1

Introduction: Count Data Containing Dispersion

This chapter is an overview summarizing relevant, established, and well-studied distributions for count data that motivate the consideration of the Conway–Maxwell–Poisson (COM–Poisson) distribution. Each of the discussed models provides an improved flexibility and computational ability for analyzing count data; yet associated restrictions help readers to appreciate the need for and usefulness of the COM–Poisson distribution, thus resulting in an explosion of research relating to this model. For completeness of discussion, each of these sections includes discussion of the relevant R packages and their contained functionality to serve as a starting point for forthcoming discussions throughout subsequent chapters. Along with the R discussion, illustrative examples aid readers in understanding distribution qualities and related statistical computational output. This background provides insights into the real implications of apparent data dispersion in count data models and the need to properly address it.

This introductory chapter proceeds as follows. Section 1.1 introduces the most well-known model for count data: the Poisson distribution. Its probabilistic and statistical properties are discussed, along with R tools to perform computations. Section 1.2, however, notes a major limitation of the Poisson distribution – namely its inability to properly model dispersed count data. Focusing first on the phenomenon of data over-dispersion, this section focuses attention on the negative binomial (NB) distribution – the most popular count distribution that allows for data over-dispersion. Section 1.3 meanwhile recognizes the existence of count data that express data under-dispersion and the resulting need for model consideration that can accommodate this property. While several flexible models allowing for data over- or under-dispersion exist in the literature, this section focuses attention on the generalized Poisson (GP) distribution for modeling such data because it is arguably (one of) the most popular option(s) for modeling

such data. Section 1.4 offers an overarching perspective about these models as special cases of a larger class of weighted Poisson distributions. Finally, Section 1.5 motivates an interest in the COM–Poisson distribution and summarizes the rest of the book, including the unifying background that will be referenced in subsequent chapters.

1.1 Poisson Distribution

The Poisson distribution is the most studied and applied distribution referenced to describe variability in count data. A random variable X with a Poisson(λ) distribution has the probability mass function

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}, \quad x = 0, 1, 2, \ldots, \tag{1.1}$$

or, on the log-scale,

$$\begin{aligned} \ln[P(X = x)] &= x \ln \lambda - \ln(x!) - \lambda \\ &= x \ln \lambda - \sum_{j=1}^{x} \ln(j) - \lambda, \end{aligned}$$

where λ is the associated intensity parameter; illustrative examples of the distributional form assuming various values of λ are provided in Figure 1.1.

Derived as the limiting distribution of a binomial(n, p) distribution where $n \to \infty$ and $p \to 0$ such that $np = \lambda$, the beauty of this distribution lies in its simplicity. Both its mean and variance equal the intensity parameter λ; thus, the dispersion index is

$$\mathrm{DI}(X) = \frac{V(X)}{E(X)} = \frac{\lambda}{\lambda} = 1. \tag{1.2}$$

The probability mass function satisfies the recursion

$$\frac{P(X = x - 1)}{P(X = x)} = \frac{x}{\lambda}, \tag{1.3}$$

with its moment generating function $M_X(t) = e^{\lambda(e^t - 1)}$, and the Poisson distribution is a member of the exponential family of the form

$$P(X = x; \theta) = H(x) \exp[\eta(\theta) T(x) - \Psi(\theta)], \quad x \in \mathbb{N}, \tag{1.4}$$

where, for $\theta = \lambda$, $\eta(\theta) = \ln(\lambda)$, $\Psi(\theta) = \lambda$, $T(x) = x$, and $H(x) = (x!)^{-1}$. The simplicity of the Poisson distribution, however, can also be viewed as theoretically constraining and not necessarily representative of real count

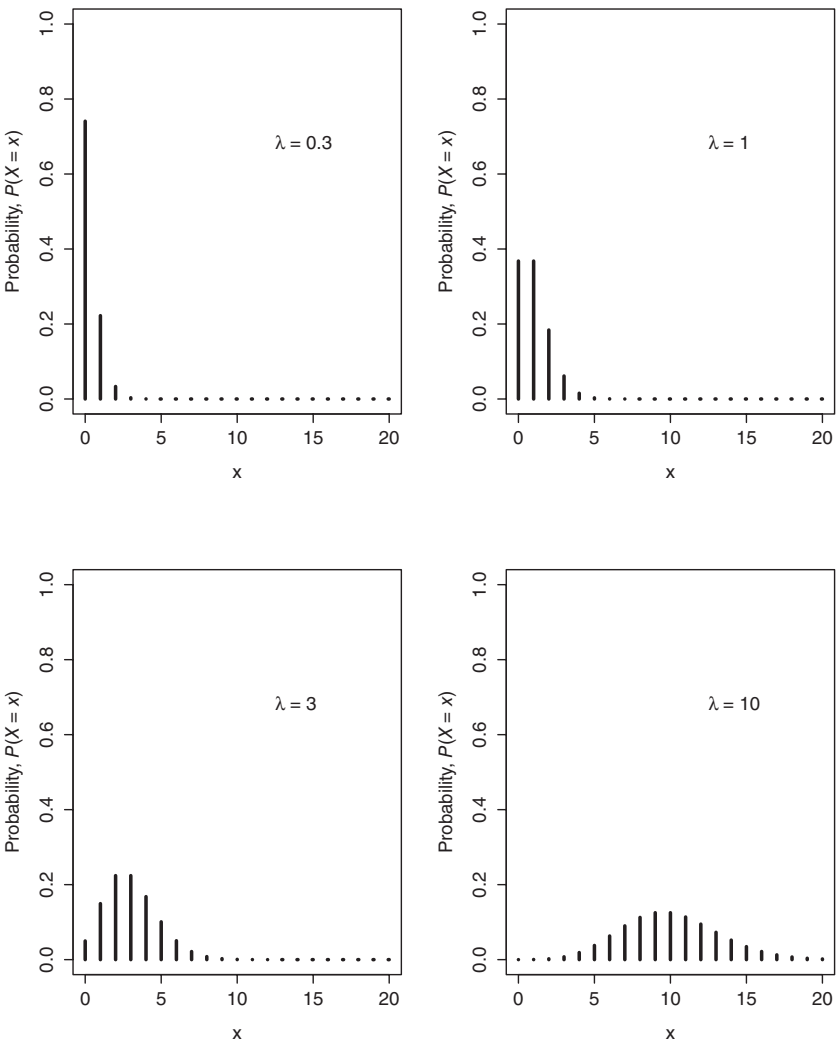

Figure 1.1 Poisson probability mass function illustrations for $\lambda \in \{0.3, 1, 3, 10\}$.

data distributions. Thus, applying statistical methods that are motivated and/or developed by the Poisson model assumption can cause significant repercussions with regard to statistical inference. This matter is discussed in more detail in the subsequent sections in Chapter 1 and throughout this reference.

1.1.1 R Computing

The `stats` package contains functions to compute the probability, distribution function, quantile function, and random number generation associated with the Poisson distribution. All of the relevant commands require the Poisson rate parameter λ (`lambda`) as an input value. The `dpois` function computes the probability/density $P(X = x)$ for a random variable X at observation x. The command has the default setting as described (`log = FALSE`), while changing the indicator input to `log = TRUE` computes the probability on the natural-log scale. The `ppois` function computes the cumulative probability $P(X \leq q)$ given a quantile value q, while `qpois` determines the quantile q (i.e. the smallest integer) for which the cumulative probability $P(X \leq q) \geq p$ for some given probability p. This quantile determination stems from the discrete nature of the Poisson probability distribution. Both commands contain the default settings `lower.tail = TRUE` and `log.p = FALSE`. The condition `lower.tail = TRUE` infers interest regarding the cumulative probability $P(X \leq q)$ while `lower.tail = FALSE` focuses on its complement $P(X > q)$ (i.e. the upper tail). The indicator `log.p = FALSE` (`TRUE`) meanwhile infers whether to consider probabilities on the original or natural-log scale, respectively. Finally, the `rpois` function produces a length n (`n`) vector of count data randomly generated via the Poisson distribution.

Demonstrative examples utilizing the respective functions are provided in Code 1.1, all of which assume the Poisson rate parameter $\lambda = 3$. The command `dpois(x=5, lambda=3)` determines that $P(X = x) = 0.1008188$; this value is illustrated in Figure 1.1 for $\lambda = 3$. Meanwhile, `dpois(x=5, lambda=3, log = TRUE)` shows that $\ln(P(X = x)) = \ln(0.1008188) = -2.29443$. The `ppois` functions demonstrate the difference between computing the lower versus upper tail, respectively; naturally, the sum of the two results equals 1. The command `qpois(p=0.9, lambda=3)` produces the expected result of 5 because we see that the earlier `ppois(q=5, lambda=3)` result showed that $P(X \leq 5) = 0.9160821 > 0.9$. Meanwhile, one can see that `qpois(p=0.9, lambda=3, lower.tail = FALSE)` produces the value 1 by considering the corresponding `ppois` commands:

`ppois(q=0, lambda=3, lower.tail=FALSE)` produces the result 0.9502129
`ppois(q=1, lambda=3, lower.tail=FALSE)` produces the result 0.8008517.

Recall that the discrete nature of the Poisson distribution requires a modified approach for determining the quantile value; the resulting quantile is

Code 1.1 Examples of R function use for Poisson distributional computing: dpois, ppois, qpois, rpois.

```
> dpois(x=5, lambda=3)
[1] 0.1008188
> dpois(x=5, lambda=3, log = TRUE)
[1] -2.29443
> ppois(q=5, lambda=3)
[1] 0.9160821
> ppois(q=5, lambda=3, lower.tail = FALSE)
[1] 0.08391794
> qpois(p=0.9, lambda=3)
[1] 5
> qpois(p=0.9, lambda=3, lower.tail = FALSE)
[1] 1
> rpois(n=10, lambda=3)
 [1] 3 4 3 5 2 0 5 5 4 3
```

determined such that the cumulative probability of interest is at least as much as the desired probability of interest. This definition suggests that, when considering the upper tail probability, the resulting quantile now implies that the corresponding upper tail probability is no more than the desired probability of interest. As noted above, $P(X > 0) = 0.9502129$ and $P(X > 1) = 0.8008517$; because the desired upper tail probability in the example is 0.9, we see that 0 produces an upper tail probability that is too large for consideration, while the upper tail probability associated with 1 is the first integer that satisfies $P(X > x) \leq 0.9$, thus producing the solution as 1. Finally, for completeness, the `rpois` function produces 10 randomly generated potential observations stemming from a Poisson(3) distribution. Given the probability mass function illustration provided in Figure 1.1 for $\lambda = 3$, these outcomes appear reasonable.

1.2 Data Over-dispersion

Over-dispersion (relative to a comparable Poisson model) describes distributions whose variance is larger than the mean, i.e. $\text{DI}(X) > 1$ for a random variable X. This is a well-studied phenomenon that occurs in most real-world datasets. Over-dispersion can be caused by any number of situations, including data heterogeneity, the existence of positive correlation between responses, excess variation between response probabilities or counts, and violations in data distributional assumptions. Apparent over-dispersion can also exist in datasets because of outliers or, in the case of regression

models, the model may not include important explanatory variables or a sufficient number of interaction terms, or the link relating the response to the explanatory variables may be misspecified. Under such circumstances, over-dispersion causes problems because resulting standard errors associated with parameter estimation may be underestimated, thus producing biased inferences. Interested readers should see Hilbe (2007) for a comprehensive discussion regarding over-dispersion and its causes.

The most popular distribution to describe over-dispersed data is the NB distribution. A random variable X with an NB(r, p) distribution has the probability mass function

$$P(X = x) = \binom{r + x - 1}{x} p^x (1 - p)^r \tag{1.5}$$

$$= \frac{\Gamma(r + x)}{x!\Gamma(r)} p^x (1 - p)^r, \qquad x = 0, 1, 2, \ldots, \tag{1.6}$$

and can be viewed as the probability of attaining a total of x successes with $r > 0$ failures in a series of independent Bernoulli(p) trials, where $0 < p < 1$ denotes the success probability associated with each trial. Alternatively, the NB distribution can be derived via a mixture model of a Poisson(λ) distribution, where λ is gamma distributed[1] with shape and scale parameters, r and $p/(1 - p)$, respectively. The latter approach is a common technique for addressing heterogeneity. Other possible distributions for λ include the generalized gamma (which produces a generalized form of the NB distribution (Gupta and Ong, 2004)), the inverse Gaussian, and the generalized inverse Gaussian (which produces the Sichel distribution (Atkinson and Yeh, 1982; Ord and Whitmore, 1986)). Various other mixing distributions have also been considered; see Gupta and Ong (2005) for discussion.

The moment generating function of the NB(r, p) random variable X is

$$M_X(t) = \left(\frac{p}{1 - (1 - p)e^t} \right)^r, \quad t < -\ln(1 - p),$$

[1] For a gamma(α, β) distributed random variable X with shape and scale parameters α and β, respectively, its probability density function (pdf) is $f(x) = \frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} e^{-x/\beta}$ (Casella and Berger, 1990).

which produces a respective mean and variance,

$$\mu \doteq E(X) = \frac{r(1-p)}{p} \quad \text{and} \tag{1.7}$$

$$V(X) = \frac{r(1-p)}{p^2} = \mu + \frac{1}{r}\mu^2, \tag{1.8}$$

where $r > 0$ can be viewed as a dispersion parameter. Given the dispersion parameter r, this distribution can be represented as an exponential family (Equation (1.4)), where $\theta = p$, $H(x;r) = \binom{r+x-1}{x}$, $T(x) = x$, $\eta(p) = \ln p$, and $\psi(p;r) = r\ln(1-p)$. Equation (1.8) demonstrates that the NB distribution can accommodate data over-dispersion ($\text{DI}(X) > 1$) because one can clearly see that the distribution's variance is greater than or equal to its mean since $r > 0$. Further, the NB distribution contains the Poisson as a limiting case; as $r \to \infty$ and $p \to 1$ such that $r(1-p) \to \lambda$, $0 < \lambda < \infty$, not only do the NB mean and variance both converge to λ, but the NB probabilities likewise converge to their respective Poisson counterparts. Figure 1.2 illustrates the distributional convergence of the NB(r,p) to the Poisson$(\lambda = 3)$ distribution, where $r \to \infty$ and $p \to 1$ such that $r(1-p) = 3$. The NB distribution likewise contains the geometric(p) as a special case when $r = 1$.

The NB distribution can alternatively be represented as NB$(r, r/(r+\mu))$ with the probability mass function

$$P(X = x) = \binom{x+r-1}{x}\left(\frac{r}{r+\mu}\right)^x\left(\frac{\mu}{r+\mu}\right)^r, \quad x = 0,1,2,\ldots, \tag{1.9}$$

where $r > 0$, $\mu > 0$; this formulation explicitly has a mean μ and a variance $\mu + \mu^2/r$. The `MASS` package in `R` utilizes this parametrization and defines the dispersion parameter as `theta` such that $V(X) = \mu + \mu^2/\theta$, i.e. $\theta \doteq r$; we will revisit this in Chapter 5. While the NB distribution has been well studied and statistical computational ability is supplied in numerous software packages (e.g. `R` and `SAS`), an underlying constraint regarding the NB distribution leads to its inability to address data under-dispersion (i.e. the dispersion index is less than 1, or the variance is smaller than the mean).

1.2.1 `R` Computing

The `stats` package provides functionality for determining the probability, distribution function, quantile function and random number generation for the NB distribution. These commands all require the inputs `size` (r) and either the success probability p (`prob`) or mean μ (`mu`), depending on the

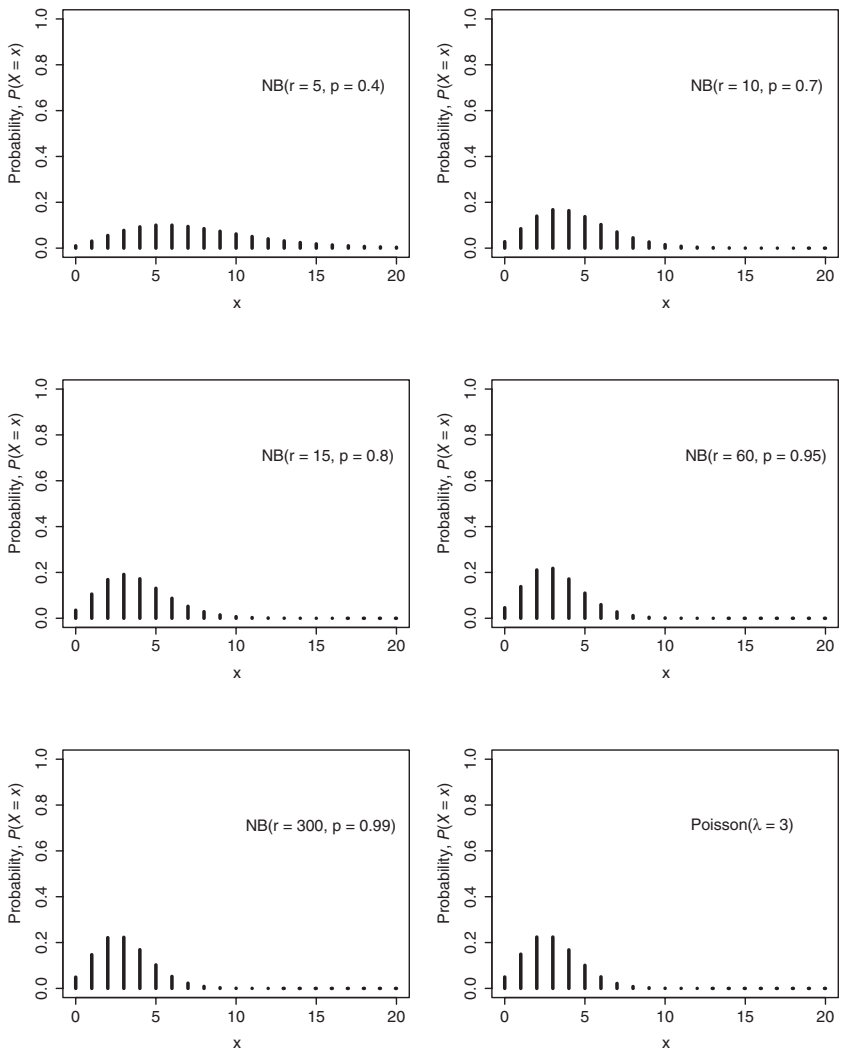

Figure 1.2 Negative binomial distribution illustrations for values of $(r, p) \in \{(5, 0.4), (10, 0.7), (15, 0.8), (60, 0.95), (300, 0.99)\}$ and the Poisson($\lambda = 3$) probability mass function. This series of density plots nicely demonstrates the distributional convergence of the negative binomial to the Poisson as $r \to \infty$ and $p \to 1$ such that $r(1 - p) \to \lambda$.

choice of parametrization. The function `dnbinom` computes the probability $P(X = x)$ for a random variable X at observation x, either on the original scale (`log = FALSE`; this is the default setting) or on a natural-log scale

Code 1.2 Examples of R commands for NB distributional computing: `dnbinom`, `pnbinom`, `qnbinom`, `rnbinom`.

```
> dnbinom(x=5, size=10, prob=0.7)
[1] 0.1374203
> dnbinom(x=5, size=10, prob=0.7, log = TRUE)
[1] -1.984712
> pnbinom(q=5, size=10, prob=0.7)
[1] 0.7216214
> pnbinom(q=5, size=10, prob=0.7, lower.tail = FALSE)
[1] 0.2783786
> qnbinom(p=0.9, size=10, prob=0.7)
[1] 8
> qnbinom(p=0.9, size=10, prob=0.7, lower.tail = FALSE)
[1] 1
> rnbinom(n=10, size=10, prob=0.7)
 [1] 1 8 7 3 5 8 4 2 5 3
```

(`log = TRUE`). For a given quantile value q, the `pnbinom` function determines the cumulative probability $P(X \leq q)$, where the default settings, `lower.tail = TRUE` and `log.p = FALSE`, imply that the resulting cumulative probability is attained by accumulating the probability from the lower tail and on the original probability scale. The command `qnbinom` meanwhile determines the smallest discrete quantile value q that satisfies the cumulative probability $P(X \leq q) \geq p$ for a given probability p. This function likewise assumes the default settings, `lower.tail = TRUE` and `log.p = FALSE`, such that the quantile q is determined from the lower tail and on the original probability scale. For both of these commands, changing the default settings to `lower.tail = FALSE` and `log.p = TRUE`, respectively allows analysts to instead consider quantile determination on the basis of the upper tail probability $P(X > q)$, and via a probability computation on the basis of the natural-log scale. Finally, the `rnbinom` function randomly generates n (`n`) observations from an NB distribution with the specified size (`size`) and success probability (`prob`).

The NB($r = 10$, $p = 0.7$) distribution is provided in Figure 1.2 and serves as a graphical reference for the illustrative commands featured in Code 1.2. All of the demonstrated functions assume $r = 10$ and $p = 0.7$ as the associated NB size and success probability parameters. The first command (`dnbinom(x=5, size=10, prob=0.7)`) shows that $P(X = x) = 0.1374203$; this probability is shown in the associated plot in Figure 1.2. Meanwhile, `dnbinom(x=5, size=10, prob=0.7, log = TRUE)` shows that $\ln(P(X = x)) = \ln(0.1374203) = -1.984712$.

The `pnbinom` functions show the results when computing the lower versus upper tail, respectively; naturally, the sum of the two computations equals 1. Calling `qnbinom(p=0.9, size=10, prob=0.7)` produces the result 8, while `qnbinom(p=0.9, size=10, prob=0.7, lower.tail = FALSE)` yields the value 1. Finally, the `rnbinom` command produces 10 randomly generated potential observations stemming from an NB($r = 10, p = 0.7$) distribution.

1.3 Data Under-dispersion

Where data over-dispersion describes excess variation in count data, under-dispersion describes deficient variation in count data. Data under-dispersion (relative to the Poisson model) refers to count data that are distributed such that the variance is smaller than the mean, i.e. its dispersion index $\text{DI}(X) < 1$ for a random variable X.

There remains some measures of debate regarding the legitimacy of data under-dispersion as a real concept. Some researchers attribute under-dispersion to the data generation (e.g. small sample values) or to the modeling process (e.g. model over-fitting), noting that the arrival process, birth–death process, or binomial thinning mechanisms can also lead to under-dispersion (Kokonendji, 2014; Lord and Guikema, 2012; Puig et al., 2016). As an example, for renewal processes where the distribution of the interarrival times has an increasing hazard rate, the distribution of the number of events is under-dispersed (Barlow and Proschan, 1965). Efron (1986), however, argues that "there are often good physical reasons for not believing in under-dispersion, however, especially in binomial and Poisson situations."

Whether real or apparent, examples across disciplines are surfacing with more frequency where data under-dispersion is present; thus there exists the need to represent such data. The most popular model that can accommodate data dispersion (whether over- or under-dispersion) is the GP distribution – a flexible two-parameter distribution that contains the Poisson distribution as a special case (Consul, 1988). A random variable X that is GP(λ_1, λ_2) distributed has the probability mass function

$$P(X = x) = \begin{cases} \dfrac{\lambda_1(\lambda_1 + \lambda_2 x)^{x-1}}{x!} \exp(-\lambda_1 - \lambda_2 x), & x = 0, 1, 2, \ldots \\ 0, & x \geq m \text{ where } \lambda_1 + \lambda_2 m \leq 0 \end{cases} \tag{1.10}$$