expected number of occurrences increases when x increases. When a is negative, $e^a < 1$, which means that the expected number of occurrences decreases when x increases.

As a recap, when we fit a Poisson model, we are finding a line with the equation $ax + b$, just like in linear regression. The difference however is in the interpretation of the coefficient of x. In linear regression, when x increases by one unit, the dependent variable increases by the magnitude of a. In Poisson regression, when x increases by one unit, the expected number of occurrences are multiplied by e^a. If a is zero we have $e^0 = 1$, which means that the expected number of occurrences are multiplied by one, so they do not change. This means that x does not affect the expected number of occurrences. If a is greater than zero, then $e^a > 1$, which means that the expected number of occurrences are multiplied by a number greater than one, so they increase. If a is less than zero, then $e^a < 1$, which means that the expected number of occurrences are multiplied by a number that is less than one, so it decreases.

As another illustration, assume that we fit a Poisson model where the dependent variable is the number of customers that entered the store today, and where the independent variable is the number of advertisements that were ran on radio the preceding day. Once we fit the model we get the following results:

$$\ln(\mu) = 0.447(ads) + 0.241$$

Here, μ is the expected number of customers that will enter the shop. What does this output mean? Since the value of the coefficient associated with the independent variable, which is ads, is 0.447, this means that when ads increases by one, the expected number of customers is multiplied by $e^{0.447} = 1.564$. This means if the shop runs five radio ads the expected number of

customers that will come is 1.564 times the expected number of customers if it runs four ads.

As another example, consider that we fit a Poisson regression model where the dependent variable is the number of times that a student goes out with his or her friends during the week and the independent variable is the student's grades. The output of the model is the following:

$$\ln(\mu) = -0.073(grades) + 6.629$$

Here, the coefficient is negative. Since $e^{-0.073} = 0.930$, the output indicates that the expected number of times that a student goes out during the week are multiplied by 0.93 (so they decrease) when grades increase by a single unit. This means that students with higher grades go out fewer times during the week.

As you can see, when the coefficient is positive, the expected count increases, and when the coefficient is negative, the expected count decrease. Since we are mostly interested in the exponential of the coefficient, and not the coefficient itself, statistical software packages allows us to directly display the value e^a instead of displaying the value of a. In that case, when e^a is greater than one, the expected count increases, and when e^a is less than one, the expected count decreases.

1.3.2 Binary Variables

So far, the independent variable has been numerical in nature. Sometimes however, including variables that are not numeric in nature is necessary. For example, what if we wanted to investigate whether the count of failed courses could be explained by the gender of the students? Here, the variable gender is not numeric. It is categorical, in that it divides the observations into categories. Since biological gender is either male or female, there are two categories in which each student might fall.

In such a case, we can create a binary variable to represent the two categories. A binary number takes on the values of zero or one. We next assign each of these values to a category. Let us assign a zero to males and a one to females. The data is shown in Table 1.4. Now that the variable gender has been quantified, it is possible to include it in a regression model. The result of running a Poisson model would be again in the form:

$$\ln(\mu) = ax + b$$

If we use a statistical software to run the model, we will get the following output:

$$\ln(\mu) = 0.495(gender) + 0.916$$

We already know how to interpret the coefficients of continuous variables, such as grades and number of advertisements. However, what does it mean that the coefficient of gender is 0.495? Remember that for males the value

Table 1.4: Records of students.

Number of failed courses	**Gender**	**Binary**
2	male	0
0	male	0
3	male	0
0	female	1
2	female	1
3	female	1
8	female	1
0	male	0
5	female	1
7	male	0
5	female	1
3	male	0
1	male	0
1	male	0
2	female	1
9	female	1
5	female	1
4	male	0
4	male	0
2	female	1

of gender is zero, while for females the value of gender is one. In order to calculate the expected count for a male and a female student, we need to use the form:

$$\mu = e^{0.495(gender)+0.916}$$

We can now calculate the expected count for each student:

Male: $\mu = e^{0.495(0)+0.916} = 2.499$

Female: $\mu = e^{0.495(1)+0.916} = 4.1$

From these expected counts, we can calculate the incidence-rate ratio:

Incidence-rate ratio: $4.1/2.499 = 1.64$

This means that females have higher expected count than males. The nice thing is that the number 1.64 happens to be $e^{0.495}$. This means that when we are dealing with binary variables, the exponent of the coefficient is the incidence-rate ratio when we compare an individual who belongs to the group that is assigned a value of one and an individual who belongs to the group that is assigned the value zero. In our case, since males were assigned a value of zero, the exponent of the coefficient is the incidence-rate ratio that we obtain when we divide the expected count of females by the expected count of males. In other words, since the coefficient is 0.495, the expected count for females is 1.64 times the expected count for males.

Let us take another example. Assume that we run a Poisson regression model where the dependent variable is the number of goals a player scores, and where the independent variable is whether the player got a good night sleep the night before or not. The independent variable is binary (either you get a good night sleep or not), so we need to assign zero to a category and a one to the other category. In our case, let's assign a zero to not getting a

good night sleep and a one to getting a good night sleep. We fit the model and get the following result:

$$\ln(\mu) = 1.104(sleep) + 0.357$$

This means that the expected number of goals scored by those who get a good night sleep is $e^{1.104} = 3.016$ times the expected number of goals scored by those who did not get a good night sleep.

1.3.3 Multiple Independent Variables

Now that we have seen how to interpret the output from Poisson regression when there is a single independent variable, let us see what changes when there are two independent variables. Table 1.5 shows the records for students. The table includes the dependent variable which is the number of courses in which the student has failed and the independent variables gender and GPA. Therefore, we have one binary variable and one continuous variable. In this case, we want to see if the dependent variable, which is the number of failed courses, depends on the gender of the student and on the GPA of the student. The equation of this model is:

$$\ln(\mu) = a_1x_1 + a_2x_2 + b$$

Each independent variable has its own coefficient now. If we run the model, the output will be:

Table 1.5: The case of two independent variables.

Number of failed courses	**GPA**	**Gender**	**Binary**
2	80	male	0
0	95	male	0
3	77	male	0
0	90	female	1
2	75	female	1
3	72	female	1
8	60	female	1
0	82	male	0
5	74	female	1
7	69	male	0
5	69	female	1
3	79	male	0
1	81	male	0
1	78	male	0
2	83	female	1
9	62	female	1
5	72	female	1
4	70	male	0
4	71	male	0
2	87	female	1

$$\ln(\mu) = -0.086(GPA) + 0.033(gender) + 7.464$$

Let us now calculate the expected count for two students where both of them

have a GPA of 74, but one is male and the other is female. First, we use the more intuitive form of the equation:

$$\mu = e^{-0.086(GPA)+0.033(gender)+7.464}$$

Male: $\mu = e^{-0.086(74)+0.033(0)+7.464} = 3.004$

Female: $\mu = e^{-0.086(74)+0.033(1)+7.464} = 3.105$

This means that the incidence-rate ratio is:

Incidence-rate ratio: $3.105/3.004 = 1.034$

A simpler way to get this value is just to calculate the exponent of the coefficient, $e^{0.033} = 1.034$. This shows that even when there are several independent variables, the coefficients retain their meanings. Therefore, to find the difference between two groups of students, just calculate e^{a_1}. The implication of this is that in a multiple regression model where there are several independent variables, when we want to investigate the effect that an independent variable has on the dependent variable, we just need to take into consideration the coefficient of the independent variable, given that the rest of the variables do not change.

To further illustrate this, let us now calculate the expected count for two female students, one of whom has a GPA of 79 and another who has a GPA of 80:

GPA of 79: $\mu = e^{-0.086(79)+0.033(1)+7.464} = 2.02$

GPA of 80: $\mu = e^{-0.086(80)+0.033(1)+7.464} = 1.853$

This means that the incidence-rate ratio is:

Incidence-rate ratio: $1.853/2.02 = 0.917$

This is also obtained by finding the exponent of the coefficient, $e^{-0.086} = 0.917$. Therefore, we see that when GPA increases by one, the expected count is multiplied by 0.917, which means that the expected count decreases.

This same logic applies whether we have three independent variables, four independent variables, or even nine independent variables. It also doesn't matter whether the variables are binary or continuous. The coefficient of each independent variable gives us information about the relationship between the independent variable and the dependent variable. All we have to do is to take the exponent of the coefficient in order to calculate the effect that the independent variable has on the odds of the event happening.

1.3.4 Categorical Variables with more than Two Categories

If you recall, Table 1.1 presented data that included the variable college, which took on two values, business and engineering. In such a case, when we perform Poisson regression, we use a binary variable since the variable college can take on one of two values. What if we had a categorical variable that divided the observations into more than two groups? In this case, we cannot use a single binary variable because there are three groups instead of two. As an example, consider the data displayed in Table 1.6. This is the same data that we used in Table 1.1 except that nine new records have been added for students in the college of life sciences (the count table for Table

Table 1.6: Records of students.

Number of failed courses	**College**
3	Business
0	Business
5	Business
7	Business
1	Engineering
2	Business
2	Business
0	Business
4	Engineering
2	Engineering
5	Engineering
3	Engineering
8	Business
0	Business
1	Engineering
2	Engineering
3	Business
4	Engineering
5	Business
9	Engineering
0	Life sciences
1	Life sciences
1	Life sciences
3	Life sciences
5	Life sciences
4	Life sciences
3	Life sciences
1	Life sciences
2	Life sciences

Table 1.7: Coding the categorical variable.

	$\mathbf{x_1}$	$\mathbf{x_2}$
Engineering	0	0
Business	1	0
Life sciences	0	1

1.6 was actually used in Table 1.3). This means that the variable college is no longer binary, since it can take on more than two values.

What we can do in this case, is to use more than one binary variable, as illustrated in Table 1.7. If you look at the column for the variable x_1, you will notice that the variable takes a value of one for business, and zero otherwise. The other binary variable, x_2, takes on a value of one for life sciences and zero otherwise. How did we know that we need three binary variables? The number of binary variables needed is the number of categories minus one. In our case, we have three categories, so it is 3 – 1 = 2. Table 1.8 displays the result of this coding exercise. The regression equation now becomes:

$$\ln(\mu) = a_1 x_1 + a_2 x_2 + b$$

For an engineering student, x_1 and x_2 are zero. For a business student, x_1 is one and x_2 is zero. For a life sciences student only x_1 is zero and x_2 is one. If we fit this model, the output will be:

$$\ln(\mu) = -0.079 x_1 - 0.438 x_2 + 1.237$$

Table 1.8: Records of students.

Number of failed courses	**College**	$\mathbf{x_1}$	$\mathbf{x_2}$
3	Business	1	0
0	Business	1	0
5	Business	1	0
7	Business	1	0
1	Engineering	0	0
2	Business	1	0
2	Business	1	0
0	Business	1	0
4	Engineering	0	0
2	Engineering	0	0
5	Engineering	0	0
3	Engineering	0	0
8	Business	1	0
0	Business	1	0
1	Engineering	0	0
2	Engineering	0	0
3	Business	1	0
4	Engineering	0	0
5	Business	1	0
9	Engineering	0	0
0	Life sciences	0	1
1	Life sciences	0	1
1	Life sciences	0	1
3	Life sciences	0	1
5	Life sciences	0	1
4	Life sciences	0	1
3	Life sciences	0	1
1	Life sciences	0	1
2	Life sciences	0	1

Let us now calculate the expected number of occurrences for each student. As usual, we use the more intuitive form of the equation:

$$\mu = e^{-0.079(x_1)-0.438(x_2)+1.237}$$

Engineering: $\mu = e^{-0.079(0)-0.438(0)+1.237} = 3.445$

Business: $\mu = e^{-0.079(1)-0.438(0)+1.237} = 3.184$

Life Sciences: $\mu = e^{-0.079(0)-0.438(1)+1.237} = 2.223$

We can now calculate the incidence-rate ratios in order to be able to compare different groups:

Business/Engineering=3.184/3.445 = 0.924

Life sciences/Engineering=2.223/3.445 = 0.645

The above means that the expected number of failed courses for business students is 0.924 times the expected number of failed courses for engineering students, and that the expected number of failed courses for life sciences students is 0.645 times the expected number of failed courses for engineering students. We can get the same values by calculating the exponents of the coefficients:

$$e^{-0.079} = 0.924 \, and \, e^{-0.438} = 0.645$$

We see that the exponent of the coefficient for each variable produces the incidence-rate ratio when we compare the group associated with the variable

to the base group, which is the group that is assigned the values of zero. In other words, in our example, engineering students are the base, or referent group, since they have a value of zero for both x_1 and x_2. Business students have a value of one for x_1, which means that the exponent of the coefficient of x_1 is the incidence-rate ratio of business students to engineering students. Life sciences students have a value of one for x_2, which means that the exponent of the coefficient of x_2 is the incidence-rate ratio of life sciences students to engineering students. Therefore, just like in the case of binary variables, the coefficient compares a group to another group. The only difference here is that there is more than one binary variable, where each is associated with a different group. In both cases, the referent group is the same.

1.3.5 Exposure

There is an issue here which you have probably not noticed. In the previous section, we calculated the incidence-rate ratios when the categorical variable took on three values, business, engineering, and life sciences. This was done using Poisson regression. However, earlier in this book, we had calculated the incidence rate ratios for the exact same data using the count table (Table 1.3). In both cases, although the same data was used, the results are different. To make comparison easier, Table 1.9 shows the different values for the incidence-rate ratios obtained using each method. Why are the results different? The answer is actually simple. When we calculated the incidence-rate ratios using the count tables, we took into consideration the total number of courses taken by each group of students. In the count tables section, we did not compare the total number of failed courses for business students with the total number of failed courses for engineering students.

Table 1.9: Comparison of incidence-rate ratios when using count tables and when using Poisson regression.

	Count tables	**Poisson regression**
Business / Engineering	0.878	0.924
Life sciences / Engineering	0.604	0.645

We compared the proportion of failed courses for business students with the proportion of failed courses for engineering students (this is why we were dividing the number of failed courses by the total number of courses). This is an important point because the larger the number of courses taken by a group, the larger the expected number of failures. For example, a student who has taken twenty courses in university has a higher probability of having failed one of these courses than a student who has only taken two courses so far. We say that the first student was exposed to the risk of failure for a longer period time. Using the same logic, a player who spends more time on the field is expected to score more goals than a player who does not spend less time on the field. Someone who has smoked for thirty years is expected to have been hospitalized more than someone who has been smoking for one year.

Because the concept of exposure is important, we need to tell the statistical package to take it into account when calculating the regression equation. So far, we have not been doing that. All what we have been doping is telling the statistical package to compare the number of occurrences while taking into consideration one or more independent variables.

Going back to the last regression model that we fit, we found that the equation was:

$$\ln(\mu) = -0.079x_1 - 0.438x_2 + 1.237$$

The above equation was obtained without taking into consideration the concept of exposure. Let us now tell the statistical software to take into account the exposure in each observation. In this case, the exposure is the total number of courses taken by each student. The data that includes the exposure variable is shown in Table 1.10. Now we need to tell the statistical software to run a Poisson regression model using the number of failed courses as a dependent variable and college as an independent variable while taking into account that different students have gone through a different number of courses. The following is the output of this model:

$$\ln(\mu) = -0.130x_1 - 0.505x_2 - 1.808$$

We see that the equation has changed. Let us now calculate the expected number of occurrences for each student using the new output. As usual, we use the more intuitive form of the equation:

$$\mu = e^{-0.130(x_1)-0.505(x_2)-1.808}$$

Engineering: $\mu = e^{-0.130(0)-0.505(0)-1.808} = 0.164$

Business: $\mu = e^{-0.130(1)-0.505(0)-1.808} = 0.144$

Life Sciences: $\mu = e^{-0.130(0)-0.505(1)-1.808} = 0.099$

We can now calculate the incidence-rate ratios in order to be able to compare

Table 1.10: Records that contain the exposure variable.

Number of failed courses	**College**	**Total number of courses**
3	Business	23
0	Business	24
5	Business	30
7	Business	28
1	Engineering	6
2	Business	17
2	Business	15
0	Business	18
4	Engineering	27
2	Engineering	15
5	Engineering	28
3	Engineering	17
8	Business	27
0	Business	9
1	Engineering	27
2	Engineering	10
3	Business	20
4	Engineering	25
5	Business	32
9	Engineering	34
0	Life sciences	17
1	Life sciences	16
1	Life sciences	20
3	Life sciences	28
5	Life sciences	32
4	Life sciences	26
3	Life sciences	17
1	Life sciences	18
2	Life sciences	28

different groups:

Business/Engineering: $0.144/0.164 = 0.878$

Life sciences/Engineering: $0.099/0.164 = 0.604$

These are the exact same values we got when we calculated the incidence-rate ratios using the count tables (refer to Table 1.10). This exercise should illustrate why when running a count model, you need to account for the different exposure times in which each subject was exposed to the risk of the event happening.

1.4 Negative Binomial Regression

Although the Poisson regression model is the basic count model, it actually rarely fits the data because of what is referred to as overdispersion. An important characteristic of the Poisson probability density function is that the mean and the variance are equal. This means that as the mean increases, so does the variability in the data, a characteristic that is called equidispersion. When this assumption is violated, we say that the data displays overdispersion.

In order to address this issue, a negative binomial regression model is used. This model accounts for overdispersion by adding the parameter alpha to the equation. To illustrate the difference, look at Figure 1.2. The figure shows the probability density function of both the negative binomial and the Poisson distributions (alpha is 1.5 in all plots). We see that in all plots, the negative binomial model allocates a higher probability for smaller counts,

specifically the probability of a count of zero. Therefore, you can think of the negative binomial model as a correction to the under prediction of zero, or low, counts.

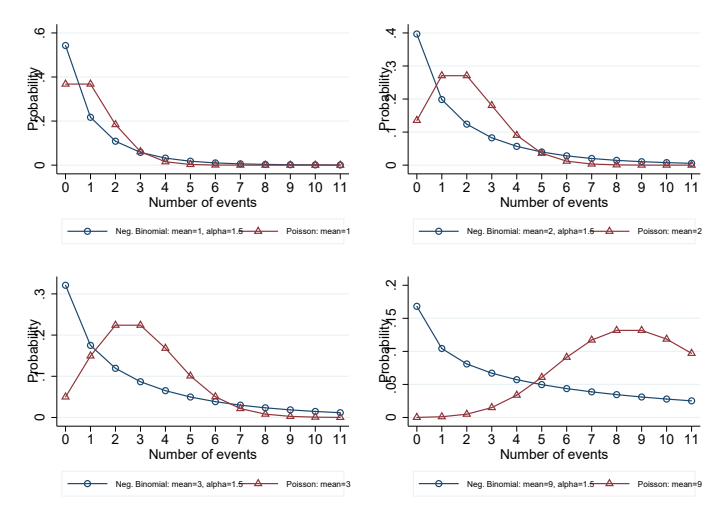

Figure 1.2: Poisson vs negative binomial at alpha = 1.5.

What implication does this have for us? Fortunately, very little. Since we are not interested in the math that goes on behind the scenes, all you need to know is that when we fit a Poisson regression model, we should always follow it up with a negative binomial regression model in order to test whether overdispersion exists. The beauty of it all is that everything we have covered with regards to the meaning of the coefficients when adding the independent variables still applies the exact way. The output of a negative binomial model is very similar to the output of a Poisson model. We get the coefficients of the variables, and when we calculate e^a we get the incidence-rate ratio. Nothing has changed.

So how do we test whether overdispersion exists? This is done by a likelihood-

ratio test that tests the null hypothesis that alpha (which is the extra parameter that is included in the negative binomial model) is equal to zero. If we fail to reject the null hypothesis, we conclude that alpha is zero, and when alpha is zero we end up with the Poisson model. This means that there is no reason to believe that there is overdispersion. If, on the other hand, we reject the null hypothesis that alpha is zero, we conclude that the overdispersion exists and that the negative binomial model should be used instead of the Poisson model.

As an example, consider the data shown in Table Table 1.10. We have already fit a Poisson regression model on this data while taking into consideration the different exposure of each subject. The resulting model was:

$$\ln(\mu) = -0.130x_1 - 0.505x_2 - 1.808$$

If we fit a negative binomial model to the same dataset (while accounting for exposure), we get the following equation:

$$\ln(\mu) = -0.130x_1 - 0.505x_2 - 1.808$$

This is the exact same equation as before. The likelihood-ratio test that is produced automatically by the statistical software tells us that the p-value of the null hypothesis is 0.5, which is much larger than the cut-off value of 0.05. This means that we cannot reject the null hypothesis (that alpha is zero). The conclusion is that we should use the Poisson model.

Usually, the difference between the parameters of the Poisson model and the negative binomial model is in the p-vales of the coefficients and not

the coefficients themselves. The p-values produced by a negative binomial model are larger than those produced by a Poisson model. The result is that a variable that is found to be statistically significant using a Poisson model will turn out not to be significant when a negative binomial model is used even though the value of the coefficient will be almost the same.

1.5 Truncated Models

Sometimes the data that we collect does not contain records with zero counts. An example is a dataset where the dependent variable is the number of semesters that a student spends in university. All students included in the dataset would have at least spent one semester in the university. As another example, I once received a dataset that contained the number of goals scored by each striker is various football (or soccer) leagues around the world. The data only contained records for players who had appeared on the scoresheet at least once, i.e. if a player never scored a goal, he was excluded from the list. This means that the minimum possible number of events was one and not zero. Figure 1.3 shows the histogram of the dependent variable, which is the total number of goals scored. In this case, the minimum is not one, but five. This means that the variable is truncated at the goals = 4 point.

In such cases, whether the data is zero-truncated or truncated at any other point, we use what is referred to as truncated models. These models take into account that a count that is less than a certain value is not possible. Once again, there is a truncated Poisson model and a truncated negative binomial

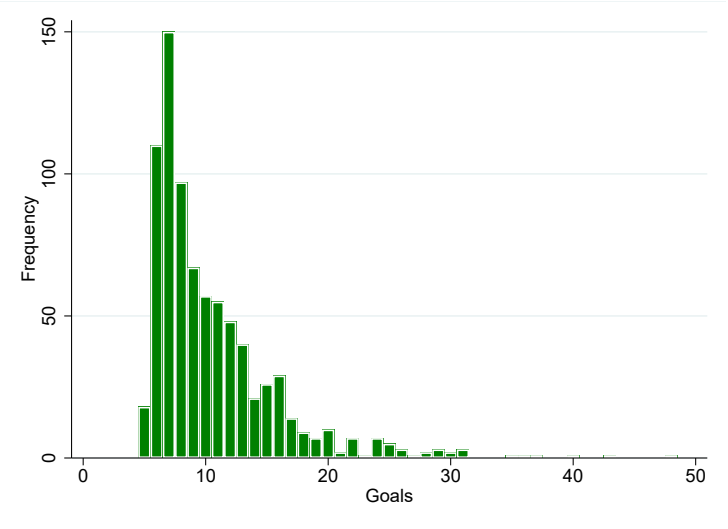

Figure 1.3: Histogram of a truncated variable.

model. Everything that we have said previously about the Poisson and the negative binomial models applies to the truncated models: the meaning of the coefficients, the incidence-rate ratios, and testing for overdispersion.

1.6 Zero-Inflated Models

When we discussed the negative binomial model, it was noted that the model corrects for the underprediction of zero counts. This is why in Figure 1.2 we saw that the probability of low counts, specifically zero, in the negative binomial model is greater than the probability of these counts in the Poisson model.

Sometimes the number of zeros in the dataset is much larger than what both the Poisson and the negative binomial model assume. In such a case, we say that the number of zeros is inflated, i.e. it is greater than usual. Why

would the number of zeros be inflated? This might be due to an underlying mechanism that is acting like a hurdle. As an example, assume that we want to model the number of heart attacks that men under 45 have suffered. In such a case, we would expect that most men at such a young age would not have suffered from a heart attack. A normal male under 45 years of age should not have suffered from a heart attack. This means that the dataset would contain a disproportionately large number of zeros. In this case, we can think of the dataset as containing two different types of men: healthy men who have a zero count, and men with health issues who have a count that is greater than zero.

When we have this type of situation, we use a zero-inflated model to account for the large number of zero counts. The math behind these models is not simple, so we will not get into it. However, it is very important to understand the idea behind these models, and this is what we will do now using the example of heart attacks for males under the age of 45.

- Step 1: If a male is healthy and living under normal conditions, we would expect that he has had no heart attacks, i.e. the count is zero. For males that are unhealthy for their age, or who lead a very stressful life, we would expect that the count would be greater than zero. This means that we can divide the observations into two groups. The members of the first group have a count of zero and the members of the second group have a count that is greater than zero. To model this situation, we can think of a dependent variable that is binary: an individual is either in the first group or in the second group. This is done by using a binary model that predicts the probability that the event has never occurred as opposed to it having occurred at least once.

- Step 2: After predicting whether an individual is in the first group (where the count is zero) or the second group (where the count is greater than zero), the analysis moves on to predicting the counts for those in the second group. This is done by using a count model, such as a Poisson model or a negative binomial model.

As can be seen above, zero-inflated models are thus made up of two parts. The first part is a logistic model that predicts whether someone has never experienced the event or has experienced it at least once. The second part models the number of times that the event has been experienced by those who have experienced the event at least once. This is why the output of zero-inflated models is divided into two parts, one part for each model. The output helps us understand what are the independent variables that lead to someone being in either of the two groups, and what are the independent variables that increase the frequency of the count for those who have experienced the event. The two sets of the variables need not be the same. This means that the variables that determine to which group an individual belong can be different than the variables that lead to a higher count.

Going back to our heart attack example, assume that we have a variable that indicates how much someone smokes, since exposure to tobacco is one of the causes of heart attacks. If someone has never smoked, this would increase the probability that they will belong to the group that contains the zero counts. It might also be the case that the variable smoke also leads to a higher count of heart attacks, but this is not necessary the case. We might find that smoking increases the probability of someone suffering at least once from a heart attack but that smoking does not increase the frequency of heart attacks for those who have suffered from it at least once.

As another example, consider a dataset that contains the count variable visits which records the number of times that a patient visits the doctor in the past year. Assume that this variable has an unusually large number of zeros. This means, that many of the patients have had no need to visit the doctor during the past year. Also, assume that recently the hospital has modified its internal policies in order to increase the efficiency of their patient care. The purpose of these reforms is to make sure that the hospital staff are able to respond quickly and efficiently to the needs of the patients thereby reducing the number of subsequent visits from the same patient. In this case, we would expect that the new reforms would reduce the frequency of the counts, but they would not have an effect on whether a patient initially visits or not. In other words, the reforms do not make people healthier, so people will continue to visit their doctor. However, the reforms make sure that when a patient visits, there will be less need for subsequent visits in the short term.

To make this clearer, consider the output that is shown in Table 1.11. This is a sample output from running a zero-inflated model where the dependent variable is the number of doctor visits during the past year. The variable male is a binary variable that indicates whether the individual is a male or not. The variable age records the age of the individual. Finally, the variable reform is a binary variable that indicates whether the observation is from the period before the reforms or after the reforms. We see that the output is divided into two parts, with the first part title “count” and the second part titled “inflate”. The “count” part is the regression output from modeling the count variable. The “inflate” part is the regression output from modeling the binary variable. The same three variables have been included in both parts in order to see which is significant and which is not. Starting with the

Table 1.11: Sample output of a zero-inflated model.

	Coefficient	$\mathbf{e^{Coefficient}}$	**P-value**
Count			
male	0.20	1.22	0.07
age	0.30	1.35	0.00
reform	-0.19	0.83	0.00
Inflate			
male	0.64	1.90	0.01
age	-0.55	0.58	0.00
reform	0.20	1.22	0.12

"count" part, we see that all three variables are significant (p-value is less than 0.05). Looking at t heir coefficients, we see that the coefficients of male and age are positive. This output is just regular count model output. As we were doing previously, all we need to do is to calculate $e^{Coefficient}$ in order to find the incidence-rate ratio. We see that males have an expected count that is 1.22 times that of females. This means that among those who visited the hospital, males visit the doctors more frequently. We also see that when age increases by one, the expected count is multiple by 1.35, so it increases. The coefficient of the variable reform is negative, with the incidence-rate ratio being 0.83. This means that among those who visited the hospital, patients who have visited the doctor after the reforms have an expected count that is 0.83 times the expected count of those who have visited the doctor before the reform. This means that the reforms have decreased the frequency of visits.

We now move onto the second part of the output which is titled "inflate". The reason why it is called inflate is that we are investigating which variables

are responsible for inflating the number of zeros. Looking at the p-values of the three variables, we see that only male and gender are significant. Since the "inflate" section presents the results of a binary regression, when we calculate $e^{Coefficient}$ we are finding the odds ratio (not the incidence-rate ratio). Looking at the table, we see that males have an odds that is 1.90 times the odds of females of not visiting the doctor. I have written "not visiting" in bold because what the second part is doing is testing which variables increase the odds of being in the zero group (this inflating the number of zeros) compared to the odds of being in the non-zero group. Since the coefficient of male is positive, it means that this variable increases the odds of being in the zero-group. Looking now at the coefficient of age, we see that it is negative. This means that when age increases, the odds of being in the zero-group decrease. Specifically, the odds of being in the zero-group are multiplied by 0.58 when age increases by one unit. This means that the older the patient is, the smaller the probability of him or her being in the zero group. Finally, the variable reform has a positive coefficient with an odds ratio of 1.22. This would mean that after the reforms, the odds of being in the zero-group are multiplied by 1.22 (so they increase). The result however is not significant.

As a recap, the results in Table 1.11 indicate that for the count model:

- Among those who have visited the hospital during the past year, the expected number of visits for males is 1.22 times the expected number of visits of females
- Among those who have visited the hospital during the past year, the expected number of visits are multiplied by 1.35 when age increases by one year

- Among those who visited the hospital during the past year, the expected number of visits for those who came after the reform is 0.83 times the expected number of visits of those who came after the reforms.

With regards to the logistic model, the results indicate that:

- The odds of males not visiting the doctor is 1.90 times the odds of females
- The odds of not visiting the doctor in the past year are multiplied by 0.58 when age increases by one year
- The reforms have no effect on the probability of a patient having visited the doctor during the past year

This example illustrates how the variables that are used in both parts of the model can be different. The variable that will increase the probability that we end up in the group with counts greater than zero might or might not be also responsible for increasing the frequency of the counts.

1.7 Model Comparisons

As you see, count data present an interesting dilemma, in that there are several models to choose from. So now the question becomes, how do we determine whether to use a Poisson model, a negative binomial model, a zero-inflated Poisson model, or a zero-inflated negative binomial model? For-

tunately, there are several tests that we can use in order to help us make these decisions.

1.7.1 Comparing Predicted Values with Observed Values

In linear regression, one of the ways to see whether the model has a good fit or not is to plot the actual observed values of the dependent variable and the predicted values on the same graph. Ideally what we want is to see that the predicted values are very close to the observed values. We can use the same tool in count models. However, instead of plotting the observed probabilities and the predicted probabilities, we can plot the difference between them. This means that when we run the four models (Poisson, negative binomial, zero-inflated Poisson, and zero-inflated negative binomial), we calculate the probabilities as predicted by each model and then calculate the difference between these probabilities and between the observed probabilities. We then plot these differences on the same graph to see which model results in the smallest differences. Figure 1.4 presents such a graph.

We see that the Poisson regression model (PRM) produces large differences with the observed probabilities for the counts zero, one and two. Clearly, this is not the right model to use. Looking at the other three models, we see that the zero-inflated negative binomial model (ZINB) and the negative binomial regression model (NBRM) produce the smallest differences, since their graphs are the closest to the zero axis.

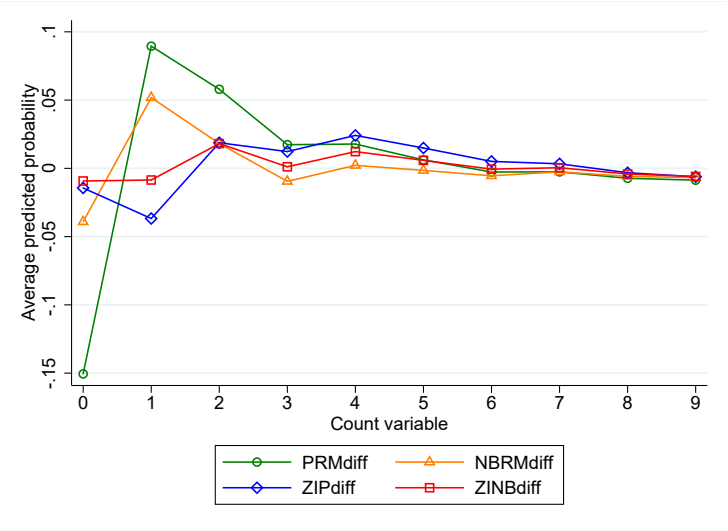

Figure 1.4: Plotting differences between average observed and predicted counts after fitting all models.

1.7.2 Likelihood-Ratio Test of Alpha

It was previously stated that a likelihood-ratio test helps us decide whether we should use a Poisson model or a negative binomial model. This test can be used to compare the Poisson model to the negative binomial model, and it can also be used to compare the zero-inflated Poisson model to the zero-inflated negative binomial model. If the test results in a p-value that is less than 0.05, then we reject the null hypothesis that alpha = 0 and we conclude that overdispersion exists, thus justifying the use of the negative binomial model.

1.7.3 Vuong Test

The likelihood-ratio test for alpha allows us to compare the Poisson model to the negative binomial model. What if we wanted to compare the Poisson model to the zero-inflated model? In this case we use the Vuong test. Like the likelihood-ratio test, if the Vuong test produces a p-value that is less than 0.05, we conclude that the zero-inflated model should be used.

1.7.4 AIC and BIC Statistics

Another group of tests that can be used to compare two models are the information criteria fit tests. These statistics are only used to compare models. This means that calculating these statistics for a single model does not inform us about the goodness of fit of the model. Instead, we calculate the statistics for the two or more models that we wish to compare. The most commonly used form of statistics in this group are the AIC and BIC. These statistics can be easily calculated by statistical software. When comparing two models, we tend to favor the one with smaller values of both AIC and BIC statistics.

1.8 Prediction

Once we have determined the best-fit model, it is time to calculated predicted values. In linear regression we use the model in order to predict the dependent variable. In logistic regression, we use the model in order to predict the probability of an event happening. In count models, we can do both. First,

we can predict the value of the dependent variable by predicting the number of events for certain values of the independent variables. For example, if we are modeling the number of courses in which a student has failed, we can use the best-fit model in order to predict the number of courses in which a student fails for several values of the independent variables. Take Figure 1.5 as an example. This figure shows the predicted number of events, which is course failures in this case, for students with different grades. As you can see, students with lower grades tend to fail more. As grades increase, the number of failed courses decrease. We also see that the graph starts to level off when the grades are above 80.

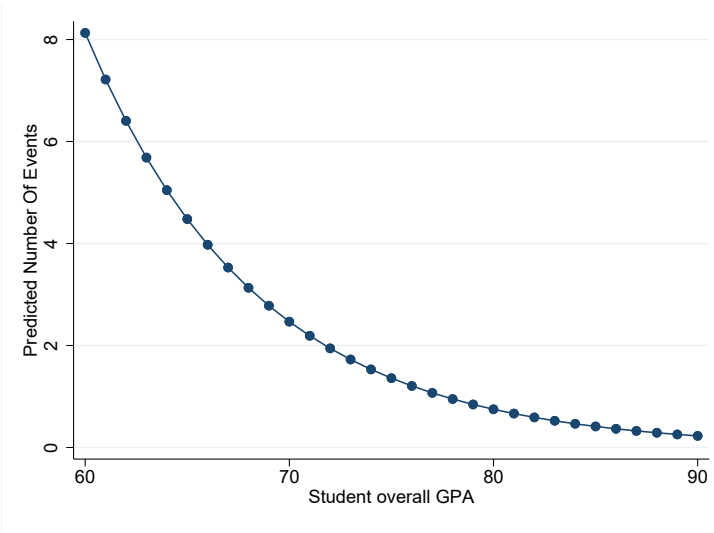

Figure 1.5: Predicting the number of events.

Second, we can use count models in order to predict the probabilities for several values of the count variable. For example, instead of predicting the number of events for certain values of a variable, we can predict the probability that the number of events will be a specific value. Figure 1.6 shows an

example of this. Unlike Figure 1.5, Figure 1.6 shows how the probability of failing in exactly four courses changes as the students' grades change.

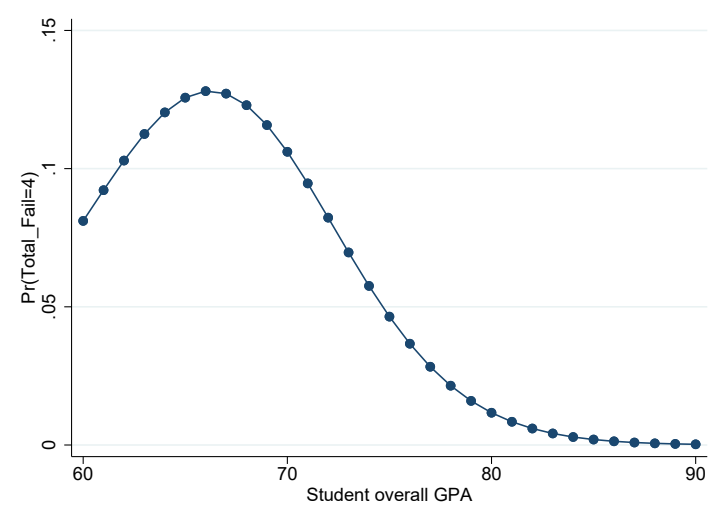

Figure 1.6: Predicting the probability that an event will occur four times.

Chapter 2

Modeling Count Data - Application

We now have the necessary tools that allow us to analyze a dataset where the dependent variable is a count. In this section, we will be looking at the dataset count_project.dta. This dataset contains the following variables:

- id: unique student identifier
- gpa: overall GPA of the student
- total_fail: the total number of courses in which the student has failed (this is the dependent variable)
- college: whether the student is in the engineering school or the business school (one means business, two means engineering)
- gender: whether the student is a male or a female (one means female, two means male)

- english: the average grade on all English courses taken by the student (data is taken from a non-English speaking country where the language of instruction in university is English)
- total_courses: the total number of courses taken by the student so far in the university

Before analyzing anything, the first thing that you should do is to install the SPost package. This package was authored by Long and Freese (2014) and it is without a doubt the single most important package when analyzing count models in Stata. The utility of this package will be clearly visible while we are analyzing the data. To install this package, you will need to execute the following commands in Stata:

```
net from http://www.indiana.edu/~jslsoc/stata/
net install spost13_ado
```

Once you have installed this package, we can start looking at the dataset.

2.1 Univariable Tests

The first thing that we should do when conducting regression analysis is to perform univariate analysis, where we try and uncover whether there is a relationship between the dependent variable and each independent variable separately. Once we have a good idea about the nature of these individual relationships, we can start building the model. In the case of count data, it is always a good idea to look at the histogram of the dependent variable in order to get an idea of the variable that we are dealing with:

```
. histogram total_fail
(bin=27, start=0, width=.81481481)
```

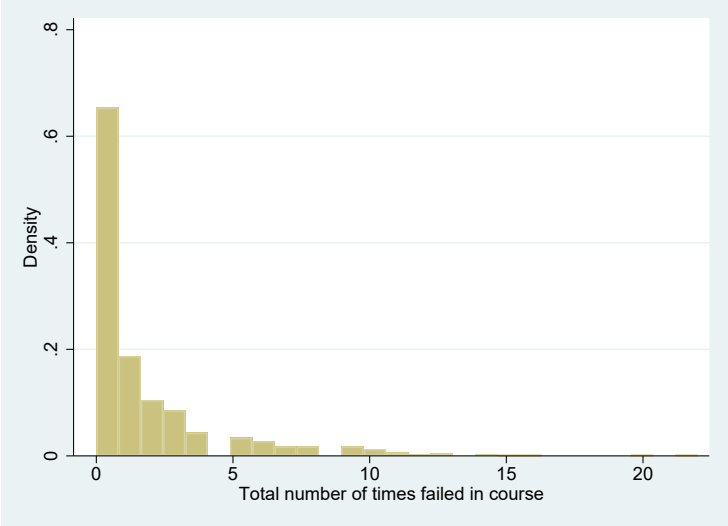

Figure 2.1: Histogram of the dependent variable.

The histogram is displayed in Figure 2.1. As we can see, the variable is not normally distributed, an outcome that is expected when looking at count data. We also see that there is a large number of zeros, which leads us to suspect that perhaps a zero-inflated model should be eventually used.

2.1.1 Continuous Variables

In linear regression, when we have a continuous independent variable, we start our analysis by plotting a scatter plot. Graphs are also useful as a starting step in count models, but their shape is different from what we are used to due to the nature of the dependent variable. For example, let us tell Stata to produce a scatter plot of the dependent variable total_fail and the continuous independent variable GPA:

```
. scatter total_fail gpa
```

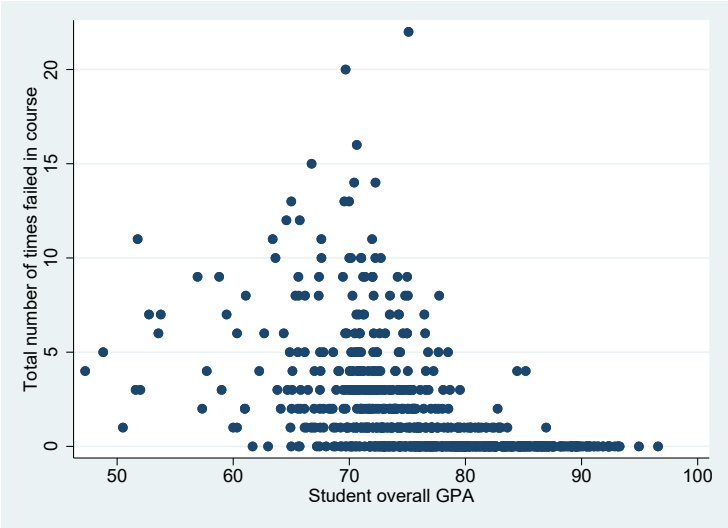

Figure 2.2: Scatterplot of total_fail and GPA.

The output is shown in Figure 2.2. The graph isn't visually appealing. The reason for this is that the outcome can only take on specific values. This is why we see that the points tend to cluster along the horizontal lines. However, if we look closely we see that when the GPA is around 80 and above, almost all of the points lie on the x-axis (the horizontal line that represents an outcome of zero). Therefore, it seems that students with high GPAs do not fail in courses.

In order to make things clearer, we can tell Stata to produce a smoothed scatter plot on top of the scatter plot:

```
. twoway (scatter total_fail gpa) (lowess total_fail gpa)
```

This command tells Stata to draw two plots on the same graph. This first is the regular scatterplot while the second is a locally smoothed plot, otherwise

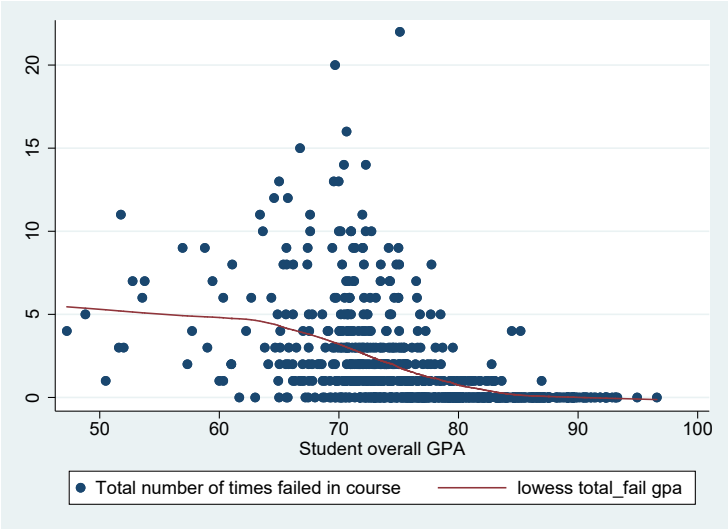

Figure 2.3: Smoothed scatterplot.

know as a loess graph. The resulting figure is shown in Figure 2.3. As can be seen in the figure, as the GPA increases, there is a visible drop in the count variable.

We next fit a Poisson model where we specify total_fail as the dependent variable and gpa as the independent variable:

```
. poisson total_fail gpa

Iteration 0:   log likelihood = -1495.7527
Iteration 1:   log likelihood = -1495.6504
Iteration 2:   log likelihood = -1495.6504

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =     817.86
                                                Prob > chi2       =     0.0000
Log likelihood = -1495.6504                     Pseudo R2         =     0.2147

------------------------------------------------------------------------------
  total_fail |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         gpa |  -.0952663   .0030879   -30.85   0.000    -.1013185   -.0892141
       _cons |   7.528266   .2187085    34.42   0.000     7.099605    7.956927
------------------------------------------------------------------------------
```

The poisson command tells Stata that we want to fit a Poisson model. The first variable after the command is the dependent variable followed by the independent variables. Looking at the output, we see that the coefficient of the gpa is -0.095. Recalling what we covered in the theory section, this means that when GPA increases by one, the expected number of occurrences is multiplied by $e^{-0.095} = 0.909$.

We can save ourselves a lot of time by specifying the irr option, which tells Stata to display the incidence-rate ratio instead of displaying the value of the coefficient:

```
. poisson total_fail gpa, irr

Iteration 0:   log likelihood = -1495.7527
Iteration 1:   log likelihood = -1495.6504
Iteration 2:   log likelihood = -1495.6504

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =     817.86
                                                Prob > chi2       =     0.0000
Log likelihood = -1495.6504                     Pseudo R2         =     0.2147

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         gpa |   .9091308   .0028073   -30.85   0.000     .9036452    .9146498
       _cons |   1859.877   406.7711    34.42   0.000     1211.488    2855.284
------------------------------------------------------------------------------
Note: _cons estimates baseline incidence rate.
```

We see that now the value 0.909 is displayed in the column titled “IRR”. We also see that the p-value is less than 0.05, hence the result is significant.

We have however, disregarded one important factor so far, and it is the exposure time. As you recall from the theory part of the course, we need to

account for the fact that different subjects were exposed to the probability of the event occurring for different period of time. In our dataset, the variable total_courses contains the total number of courses taken by the student. Stata makes it easy to include this as the exposure variable:

```
. poisson total_fail gpa, irr exposure(total_courses)

Iteration 0:   log likelihood =  -1146.808
Iteration 1:   log likelihood = -1146.7986
Iteration 2:   log likelihood = -1146.7986

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =    1345.83
                                                Prob > chi2       =     0.0000
Log likelihood = -1146.7986                     Pseudo R2         =     0.3698

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         gpa |    .868657   .0029586   -41.34   0.000     .8628775    .8744752
       _cons |   1694.273   408.1211    30.87   0.000     1056.681    2716.584
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------

Note: _cons estimates baseline incidence rate.
```

As we can see, the value of IRR for the variable gpa is now slightly different. We also see that there is a new row in the regression table that contains the elements ln(total_courses). From this point forward, we will be including the option exposure(total_courses) in all commands in order to take into account the different exposure levels of each student.

We next take a look at the other continuous variable in our dataset, which is the variable english:

```
. twoway (scatter total_fail english) (lowess total_fail english)
```

The figure produced by this command is shown in Figure 2.4. Once again we

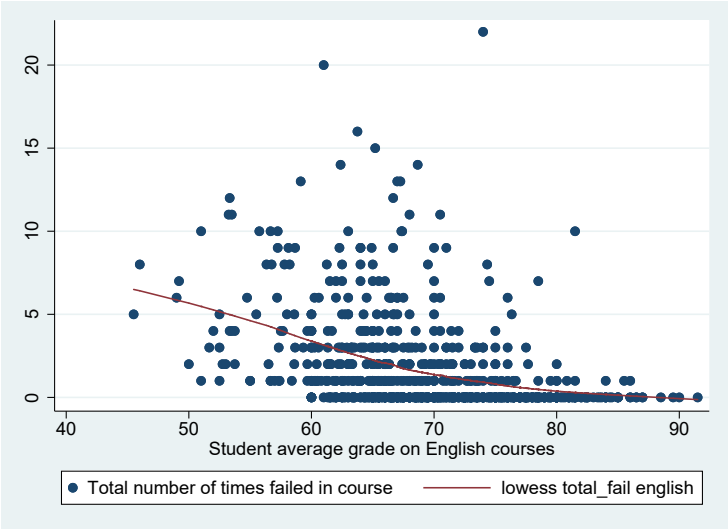

Figure 2.4: Smoothed scatterplot of total_fail and english.

see evidence that as the value of english increases that the number of events decreases. This makes sense because if students don't have a good grasp of the English language then they will have difficulties in passing subjects that are taught in English.

We next include the variable in a Poisson model:

```
. poisson total_fail english, irr exposure(total_courses)

Iteration 0:   log likelihood = -1406.8576
Iteration 1:   log likelihood = -1406.8565
Iteration 2:   log likelihood = -1406.8565

Poisson regression                              Number of obs     =        755
                                                LR chi2(1)        =     801.08
                                                Prob > chi2       =     0.0000
Log likelihood = -1406.8565                     Pseudo R2         =     0.2216

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     english |   .8893731   .0036963   -28.21   0.000      .882158    .8966472
       _cons |   131.3423   35.34142    18.13   0.000     77.51119    222.5589
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------
```

```
Note: _cons estimates baseline incidence rate.
```

We see that when english increases by one, that the expected number of failed courses is multiplied by 0.889.

2.1.2 Binary Variables

Now that we have seen how to analyze the relationship between the binary dependent variable and a continuous independent variable, we move onto other types of variables. Looking at our dataset, we notice that the variables gender and college are binary. Both take on two values. Let us include each of these two variable separately:

```
. poisson total_fail i.college, irr exposure(total_courses)

Iteration 0:   log likelihood = -1813.5521
Iteration 1:   log likelihood = -1813.5521

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =      12.33
                                                Prob > chi2       =     0.0004
Log likelihood = -1813.5521                     Pseudo R2         =     0.0034

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     college |
Engineering |   .8180037   .0465096    -3.53   0.000     .7317423    .9144341
       _cons |   .0571091   .0024783   -65.97   0.000     .0524525     .062179
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------
Note: _cons estimates baseline incidence rate.
```

Notice that when we include a categorical variable that we use the i prefix. It is always good practice to tell Stata that a certain variable is an indicator,

or categorical, variable. Looking at the output, we see that the IRR is 0.818. What this means is that the expected number of course withdrawals for engineers is 0.818 times the expected number of course withdrawals for the reference group, which is business students in this case. We can also see that the result is significant at the p ¡ 0.05 level.

It is possible to tell Stata to use the group "engineering" as the reference group:

```
. poisson total_fail b2.college, irr exposure(total_courses)

Iteration 0:   log likelihood = -1813.5521
Iteration 1:   log likelihood = -1813.5521

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =      12.33
                                                Prob > chi2       =     0.0004
Log likelihood = -1813.5521                     Pseudo R2         =     0.0034

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     college |
    Business |   1.222488   .0695075     3.53   0.000     1.093573    1.366601
       _cons |   .0467154   .0017161   -83.40   0.000     .0434701     .050203
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------

Note: _cons estimates baseline incidence rate.
```

As can be seen from the output, the display shows the incidence-rate ratio of business students in comparison to engineering students. We see that the expected number of course withdrawals for business students is 1.222 times the expected number of course withdrawals for engineering students.

We next include the binary variable gender:

```
. poisson total_fail i.gender, irr exposure(total_courses)
```

```
Iteration 0:   log likelihood = -1712.8214
Iteration 1:   log likelihood = -1712.8145
Iteration 2:   log likelihood = -1712.8145

Poisson regression                              Number of obs     =        760
                                                LR chi2(1)        =     213.80
                                                Prob > chi2       =     0.0000
Log likelihood = -1712.8145                     Pseudo R2         =     0.0587

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      gender |
        male |   2.845421   .2288983    13.00   0.000     2.430369    3.331356
       _cons |   .0224076   .0016702   -50.96   0.000      .019362    .0259322
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------

Note: _cons estimates baseline incidence rate.
```

We see that the expected number of failed courses for males is 2.845 times the expected number of failed courses for females.

2.2 Multivariate Analysis

After looking at each independent variable by itself, we need to start building a more complex model. This means that we need a model that includes more than one independent variable. We start with a model that includes all the variables that were found to be significant when we conducted the univariate analysis:

```
. poisson total_fail gpa english i.college i.gender, irr exposure(total_courses)

Iteration 0:   log likelihood = -1099.3037
Iteration 1:   log likelihood = -1099.2339
Iteration 2:   log likelihood = -1099.2339

Poisson regression                              Number of obs     =        755
```

```
                                                LR chi2(4)      =    1416.32
                                                Prob > chi2     =     0.0000
Log likelihood = -1099.2339                     Pseudo R2       =     0.3918

------------------------------------------------------------------------------
  total_fail |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         gpa |   .8876014   .0041204   -25.68   0.000     .8795622     .895714
     english |   .9663382      .0051    -6.49   0.000      .956394    .9763859
             |
     college |
Engineering  |   1.113613   .0669225     1.79   0.073     .9898769    1.252815
             |
      gender |
        male |   1.413999   .1209597     4.05   0.000     1.195731    1.672109
       _cons |   2418.674   780.7668    24.14   0.000     1284.703    4553.568
ln(total_~s) |          1  (exposure)
------------------------------------------------------------------------------

Note: _cons estimates baseline incidence rate.
```

Looking at the model, we see that all variable retain their significance levels except for the variable college.

2.3 Negative Binomial Regression

Now it is time to see whether the data displays overdispersion. As you recall, when there is evidence that overdispersion exists, we will need to fit a negative binomial model that will estimate the new parameter alpha. To fit the negative binomial model in Stata, we use the following command:

```
. nbreg total_fail gpa english i.college i.gender, irr exposure(total_courses) nolog

Negative binomial regression                    Number of obs     =        755
                                                LR chi2(4)        =     540.91
Dispersion     = mean                           Prob > chi2       =     0.0000
Log likelihood = -987.46614                     Pseudo R2         =     0.2150
```

total_fail	IRR	Std. Err.	z	P>|z|	[95% Conf.	Interval]
gpa	.8562923	.0082241	-16.15	0.000	.8403241	.8725639
english	.9626258	.0078959	-4.64	0.000	.9472738	.9782266
college						
Engineering	1.221273	.1176047	2.08	0.038	1.011218	1.474961
gender						
male	1.327607	.1531766	2.46	0.014	1.058912	1.664484
_cons	39032.16	26223.31	15.74	0.000	10460.5	145644
ln(total_~s)	1	(exposure)				
/lnalpha	-.6676535	.1361952			-.9345912	-.4007157
alpha	.5129107	.069856			.3927464	.6698404

Note: Estimates are transformed only in the first equation.

Note: _cons estimates baseline incidence rate.

LR test of alpha=0: chibar2(01) = 223.54 Prob >= chibar2 = 0.000

Notice that the only that is different is that instead using the command poisson, we used the command nbreg (the nolog option is specified in order to suppress the log-likelihood estimation since the take up a lot of space). Everything else is the same. Of primary importance for us is the last line in the output, the one which reports the result of the likelihood-ratio test with regards to the value of alpha. As you recall, a likelihood-ratio test is used to test the null hypothesis that alpha is equal to zero, which means that there is no overdispersion. Looking at our output, we see that the p-value is less than 0.05, thus leading to the rejection of the null hypothesis. We therefore conclude that there is overdispersion and that the negative binomial regression model should be used instead of the Poisson model.

Zero-Inflated Models

When we plotted the histogram of the dependent variables, we noted that the number of zeros seems to be too high. This means that perhaps a zero-inflated model would be of better use. The zero-inflated Poisson model is fit using the zip command and the zero-inflated negative binomial model is fit using the zinb command. Since we have found that there is overdispersion in the data, it would make sense for us to fit the zero-inflated negative binomial regression model:

```
. zinb total_fail gpa english i.college i.gender, exposure(total_courses) ///
>  inflate(gpa english i.college i.gender) nolog

Zero-inflated negative binomial regression      Number of obs     =        755
                                                Nonzero obs       =        351
                                                Zero obs          =        404

Inflation model = logit                         LR chi2(4)        =     253.69
Log likelihood  = -957.9385                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
  total_fail |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
total_fail   |
         gpa |  -.1057381   .0090785   -11.65   0.000    -.1235317   -.0879445
     english |  -.0304713   .0076049    -4.01   0.000    -.0453766    -.015566
             |
     college |
Engineering  |   .1139091   .0943508     1.21   0.227     -.071015    .2988332
             |
      gender |
        male |   .2515483   .1220756     2.06   0.039     .0122845    .4908122
       _cons |   6.801989     .66255    10.27   0.000     5.503415    8.100564
ln(total_~s) |          1  (exposure)
-------------+----------------------------------------------------------------
inflate      |
         gpa |   .3518453   .0541493     6.50   0.000     .2457145    .4579761
     english |   .0159122   .0350812     0.45   0.650    -.0528457    .0846701
             |
     college |
```

Engineering	-.3981901	.4403973	-0.90	0.366	-1.261353	.4649727
gender						
male	-.0967932	.4380667	-0.22	0.825	-.9553881	.7618018
_cons	-28.72857	3.672004	-7.82	0.000	-35.92556	-21.53157
/lnalpha	-1.320756	.1988137	-6.64	0.000	-1.710424	-.9310886
alpha	.2669333	.05307			.1807891	.3941244

The command needs some explanation. First, not that I have excluded the irr option (the reason will be explained below). Second, we have used the inflate() option. As you recall from the theory part, the zero-inflated model is made up to two parts. One part is binary, in that it predicts whether an individual will be in the zero group or in the non-zero group. The other part is the count part, where the expected number of occurrences is predicted. In the command above, the inflate() option is used to tell Stata which variables to use as independent variables in the binary part of the model. In other words, we want to see which variables might be causing the inflated number of zeros. We have included all variables because at the point we do not know which variables will be significant and which will not be significant.

Notice that the output is divided into two parts. The top parts represents the count model regression, while the bottom part, titled "inflate", represents the binary regression where we are trying to predict group membership (the zero-group vs the non-zero group). The following is very important. The bottom part is trying to see which variables are causing the inflated number of zeros. What this means is that if a variable has a positive value for the coefficient, then increasing values of this variable will increase the probability that the individual will end up in the zero-group. If you look at the variable

gpa in the "inflate" section, you can see that the coefficient is 0.352, which is positive. This basically means that higher values of GPA will lead to an increase in the probability that the student will remain in the zero-group. This makes sense. Students with higher GPAs are more likely to fail in zero courses. We also see that the coefficient of english is positive. The same logic applies here. Higher grades on the English courses will increase the probability that the student will remain in the zero group.

Looking at the variable college, we see that the coefficient is negative. As you know, this is a binary variable, and the reference group here is business. What this coefficient means is that engineering students are less likely than business students to remain in the zero group. How much less likely? In count models, when we find $e^{coefficient}$ we are finding the incidence-rate ratio. In logistic regression, when the dependent variable is binary, calculating $e^{coefficient}$ gives us the odds ratio, which is the ratio of the odds that the event happens in one group compared to the reference group. In our case $e^{-0.398} = 0.672$. What this means is that the odds of an engineering student being in the zero group are 0.672 times the odds that a business student being in the zero group.

At this point you might be asking yourself why didn't we just use the irr option in order to calculate $e^{coefficient}$ instead of us doing it by hand. The reason is that including this option in the zinb model will only result in the exponentiated coefficients of the part on top, which is the count model. The "inflate" part will continue to display the coefficients (go ahead and try including the irr option). Stata allows us to resolve this issue by executing the listcoef command directly after the zinb command:

```
. listcoef

zinb (N=755): Factor change in expected count
```

```
  Observed SD:  2.8663

Count equation: Factor change in expected count for those not always 0
```

	b	z	P>|z|	e^b	e^bStdX	SDofX
gpa	-0.1057	-11.647	0.000	0.900	0.450	7.555
english	-0.0305	-4.007	0.000	0.970	0.800	7.338
college						
Engineering	0.1139	1.207	0.227	1.121	1.058	0.494
gender						
male	0.2515	2.061	0.039	1.286	1.125	0.467
constant	6.8020	10.266	0.000	.	.	.
alpha						
lnalpha	-1.3208	.	.	.	.	.
alpha	0.2669	.	.	.	.	.

```
Binary equation: factor change in odds of always 0
```

	b	z	P>|z|	e^b	e^bStdX	SDofX
gpa	0.3518	6.498	0.000	1.422	14.269	7.555
english	0.0159	0.454	0.650	1.016	1.124	7.338
college						
Engineering	-0.3982	-0.904	0.366	0.672	0.821	0.494
gender						
male	-0.0968	-0.221	0.825	0.908	0.956	0.467
constant	-28.7286	-7.824	0.000	.	.	.

We see that the output continues to show the coefficients in the first column, in addition to the values $e^{coefficient}$ in the column title e^b. If you look at the variable college in the bottom part, you will see that the coefficient is -0.3982 and that the odds ratio is 0.672, just like we calculated.

We are interested in the p-values of the inflate part since we originally included all the independent variables in order to see which ones might be inflating the number of zeros. We had previously seen that that in the "inflate" part, only the variable gpa is significant. This would indicate that we might be better off fitting a model that only included it in the inflate() option:

```
. zinb total_fail gpa english i.college i.gender, exposure(total_courses) ///
> inflate(gpa) nolog

Zero-inflated negative binomial regression      Number of obs     =        755
                                                Nonzero obs       =        351
                                                Zero obs          =        404

Inflation model = logit                         LR chi2(4)        =     259.07
Log likelihood  = -958.4367                     Prob > chi2       =     0.0000
```

total_fail	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
total_fail						
gpa	-.1059945	.0090624	-11.70	0.000	-.1237565	-.0882324
english	-.0316688	.0073566	-4.30	0.000	-.0460876	-.0172501
college						
Engineering	.1447259	.0880362	1.64	0.100	-.0278219	.3172737
gender						
male	.2656364	.1119332	2.37	0.018	.0462513	.4850214
_cons	6.86581	.6550236	10.48	0.000	5.581988	8.149633
ln(total_~s)	1	(exposure)				
inflate						
gpa	.360155	.0459039	7.85	0.000	.270185	.4501251
_cons	-28.61654	3.626404	-7.89	0.000	-35.72416	-21.50892
/lnalpha	-1.312343	.1968799	-6.67	0.000	-1.69822	-.9264651
alpha	.2691887	.0529978			.183009	.3959509

The output shows that gpa is significant both as an inflation variable and as a count variable. The output also shows that the variable college is no longer significant.

2.4 Comparing Count Models

So which is it? Is it the negative binomial model or the zero-inflated binomial model? Fortunately, this is where the package SPost helps save the day. This package contains several commands which are very useful, specifically the command countfit, which automates the process of comparing the four models: Poisson, negative binomial, zero-inflated Poisson, and zero-inflated negative binomial. This is how we execute this command:

```
countfit total_fail gpa english i.college i.gender, inflate(gpa)
```

Note that the syntax of the command is similar to the syntax of the count model commands that we have been using so far. After specifying the name of the command, we list the dependent variable followed by the independent variables. We also specify the option inflate() in which we list the variables that we suspect are causing the high number of zero counts. Note that the command countfit does not include the exposure() option, so we cannot include it.

The output of the command is very long, but extremely useful. Let's go through the output step by step.

Variable	PRM	NBRM	ZIP	ZINB

total_fail				
Student overall GPA	0.928	0.859	0.966	0.941
	-17.66	-11.68	-6.74	-4.75
Student average grade on Eng-h	0.967	0.971	0.985	0.975
	-6.92	-2.84	-2.95	-2.65
College				
Engineering	1.218	1.450	1.177	1.291
	3.28	3.04	2.53	2.25
Gender				
male	1.553	1.426	1.519	1.459
	5.11	2.56	4.54	2.77
Constant	2409.199	5.01e+05	66.849	689.760
	25.64	14.77	11.20	7.17
lnalpha				
Constant		1.222		0.731
		2.05		-2.26
inflate				
Student overall GPA			1.318	1.485
			11.59	7.91
Constant			0.000	0.000
			-11.61	-7.83
Statistics				
alpha		1.222		
N	755	755	755	755
ll	-1427.307	-1098.999	-1195.208	-1064.134
bic	2887.747	2237.759	2436.802	2181.282
aic	2864.614	2209.999	2404.415	2144.268

legend: b/t

In the first part, the command is displaying the exponentaited parameters for the four models. We note that the values are very similar across the models. Most importantly, the direction of the variables is the same. By direction I mean that the models agree whether a variable increases or decreases the counts. Since the values that are displayed are exponentiated,

these are incidence-rate ratios. This means that values that are greater than one indicate that an increase in the variable will lead to an increase in the count, and a value that is less than one indicates that an increase in the variable will lead to a decrease in the count. We also see that the output includes an "inflate" section for the zero-inflated models. The output also contains the AIC and BIC goodness of fit statistics at the bottom. Models that produce lower values of these statistics are considered a better fit. We see that the zero-inflated negative binomial model produces the lowest AIC and BIC statistics.

```
Comparison of Mean Observed and Predicted Count

             Maximum       At      Mean
Model     Difference    Value    |Diff|
-----------------------------------------------
PRM          0.204         0      0.049
NBRM        -0.027         1      0.008
ZIP          0.079         1      0.020
ZINB         0.021         1      0.007
```

The second part of the output, displays a comparison of the differences between the observed counts and the predicted counts of each model followed by separate tables of predicted counts for each model. We are mainly interested in the table of differences. Ideally, the best model is the one that most accurately predicts the correct values. This means that the model that produces the smallest average difference between the actual and the predicted counts is the best. Once again, we see that the zero-inflated negative model produces the smallest average difference, thus lending further support to the idea that it is the best model.

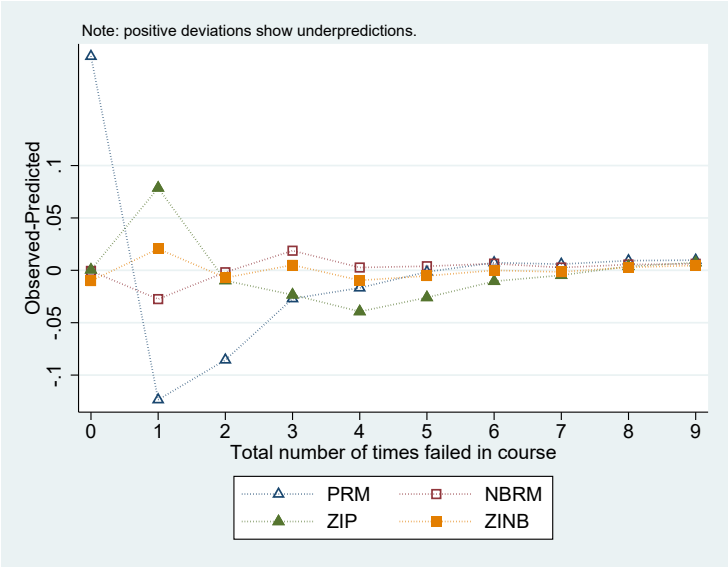

Figure 2.5: Difference between observed and predicted values in each of the four models, as plotted by countfit.

The command also produces a plot for us which helps us visualise the differences. We see in Figure 2.5 that the Poisson regression model (PRM) produces the largest differences between observed and predicted values, followed by the zero-inflated Poisson model (ZIP). The smallest deviations are produced by the negative binomial regression model (NBRM) and the zero-inflated negative binomial (ZINB) model, with the ZINB doing a slightly better job.

```
Tests and Fit Statistics

PRM             BIC=  2887.747  AIC=  2864.614  Prefer  Over  Evidence
------------------------------------------------------------------------
  vs NBRM       BIC=  2237.759  dif=   649.988  NBRM    PRM   Very strong
                AIC=  2209.999  dif=   654.615  NBRM    PRM
                LRX2=  656.615  prob=    0.000  NBRM    PRM   p=0.000
------------------------------------------------------------------------
  vs ZIP        BIC=  2436.802  dif=   450.945  ZIP     PRM   Very strong
                AIC=  2404.415  dif=   460.199  ZIP     PRM
```

```
              Vuong=       .  prob=        .  ZIP     PRM   p=.
-------------------------------------------------------------------------------
  vs ZINB     BIC=  2181.282  dif=   706.465  ZINB    PRM   Very strong
              AIC=  2144.268  dif=   720.345  ZINB    PRM
-------------------------------------------------------------------------------
NBRM          BIC=  2237.759  AIC=  2209.999  Prefer  Over  Evidence
-------------------------------------------------------------------------------
  vs ZIP      BIC=  2436.802  dif=  -199.043  NBRM    ZIP   Very strong
              AIC=  2404.415  dif=  -194.416  NBRM    ZIP
-------------------------------------------------------------------------------
  vs ZINB     BIC=  2181.282  dif=    56.477  ZINB    NBRM  Very strong
              AIC=  2144.268  dif=    65.731  ZINB    NBRM
              Vuong=       .  prob=        .  ZINB    NBRM  p=.
-------------------------------------------------------------------------------
ZIP           BIC=  2436.802  AIC=  2404.415  Prefer  Over  Evidence
-------------------------------------------------------------------------------
  vs ZINB     BIC=  2181.282  dif=   255.520  ZINB    ZIP   Very strong
              AIC=  2144.268  dif=   260.147  ZINB    ZIP
              LRX2=  262.147  prob=    0.000  ZINB    ZIP   p=0.000
-------------------------------------------------------------------------------

Vuong test is not appropriate for testing zero-inflation. To force the
the computation of the test, use option -forcevuong-.
```

Finally, and most importantly, the last part of the output shows a table that directly compares each model with the three other models. Starting at the top, the table compares the PRM with the NBRM using the BIC and AIC statistics, and using the likelihood-ratio test that is used to test the null hypothesis that alpha is zero. We see that all three tests favor the NBRM.

The table next displays a comparison of the PRM and the ZIP. We see that the three tests favor the ZIP. We then see a comparison of the PRM and the ZINB with both the AIC and the BIC statistics favoring the ZINB. The conclusion to the first part is that the ZIP is the worst out of all the models.

The second part starts by comparing the NBRM with the ZIP. We see that both the AIC and the BIC statistics favor the NBRM. A comparison of the

NBRM and the ZINB leads to the conclusion that the ZINB model is better. The conclusion of this is that the NBRM model is better than the ZIP and that the ZINB model is better than the NBRM.

We finally come to the last entry that directly compares the ZIP with the ZINB model. We see that all three tests favor the ZINB model. Therefore, the entire output of the countfit command is telling us the same thing: go with the zero-inflated negative binomial model.

2.5 Model Interpretation and Prediction

2.5.1 Predicted Number of Events

Now that we have seen that the model fit is good, it is time to interpret the obtained model parameters. Stata allows us to do this using both graphical tools and non-graphical tools. It is important to note that in count regression models, the dependent variable is the expected number of outcomes. In the dataset that we have been working on, this is the expected number of failed courses. Therefore, once the model is fit, we should be able to predict the expected number of failed courses for each student. Note that in the previous section, we found that best-fit model was the zero-inflated negative model, which is obtained by running the following command:

```
zinb total_fail gpa english i.college i.gender, exposure(total_courses) inflate(gpa)
```

We have already covered how to interpret the output produced by this command. Since the irr option is not specified, the output includes the coefficients. To make sense of these coefficients we need to exponentiate them.

Once we do that, we can find the factor by which the expected number of outcomes is multiplied when the independent variable increases by one unit.

The SPost package contains a very useful command that will also help us in making sense of the output. This command is mchange. In order to see it in action, after fitting the model, execute the following:

```
. mchange

zinb: Changes in mu | Number of obs = 755

Expression: Predicted number of total_fail, predict()
```

	Change	p-value
gpa		
+1	-0.215	0.000
+SD	-1.170	0.000
Marginal	-0.226	0.000
english		
+1	-0.050	0.000
+SD	-0.331	0.000
Marginal	-0.051	0.000
college		
Engineering vs Business	0.229	0.100
gender		
male vs female	0.388	0.010

```
Average prediction

    1.595
```

Let us go through the output for each variable starting with gpa. We see that, "on average, when there is a +1 increase in gpa the expected number of outcomes will change by -0.215". We also see that the result is significant. Moving on to the variable english, we see that, "on average, a +1 increase in english decreases the expected number of failed courses by 0.05".

Moving on to the binary variables, we see that "on average, being an engi-

neering student increases the expected number of failed courses by 0.229", and "on average, being a male increases the expected number of failed courses by 0.388". Note, however, that the result for the variable college is not significant.

As you can see, the command mchange allows for a more intuitive explanation. Previously, the we used the incidence-rate ratio to calculate the factor by which the expected number of events is multiplied. Using the mchange command, we can get the exact values of the increase or the decrease.

We can also use the mchange command in order to look at the effect of changes that are different than +1. For example, we can tell Stata to display the change in the expected number of failed courses when gpa increases by ten:

```
. mchange gpa, delta(10)

zinb: Changes in mu | Number of obs = 755

Expression: Predicted number of total_fail, predict()

              |    Change    p-value
--------------+-------------------------
gpa           |
          +1  |    -0.215      0.000
      +delta  |    -1.350      0.000
    Marginal  |    -0.226      0.000

Average prediction

    1.595

1: Delta equals 10.
```

Notice here that we speficied the variable gpa as part of the command. This is done in order to tell Stata to display the output for this variable only, as opposed to displaying the output for all variables. We also used the delta(10)

option in order to tell Stata that we want the change in the dependent variable when gpa changes by ten.

Looking at the output, we see that when gpa changes by delta, which is ten in this case, the expected number of failed courses decreases by 1.35. We can also ask Stata to give us more detail by telling it to show us the changes in the dependent variable:

```
. mchange gpa, delta(10) stat(change from to pvalue)

zinb: Changes in mu | Number of obs = 755

Expression: Predicted number of total_fail, predict()
```

	Change	From	To	p-value
gpa				
+1	-0.215	1.595	1.379	0.000
+delta	-1.350	1.595	0.245	0.000
Marginal	-0.226	.z	.z	0.000

```
Average prediction

    1.595

1: Delta equals 10.
```

We specified the stat() option in order to tell Stata that we want to see the change in the expected number of events, the value from which the expected number changed, the value to which the expected number changes when GPA increases by ten, and the p-value of the result. We see that, on average, when gpa increases by ten, the expected number of failed courses decreases from 1.595 to 0.245.

We can also use the mchange command with categorical variables:

```
. mchange gender, stat(change from to pvalue)

zinb: Changes in mu | Number of obs = 755
```

```
Expression: Predicted number of total_fail, predict()

                |      Change       From         To    p-value
----------------+-----------------------------------------------
gender          |
 male vs female |       0.388      1.275      1.663      0.010

Average prediction

    1.595
```

Notice that we do not use the delta() option with binary variables because these variables can only change by one unit. Looking at the output we see that the expected number increases from 1.275 to 1.663 when moving from females to males.

2.5.2 Calculating the Probabilities of Different Outcomes

As mentioned previously, when we fit count regression models, we are also interested in finding the probability that a certain individual will display certain levels of the count variable. So far, we have been estimating the predicted number of outcomes. What if we want to calculate the probability that the count variable will take on the values zero, one, two, and three? In other words, what if we wanted to find the probabilities for each level of the dependent variable? Fortunately, the package SPost contains a command that makes life easier for us, and this command is mtable. Once again, when the regression model is fit, we can use this command in order to find the probabilities that a student with fail in one course, two courses, three courses, and so on:

```
. mtable, pr(0/5)

Expression: Pr(total_fail), predict(pr())

       0         1         2         3         4         5
------------------------------------------------------------
   0.526     0.143     0.104     0.071     0.048     0.032

Specified values where .n indicates no values specified with at()

           | No at()
-----------+---------
   Current |      .n
```

We use the pr(0/5) option in order to tell Stata that we want to predict the probability that a student will fail in one, two, three, four, and five courses. We see that the probability that a student will have failed from zero courses is the highest (0.526). We also see that there is a probability of 0.071 that a student would have failed from three courses.

In order to really take advantage of the command mtable, we need to compare different individuals. This can be done by using the at() option:

```
. mtable, at(gender=(1 2)) pr(0/5)

Expression: Pr(total_fail), predict(pr())

           |  gender         0         1         2         3         4         5
-----------+-------------------------------------------------------------------
         1 |       1     0.554     0.157     0.106     0.067     0.041     0.026
         2 |       2     0.515     0.140     0.106     0.074     0.051     0.034

Specified values where .n indicates no values specified with at()

           | No at()
-----------+---------
   Current |      .n
```

Here, we specified that we wanted to calculate the probabilities for two groups, those who have a value of one for gender (females) and those who

have a value of two for gender (males). We see that we now have the predicted probabilities for each outcome level for both genders. Females, for example, have a higher probability of having failed in zero courses than males, but both genders have equal probabilities to have failed in two courses.

We can also use the command mtable to produce probabilities for specific values of more than one variable. For example, what if we wanted to calculate the probabilities for female students for different values of gpa? This can be easily done using the command:

```
. mtable, at(gender=(1) gpa=(70(2)80)) pr(0/5)

Expression: Pr(total_fail), predict(pr())
```

	gpa	0	1	2	3	4	5
1	70	0.244	0.225	0.178	0.125	0.084	0.054
2	72	0.317	0.242	0.172	0.110	0.066	0.039
3	74	0.410	0.244	0.155	0.089	0.048	0.025
4	76	0.525	0.225	0.127	0.064	0.031	0.015
5	78	0.655	0.184	0.091	0.040	0.017	0.007
6	80	0.780	0.130	0.056	0.022	0.008	0.003

```
Specified values of covariates
```

	gender
Current	1

This command is telling Stata to calculate the probabilities only for females (since gender takes on only one value, which is one). The command is also telling Stata to calculate the probabilities for different values of gpa. In this case, we are telling Stata that the variable gpa should start at 70 and vary in increments of two until it reached 80. This basically means that Stata should display the probabilities when gpa is 70, 72, 74, 76, 78, and 80. We see that Stata is telling us that the variable gender has been specified to be equal to

one (at the bottom of the output). We also see that Stata has calculated the probabilities for the levels of gpa which we had specified. We see, for example, that a female with a GPA of 74 has a 0.244 probability of failing in one course. As you can see, this command is very powerful because it allows us to calculate the probability for any level of the outcome for specific values of the independent variables.

2.6 Visualizing the Results

As we saw in the previous section, interpreting the results in terms of the predicted number of outcomes and the calculated probabilities is very intuitive and clear. In my opinion, the best way to understand models is to summarize them using meaningful graphs. As an example, take the case of the independent GPA. We would like to know how the expedted number of failed courses changes with respect to changes in the value of student GPA. Stata offers us two very powerful tools that make out life easier, and these are the command margins and marginsplot. These commands tend to be used together where the job of margins is to calculate the values and the job of marginsplot is to plot the calculated values. As with other prediction commands, both of these commands need to be executed after fitting the model.

Let us start by plotting the relationship between the dependent variable and GPA:

```
. margins, at(gpa=(60(1)100)) noatlegend

Predictive margins                              Number of obs     =        755
Model VCE    : OIM
```

Expression : Predicted number of events, predict()

	Delta-method					
	Margin	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	8.17398	.940802	8.69	0.000	6.330042	10.01792
2	7.349038	.7871504	9.34	0.000	5.806251	8.891824
3	6.60623	.6559777	10.07	0.000	5.320537	7.891923
4	5.937059	.5444319	10.91	0.000	4.869992	7.004126
5	5.333816	.4500319	11.85	0.000	4.451769	6.215862
6	4.789483	.3706268	12.92	0.000	4.063068	5.515898
7	4.297646	.3043592	14.12	0.000	3.701113	4.894179
8	3.852409	.2496301	15.43	0.000	3.363143	4.341675
9	3.448318	.2050607	16.82	0.000	3.046407	3.85023
10	3.080298	.1694419	18.18	0.000	2.748198	3.412398
11	2.7436	.1416618	19.37	0.000	2.465948	3.021252
12	2.433785	.120607	20.18	0.000	2.197399	2.67017
13	2.146753	.1050601	20.43	0.000	1.940839	2.352667
14	1.878849	.0936582	20.06	0.000	1.695283	2.062416
15	1.627071	.0849808	19.15	0.000	1.460512	1.793631
16	1.389392	.0777727	17.86	0.000	1.236961	1.541824
17	1.165151	.0712141	16.36	0.000	1.025573	1.304728
18	.9553532	.0650899	14.68	0.000	.8277793	1.082927
19	.7626494	.059647	12.79	0.000	.6457434	.8795553
20	.5907245	.0550489	10.73	0.000	.4828306	.6986185
21	.4431288	.0508594	8.71	0.000	.3434462	.5428113
22	.3219574	.0462402	6.96	0.000	.2313283	.4125866
23	.2270247	.0406621	5.58	0.000	.1473284	.3067211
24	.1559117	.034267	4.55	0.000	.0887495	.2230739
25	.1047316	.0276482	3.79	0.000	.0505422	.1589211
26	.069116	.0214449	3.22	0.001	.0270848	.1111473
27	.0449911	.0160871	2.80	0.005	.0134611	.0765212
28	.0289866	.0117441	2.47	0.014	.0059685	.0520046
29	.018534	.0083898	2.21	0.027	.0020902	.0349778
30	.0117855	.0058919	2.00	0.045	.0002376	.0233334
31	.0074647	.0040822	1.83	0.067	-.0005362	.0154656
32	.0047147	.0027981	1.68	0.092	-.0007695	.010199
33	.0029719	.0019016	1.56	0.118	-.0007552	.006699
34	.0018707	.0012834	1.46	0.145	-.0006446	.0043861
35	.0011764	.0008612	1.37	0.172	-.0005115	.0028643
36	.0007393	.0005751	1.29	0.199	-.0003879	.0018665

```
------------------------------------------------------------------------------
          37 |   .0004644   .0003825     1.21   0.225    -.0002854    .0012141
          38 |   .0002916   .0002536     1.15   0.250    -.0002054    .0007885
          39 |    .000183   .0001676     1.09   0.275    -.0001454    .0005114
          40 |   .0001149   .0001104     1.04   0.298    -.0001016    .0003313
          41 |   .0000721   .0000726     0.99   0.321    -.0000702    .0002144
------------------------------------------------------------------------------
```

This command tells Stata to calculate the predicted number of outcomes when GPA is varied from 60 to 100 in increments of one. The output of the command is very long, so I used the noatlegend option in order to suppress some parts of it. However, if you take a look at the output, you will notice that Stata is simply calculating the predicted number of events when GPA is 60, 61, 62, and so on.

After calculating these values, we can tell Stata to plot them:

```
. marginsplot, noci

  Variables that uniquely identify margins: gpa
```

We specify the noci option in order to tell Stata not to plot the confidence intervals. The resulting graph is shown in Figure 2.6. We see that there is a drop in the expected number of failed courses as GPA increases, and that this drop tends to level off when the GPA is above 80 since students with a GPA that is higher than 80 are expected to fail in zero courses. There is no difference between a student who has a GPA of 87 and one who has a GPA of 94.

We can go a step further and plot group differences:

```
. margins, at(gpa=(60(1)100) college=(1 2)) noatlegend

Predictive margins                              Number of obs     =        755
Model VCE    : OIM
```

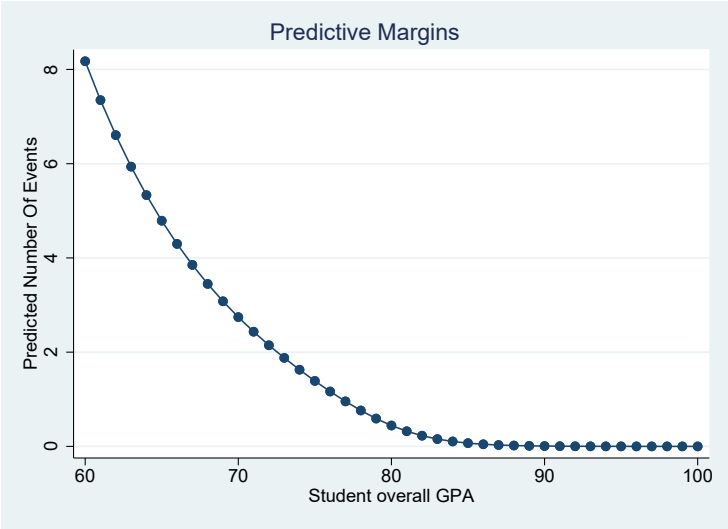

Figure 2.6: Using marginsplot to visualize the relationship between the expected number of events and GPA.

```
Expression   : Predicted number of events, predict()
```

	Margin	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
_at						
1	7.456061	.891972	8.36	0.000	5.707828	9.204294
2	8.617139	1.066015	8.08	0.000	6.527787	10.70649
3	6.703573	.7554264	8.87	0.000	5.222965	8.184182
4	7.747472	.8983695	8.62	0.000	5.9867	9.508244
5	6.026006	.6391183	9.43	0.000	4.773357	7.278655
6	6.964392	.7549404	9.23	0.000	5.484736	8.444048
7	5.415608	.5404651	10.02	0.000	4.356316	6.4749
8	6.258941	.632655	9.89	0.000	5.01896	7.498922
9	4.865348	.4571987	10.64	0.000	3.969255	5.761441
10	5.622993	.5288295	10.63	0.000	4.586506	6.65948
11	4.368823	.3873197	11.28	0.000	3.609691	5.127956
12	5.049148	.4411228	11.45	0.000	4.184564	5.913733
13	3.920184	.3290535	11.91	0.000	3.275251	4.565117
14	4.530646	.3674943	12.33	0.000	3.810371	5.250922
15	3.514053	.2808083	12.51	0.000	2.963678	4.064427

16	4.06127	.3061637	13.27	0.000	3.4612	4.66134
17	3.145453	.2411346	13.04	0.000	2.672838	3.618068
18	3.635272	.25557	14.22	0.000	3.134364	4.13618
19	2.809756	.2086915	13.46	0.000	2.400728	3.218784
20	3.247299	.2143262	15.15	0.000	2.827227	3.667371
21	2.50263	.1822204	13.73	0.000	2.145485	2.859775
22	2.892346	.1811664	15.97	0.000	2.537267	3.247426
23	2.220026	.1605329	13.83	0.000	1.905387	2.534665
24	2.565734	.154883	16.57	0.000	2.262169	2.8693
25	1.958204	.1425115	13.74	0.000	1.678887	2.237521
26	2.263141	.1342646	16.86	0.000	1.999987	2.526295
27	1.71383	.1271265	13.48	0.000	1.464667	1.962994
28	1.980713	.1180584	16.78	0.000	1.749322	2.212103
29	1.484166	.1134754	13.08	0.000	1.261758	1.706573
30	1.715284	.1050012	16.34	0.000	1.509486	1.921083
31	1.267362	.1008597	12.57	0.000	1.069681	1.465044
32	1.464719	.0939522	15.59	0.000	1.280577	1.648862
33	1.062816	.088907	11.95	0.000	.888561	1.23707
34	1.22832	.0841147	14.60	0.000	1.063458	1.393182
35	.8714447	.0776809	11.22	0.000	.7191929	1.023696
36	1.007148	.0752281	13.39	0.000	.8597041	1.154593
37	.695666	.067613	10.29	0.000	.5631469	.8281851
38	.803997	.0674901	11.91	0.000	.6717189	.9362751
39	.5388413	.0590985	9.12	0.000	.4230103	.6546723
40	.6227511	.061052	10.20	0.000	.5030915	.7424108
41	.4042088	.0519734	7.78	0.000	.3023428	.5060749
42	.4671533	.0554649	8.42	0.000	.3584442	.5758625
43	.29368	.0455083	6.45	0.000	.2044853	.3828746
44	.3394126	.0498162	6.81	0.000	.2417746	.4370506
45	.2070852	.0390222	5.31	0.000	.1306031	.2835672
46	.239333	.0434606	5.51	0.000	.1541518	.3245143
47	.142218	.032368	4.39	0.000	.0787779	.205658
48	.1643646	.0364459	4.51	0.000	.0929318	.2357973
49	.0955331	.0258601	3.69	0.000	.0448482	.146218
50	.1104097	.0293174	3.77	0.000	.0529487	.1678708
51	.0630456	.0199333	3.16	0.002	.023977	.1021142
52	.0728632	.0226964	3.21	0.001	.0283791	.1173473
53	.0410396	.0148921	2.76	0.006	.0118516	.0702275
54	.0474303	.0170048	2.79	0.005	.0141015	.0807592
55	.0264407	.0108415	2.44	0.015	.0051917	.0476896
56	.0305581	.0124039	2.46	0.014	.006247	.0548692
57	.0169061	.0077297	2.19	0.029	.0017562	.0320561
58	.0195388	.0088561	2.21	0.027	.0021812	.0368964

59	.0107504	.0054204	1.98	0.047	.0001266	.0213742
60	.0124245	.0062168	2.00	0.046	.0002398	.0246091
61	.0068091	.0037513	1.82	0.070	-.0005434	.0141616
62	.0078694	.004306	1.83	0.068	-.0005701	.016309
63	.0043007	.0025692	1.67	0.094	-.0007348	.0093361
64	.0049704	.0029509	1.68	0.092	-.0008133	.010754
65	.0027109	.0017448	1.55	0.120	-.0007088	.0061307
66	.0031331	.0020051	1.56	0.118	-.0007969	.007063
67	.0017064	.0011769	1.45	0.147	-.0006002	.0040131
68	.0019722	.001353	1.46	0.145	-.0006797	.0046241
69	.0010731	.0007893	1.36	0.174	-.000474	.0026202
70	.0012402	.0009078	1.37	0.172	-.0005391	.0030195
71	.0006744	.0005269	1.28	0.201	-.0003584	.0017071
72	.0007794	.0006062	1.29	0.199	-.0004088	.0019676
73	.0004236	.0003504	1.21	0.227	-.0002631	.0011103
74	.0004895	.0004032	1.21	0.225	-.0003007	.0012798
75	.000266	.0002322	1.15	0.252	-.0001891	.000721
76	.0003074	.0002672	1.15	0.250	-.0002164	.0008312
77	.000167	.0001534	1.09	0.276	-.0001337	.0004676
78	.000193	.0001766	1.09	0.275	-.0001532	.0005391
79	.0001048	.0001011	1.04	0.300	-.0000933	.0003029
80	.0001211	.0001164	1.04	0.298	-.000107	.0003492
81	.0000658	.0000664	0.99	0.322	-.0000644	.000196
82	.000076	.0000765	0.99	0.321	-.000074	.000226

Here, we are telling Stata to calculate the predicted number of events when both gpa and college vary. We then plot the result:

```
. marginsplot, noci

  Variables that uniquely identify margins: gpa college
```

The result is shown in Figure 2.7. We see that the differences between engineering students and business students is actually small (as you recall, the result is not significant), and that these differences vanish for students with high GPAs.

Previously, we used the mtable command in order to calculate the probability

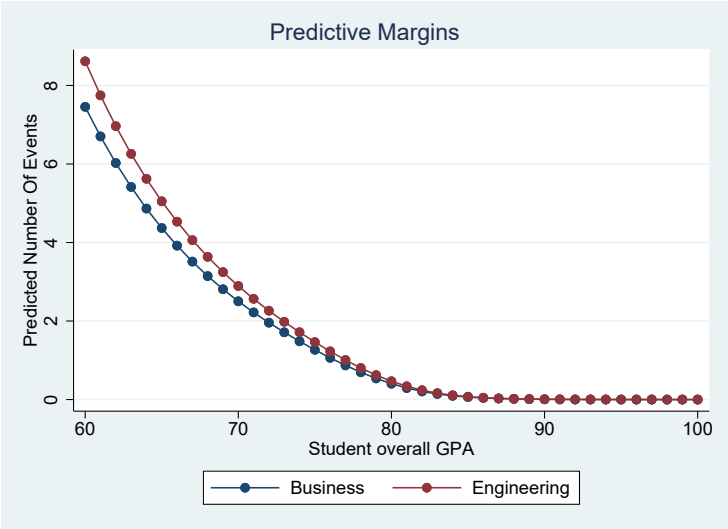

Figure 2.7: Visualizing the effect that two variables have on the expected number of outcomes.

that the event will occur a certain number of times. Not only can we visualize the expected number of events, but we can also use the margins command in order to visualize the probability that the event will occur a certain number of times. For example, assume that we want to calculate the probability that a student will fail in four courses for different values of the variable gpa and for each of the genders. This can be accomplished using the following command:

```
. margins, at(gpa=(60(1)100) gender=(1 2)) predict(pr(4)) noatlegend

Predictive margins                              Number of obs     =        755
Model VCE    : OIM

Expression   : Pr(total_fail=4), predict(pr(4))
```

	Delta-method				
	Margin	Std. Err.	z	P>|z|	[95% Conf. Interval]
_at					
1	.0933833	.0085689	10.90	0.000	.0765885 .1101781

2	.0778393	.0075439	10.32	0.000	.0630536	.092625
3	.0984405	.0072088	13.66	0.000	.0843116	.1125694
4	.0843417	.0067339	12.52	0.000	.0711435	.09754
5	.102504	.0058716	17.46	0.000	.0909959	.1140122
6	.0904192	.0058324	15.50	0.000	.0789878	.1018506
7	.1053562	.0048023	21.94	0.000	.0959439	.1147686
8	.0958388	.0049439	19.39	0.000	.086149	.1055286
9	.1068132	.0043247	24.70	0.000	.098337	.1152894
10	.1003579	.0042242	23.76	0.000	.0920786	.1086372
11	.1067415	.0045854	23.28	0.000	.0977544	.1157287
12	.1037392	.0038505	26.94	0.000	.0961923	.1112861
13	.1050715	.0053328	19.70	0.000	.0946195	.1155236
14	.1057673	.003897	27.14	0.000	.0981294	.1134053
15	.1018059	.0062107	16.39	0.000	.0896331	.1139786
16	.1062665	.0042398	25.06	0.000	.0979566	.1145763
17	.0970219	.0069914	13.88	0.000	.083319	.1107248
18	.1051156	.0046693	22.51	0.000	.0959639	.1142674
19	.0908672	.0075626	12.02	0.000	.0760448	.1056896
20	.1022609	.0050268	20.34	0.000	.0924085	.1121134
21	.0835502	.0078796	10.60	0.000	.0681065	.0989939
22	.0977231	.0052326	18.68	0.000	.0874675	.1079788
23	.0753259	.0079366	9.49	0.000	.0597704	.0908815
24	.0916004	.0052686	17.39	0.000	.0812741	.1019268
25	.0664814	.0077499	8.58	0.000	.051292	.0816709
26	.0840674	.0051577	16.30	0.000	.0739584	.0941764
27	.057321	.0073467	7.80	0.000	.0429217	.0717203
28	.0753712	.0049419	15.25	0.000	.0656852	.0850572
29	.0481544	.0067602	7.12	0.000	.0349048	.0614041
30	.0658273	.0046618	14.12	0.000	.0566904	.0749642
31	.0392871	.0060279	6.52	0.000	.0274726	.0511017
32	.0558136	.004341	12.86	0.000	.0473055	.0643218
33	.0310099	.0051948	5.97	0.000	.0208282	.0411916
34	.0457599	.0039848	11.48	0.000	.0379499	.05357
35	.0235832	.0043151	5.47	0.000	.0151258	.0320407
36	.0361241	.0035914	10.06	0.000	.0290851	.0431632
37	.0172134	.0034493	4.99	0.000	.0104529	.0239739
38	.0273481	.0031645	8.64	0.000	.0211458	.0335504
39	.0120222	.002654	4.53	0.000	.0068205	.0172239
40	.0197932	.0027149	7.29	0.000	.014472	.0251143
41	.0080229	.0019687	4.08	0.000	.0041644	.0118814
42	.0136738	.0022556	6.06	0.000	.009253	.0180947
43	.0051185	.0014098	3.63	0.000	.0023555	.0078816
44	.009021	.0018017	5.01	0.000	.0054898	.0125522

45	.0031299	.0009747	3.21	0.001	.0012195	.0050403
46	.0056973	.0013742	4.15	0.000	.0030038	.0083908
47	.0018417	.0006503	2.83	0.005	.0005673	.0031162
48	.0034583	.0009975	3.47	0.001	.0015032	.0054133
49	.001048	.0004186	2.50	0.012	.0002275	.0018684
50	.0020272	.0006893	2.94	0.003	.0006761	.0033782
51	.0005794	.0002604	2.22	0.026	.000069	.0010899
52	.0011532	.0004552	2.53	0.011	.0002609	.0020454
53	.0003128	.000157	1.99	0.046	5.04e-06	.0006205
54	.0006395	.0002888	2.21	0.027	.0000736	.0012055
55	.0001655	.000092	1.80	0.072	-.0000149	.0003459
56	.0003471	.0001769	1.96	0.050	4.22e-07	.0006938
57	.0000861	.0000526	1.63	0.102	-.0000171	.0001893
58	.000185	.0001052	1.76	0.079	-.0000211	.0003911
59	.0000441	.0000295	1.50	0.134	-.0000136	.0001019
60	.0000971	.0000609	1.59	0.111	-.0000224	.0002165
61	.0000224	.0000162	1.38	0.168	-9.40e-06	.0000541
62	.0000502	.0000346	1.45	0.146	-.0000175	.000118
63	.0000112	8.76e-06	1.28	0.201	-5.97e-06	.0000284
64	.0000257	.0000192	1.34	0.181	-.000012	.0000634
65	5.57e-06	4.67e-06	1.19	0.233	-3.59e-06	.0000147
66	.000013	.0000105	1.24	0.216	-7.60e-06	.0000336
67	2.75e-06	2.46e-06	1.12	0.265	-2.08e-06	7.57e-06
68	6.53e-06	5.67e-06	1.15	0.250	-4.58e-06	.0000176
69	1.34e-06	1.28e-06	1.05	0.295	-1.17e-06	3.86e-06
70	3.25e-06	3.02e-06	1.08	0.282	-2.67e-06	9.16e-06
71	6.53e-07	6.61e-07	0.99	0.323	-6.43e-07	1.95e-06
72	1.60e-06	1.59e-06	1.01	0.312	-1.51e-06	4.71e-06
73	3.16e-07	3.38e-07	0.93	0.350	-3.47e-07	9.78e-07
74	7.86e-07	8.25e-07	0.95	0.341	-8.32e-07	2.40e-06
75	1.52e-07	1.71e-07	0.89	0.376	-1.84e-07	4.87e-07
76	3.83e-07	4.25e-07	0.90	0.368	-4.51e-07	1.22e-06
77	7.26e-08	8.62e-08	0.84	0.400	-9.64e-08	2.42e-07
78	1.85e-07	2.17e-07	0.85	0.394	-2.40e-07	6.11e-07
79	3.46e-08	4.31e-08	0.80	0.422	-4.99e-08	1.19e-07
80	8.93e-08	1.10e-07	0.81	0.417	-1.26e-07	3.05e-07
81	1.64e-08	2.14e-08	0.77	0.443	-2.55e-08	5.84e-08
82	4.28e-08	5.54e-08	0.77	0.440	-6.57e-08	1.51e-07

The only new thing in this command is the option predict(pr(4)) which is telling Stata that instead of calculating the expected number outcomes that

it should calculate the probability that the outcome will be four. We use the at() option in order to calculate these probabilities for different values of the variable gpa and gender. We then use the marginsplot command to produce the graph shown in Figure 2.8. If you look at the title of the y-axis, Stata is clearly telling us that the predicted values are the probabilities that the total failed courses is equal to four.

```
. marginsplot, noci

  Variables that uniquely identify margins: gpa gender
```

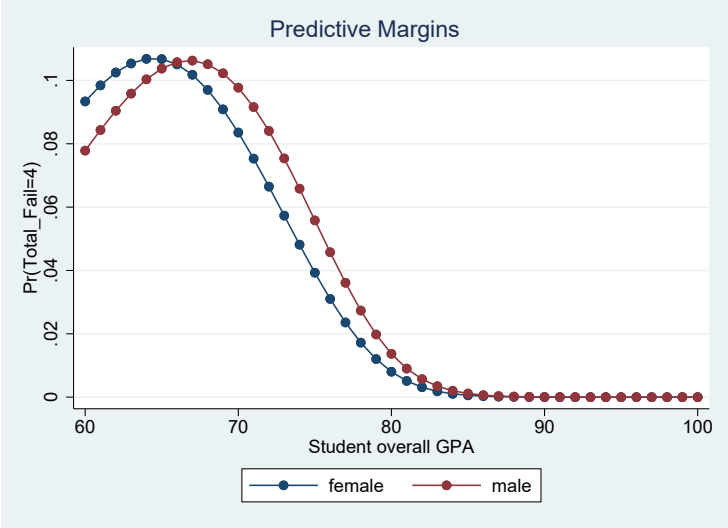

Figure 2.8: Plotting the probability that the event will occur exactly four times.

Chapter 3

References

Hilbe, J.M. (2011). Negative Binomial Regression. 2nd edition. Cambridge University Press.

Long, J.S. & Freese, J. (2014). Regression Models for Categorical Dependent Variables using Stata. 3rd edition. Stata Press.

Mitchell, M.N. (2012). Interpreting and Visualizing Regression Models using Stata. Stata Press.