Generalized Linear Models for Count Outcomes and Zero-Inflated Count Outcomes

- Topics:
  - Roadmap of models for non-normal outcomes
  - Generalized linear models for count outcomes
  - Model adjustments for misfit to Poisson distribution
    - Overdispersion, missing zeros, and too many zeros
  - Summary of options in SAS, STATA, and R software for estimating models predicting discrete outcomes

More General*ized* Linear Models

- **General*ized* linear models:** link-transformed conditional mean is predicted by the linear model; ML estimator uses not-normal conditional distributions in the outcome data likelihood
  - **Btw, in multilevel models**, level-1 conditional model has some not-normal distribution, but level-2 random effects are usually multivariate normal

- **Two parts: Link function** + **other conditional distribution**
  - ***Done:*** **Categorical → Logit/Probit/Log-Log/Comp-Log-Log**
    - **Bernoulli for binary; multinomial for ordinal or nominal**
  - ***Now:*** **Counts → Log + some kind of Poisson or Negative Binomial**
    - **Zero-inflated counts → zero-inflated or hurdle variants**
  - ***Later:*** **Proportions → Logit** + **some kind of Binomial/Beta**
  - ***Later:*** **Truncated/Bounded → Tobit** + **normal**
  - ***Later:*** **Skewed Continuous → Log** + **Log-Normal/Gamma**
    - **Zero-inflated continuous → hurdle variants**

A Taxonomy of Not-Normal Outcomes

- **"Discrete" outcomes**—all responses are **whole** numbers
  - **Categorical variables** in which **values are labels**, not numbers
    - Bernoulli (2 options) or multinomial (3+ options) distributions
    - Question: Are the values ordered → **Which link function?**
  - **Count of things that happened**, so values < 0 cannot exist
    - Outcome values range from 0 to +∞ (whole numbers only)
    - Usually some kind of Poisson or Negative Binomial distribution
    - **Usually log link so predicted outcomes can't go below 0**
    - Questions: Which conditional distribution? Are there *extra* 0 values?

- **"Continuous" outcomes**—responses can be **any** number
  - Question: What does the residual distribution look like?
    - Symmetric or skewed? Is there an arbitrary boundary (=censored)?

Natural Log Link for Count Outcomes

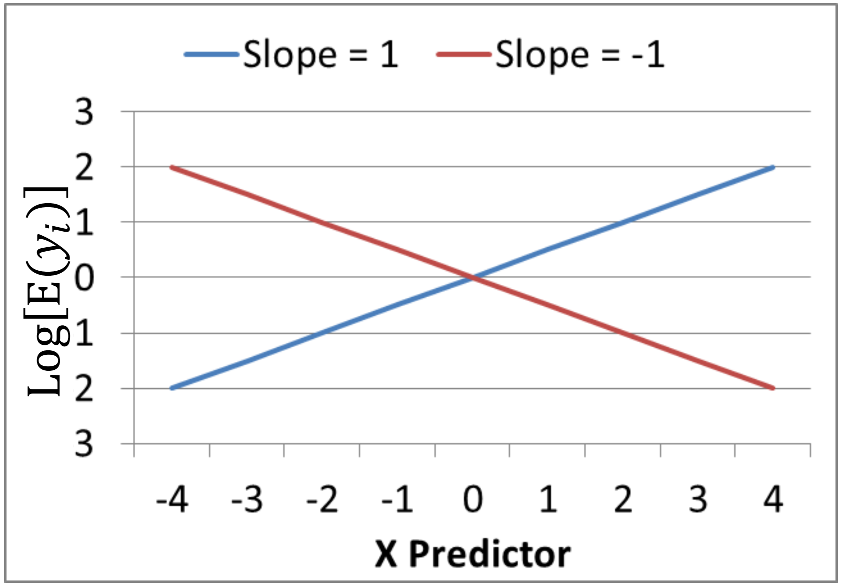
$\mathrm{E}(y_i)$ → the model-predicted outcome from linear predictor

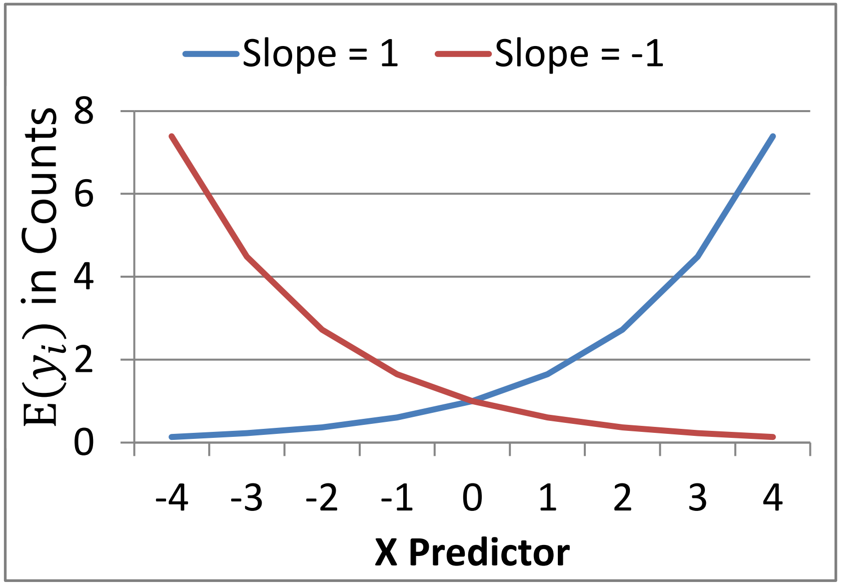
This is an ***unbounded* linear model** that predicts the log of the expected count…

$$\mathrm{Log}[\mathrm{E}(y_i)] = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}(\boldsymbol{x_i})$$

…that becomes an expected count bounded at 0 via an inverse link of exp(log count):

$$[\mathrm{E}(y_i)] = \mathbf{exp}[\boldsymbol{\beta_0} + \boldsymbol{\beta_1}(\boldsymbol{x_i})]$$





Models for Count Outcomes

- **Counts**: non-negative integer responses (unbounded positive)
  - **Link:** $\mathrm{g}(\bullet)$, where $\bullet = \mathrm{E}(y_i)$ → $\mathrm{Log}[\mathrm{E}(y_i)] = \mathrm{Log}(\mu_i) = [\text{linear predictor}]$
  - **Inverse Link:** $\mathrm{g}^{-1}(\bullet)$, where $\bullet = \mathrm{E}(y_i)$ → $\mathrm{E}(y_i) = \exp(\hat{y}_i)$
  - e.g., if model-scale predicted log of expected count: $\mathrm{Log}(\mu_i) = \hat{y}_i = -1$, the data-scale expected count is: $\mu_i = \exp(-1) = 0.368$
    - So even though counts are only integers, expected counts are not!
  - Btw, you can control for differences in time measured via an "**offset"** (or **"exposure"**) log-transformed predictor variable whose slope is fixed =1

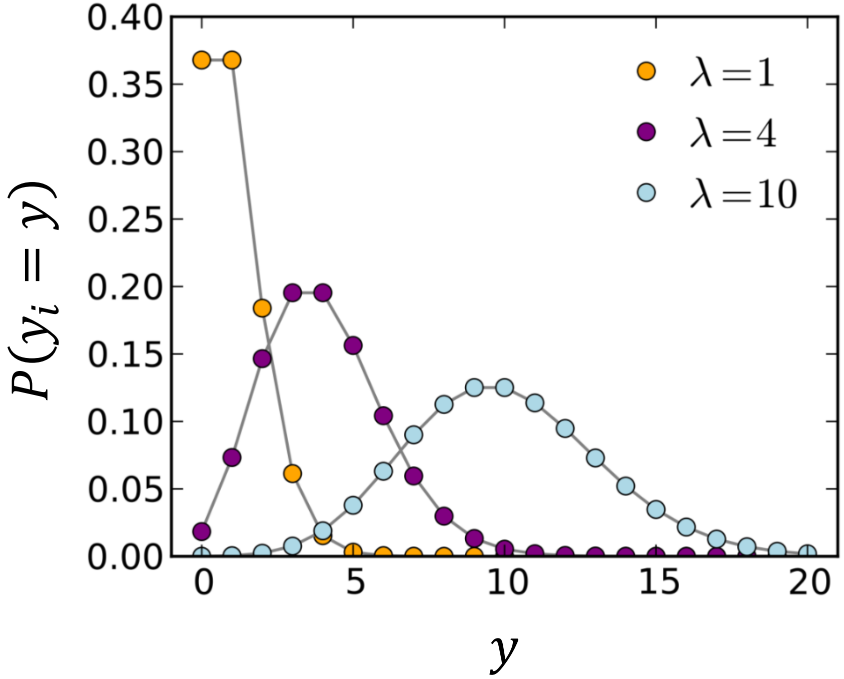
- $\mathbf{exp(\boldsymbol{\beta}_x)}$ gives an effect size called an "**incidence-rate ratio**" (**IRR**) that is on same scale as an odds ratio (IRR = 1 means no effect)
  - e.g., IRR = 1.25 for $x_i = 0$ vs 1? Then $x_i = 1$ counts are "25% higher"
  - e.g., IRR = 0.75 for $x_i = 0$ vs 1? Then $x_i = 1$ counts are "25% lower"

- Choosing the "right" **conditional distribution** is the tricky part!
  - **This could be a whole semester by itself—here are the highlights…**

Poisson Conditional Distribution

- Poisson distribution has **one parameter, λ,** which is both its mean and its variance (in which λ below is μ in our notation)
- PDF: $f(y_i) = \text{prob}(y_i = y) = \frac{\exp(-\mu)*\mu^y}{y!}$

$y!$ = factorial of y = gamma function $\Gamma(y+1)$



The dots are used to indicate that only integer values are observed.

Distributions with a small expected value (as given by μ or λ here) are predicted to have a lot of 0 values.

Once $\lambda > 6$ or so, the shape of the distribution is close to normal.

- In SAS GLIMMIX LINK=LOG, DIST=POISSON; STATA POISSON or GLM link(log) family(poisson); R GLM family="poisson" or VGAM; also Mplus

Image borrowed from: https://en.wikipedia.org/wiki/Poisson_distribution

3 potential problems with Poisson…

- The standard Poisson distribution is rarely sufficient, though

- **Problem #1: When mean ≠ variance**
  - If variance < mean, this leads to "under-dispersion" (not as likely)
  - If variance > mean, this leads to "over-dispersion" (happens frequently)

- **Problem #2: When there are *no* 0 values**
  - Some 0 values are expected from count models, but in some contexts $y_i > 0$ always (but subtracting 1 won't fix it correctly; need to adjust the model)

- **Problem #3: When there are *too many* 0 values**
  - Some amount of 0 values are expected from count distributions already, but in many cases, there are even more 0 values observed than that
  - To fix it, there are two main options, depending on what you do to the 0's

- Each of these problems requires a model adjustment to fix it…

Problem #1: Variance > mean = over-dispersion

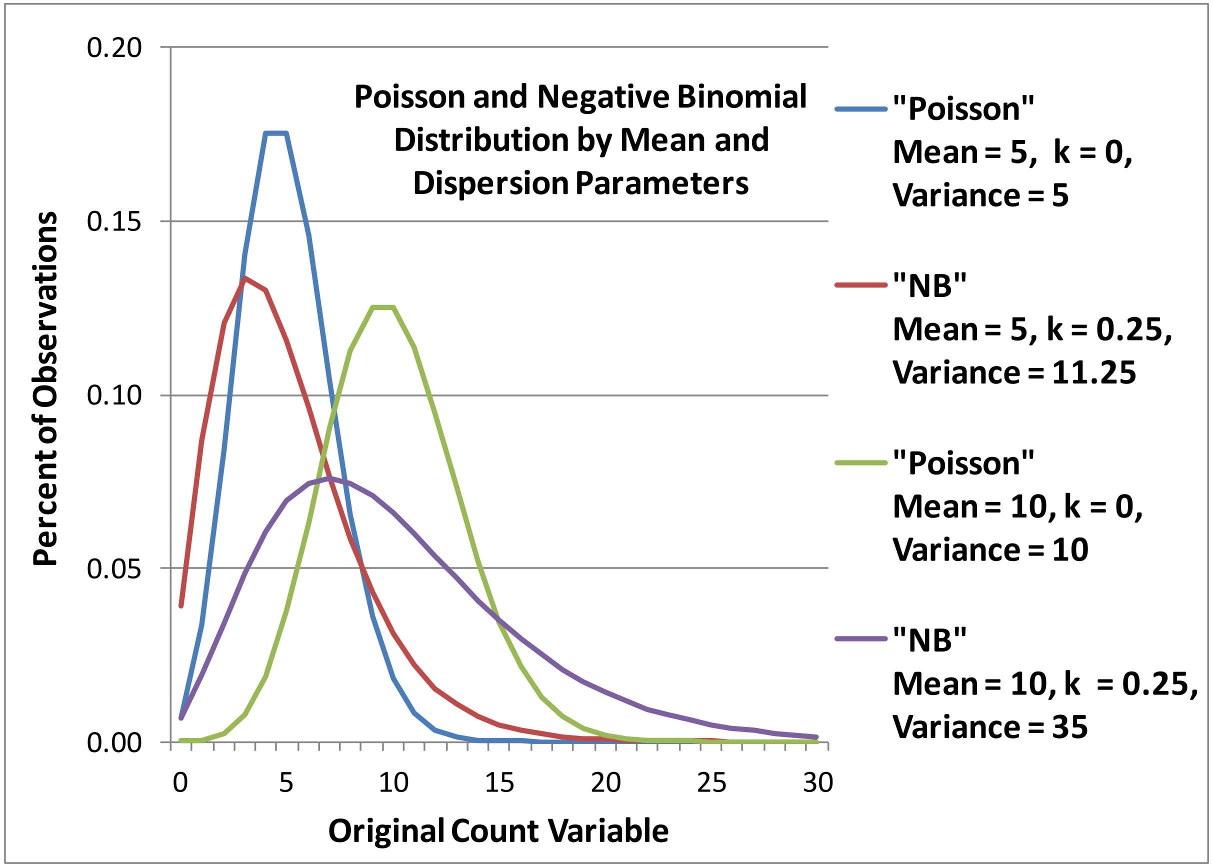
- To fix it, we must add a parameter that allows the variance to exceed the mean… it then can become a **Negative Binomial (Negbin)** distribution
  - Two types of extra variance: constant = NB-1, quadratic = NB-2 (better)

- **NB-2** has **mean = $\boldsymbol{\mu}$** and **dispersion** = **"scale" $\boldsymbol{k}$** (or $1/k = \theta$ instead):
  - PDF: $\text{prob}(y_i = y\ ) = \frac{\Gamma\left(y+\frac{1}{k}\right)}{\Gamma(y+1)*\Gamma\left(\frac{1}{k}\right)} * \left(\frac{1}{1+k\mu}\right)^{\frac{1}{k}} * \left(1-\frac{1}{1+k\mu}\right)^{y}$
  - $\boldsymbol{k}$ **is a multiplier:** $\text{Var}(y_i) = \mu + \boldsymbol{k}\mu^2$ (so Negbin ≈ Poisson if $k = 0$)
  - Can test if $k > 0$ via LRT (−2ΔLL), although LL for $k = 0$ is undefined
  - In SAS GLIMMIX DIST = NEGBIN (as k = "scale"); STATA NBREG or GLM (as k = "alpha"); R VGAM, MASS (as θ), or PSCL (as θ); more about R here

$$\boxed{\begin{array}{c}\text{Log}(\mu) = \hat{y} + \epsilon, \\ \exp(\epsilon)\sim gamma\end{array}}$$

- An alternative model based on the same idea is **generalized Poisson**:
  - Mean: $\frac{\mu}{1-k}$, Variance: $\frac{1}{(1-k)^3}$, so LL is actually defined for $k = 0$
  - Much less commonly used, but it's in SAS FMM (and in SAS GLIMMIX via user-defined functions); STATA GPOISSON; R VGAM; also Mplus

Negative Binomial (NB) = "Stretchy" Poisson…



$\text{Mean} = \mu$
$\text{Dispersion} = k$

$$\text{Var}(y_i) = \mu + k\mu^2$$

A Negative Binomial model can be useful for count outcomes with extra skewness.

- Because its k dispersion parameter is fixed to 0, the Poisson model is nested within the Negative Binomial model—to test improvement in fit:
- Is $-2\left(LL_{Poisson} - LL_{NegBin}\right) > 3.84$ for $DF = 1$? Then if $p < .05$, keep NB
- Should use a mixture of $DF = 0$ and $DF = 1$: use $-2\Delta LL > 2.71$ instead

Pause: Clarifying Terminology

Unfortunately, the same words can refer to many different things, such as:

- "**Nonlinear**"—this adjective could refer to:
    - A category of model that's not in "slope*variable + slope*variable" form
    - A consequence of a slope (e.g., that creates a quadratic relationship)
    - A type of regression (e.g., using link functions, as opposed to "linear")

- "**Fit**"—this could mean:
    - Mistakenly used to refer to predictive quality (i.e., amount of variance explained)—this is NOT fit, it's overall model effect size
        - LRTs against "saturated" predictor models fall into this category
    - In **multivariate** models, **fit usually refers to the match** between the model-predicted and data-estimated covariance matrices (or cross-tabs), and has nothing to do with effect size in terms of model predictive quality
        - e.g., in SEM, things like RMSEA and SRMR; LRT against H1 saturated model
    - In generalized linear models, **fit can also refer to the match of the chosen conditional distribution to the observed outcome**… *this is the one I am talking about next!*

Absolute Conditional Distribution Fit

- In addition to comparing the **relative fit** of the Poisson and Negative Binomial (NB) distributions, we also need to examine the "**absolute fit**" of the **conditional distribution** for the observed outcome distribution
  - e.g., NB may be relatively better than Poisson, but is NB "good enough"?

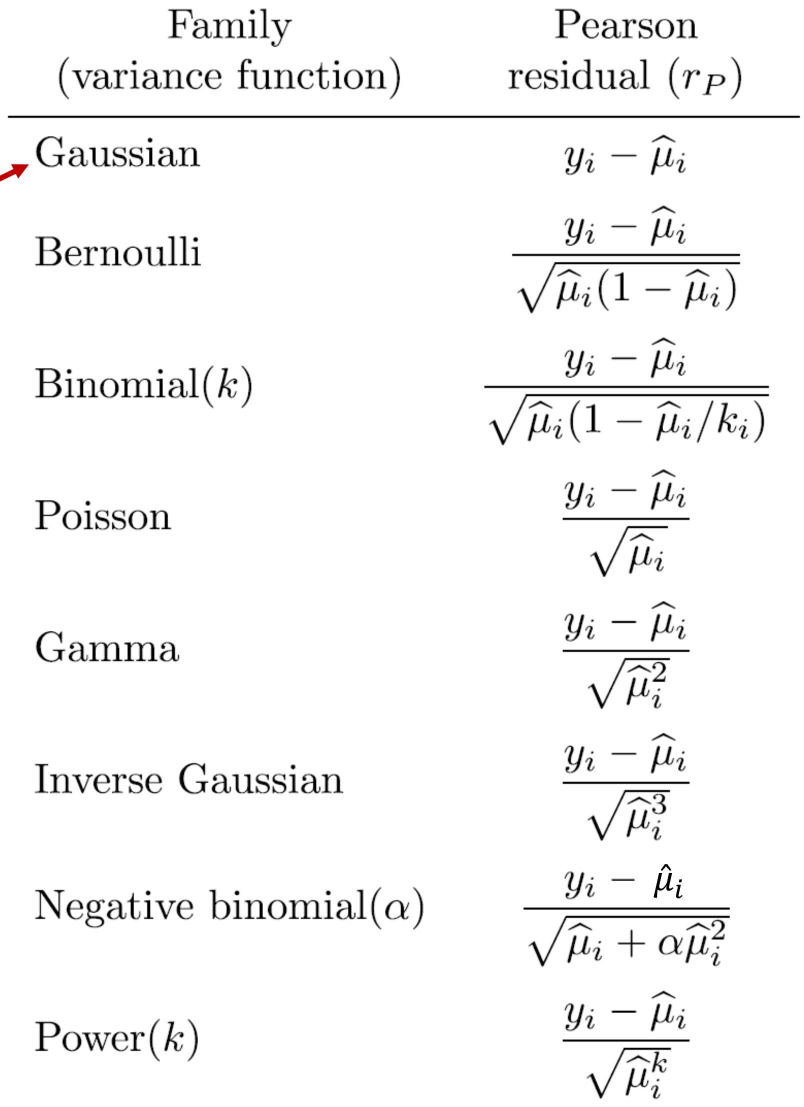
- **Conditional distribution fit** can be examined using a statistic given as **Pearson χ^2 / degrees of freedom** in which 1 = good fit
  - Sum over persons of $\left(\frac{y_i - \mu_i}{\text{SD at } \mu_i}\right)^2$, then divide by N (− #parms) as DF
  - = squared average residual / model-expected residual (should be same, 1)
  - Available in SAS via GENMOD or GLIMMIX (possibly others); in STATA via GLM (and more tests in COUNTFIT); in R manually by summing residuals
  - See details on next slide for different conditional distributions

- Btw, Hardin & Hilbe (2018) describe other "generalized" negative binomial models, including the "heterogeneous negative binomial" in which the dispersion scale factor itself can be predicted—see Example 3 for a demo

Pearson Residuals by Distribution

Table A.9 (Hardin & Hilbe, 2018)
(where μ_i = $\hat{\mu}_i$ in their notation)

- Gaussian (normal) doesn't have a denominator, so Pearson χ^2 / DF
  → **is residual variance (and NOT misfit of normal distribution**)
  - So please ignore this part of ALL my class materials from the 2020 version of this class (my bad)!
- For others, Pearson $\chi^2/DF > 1$ indicates **overdispersion**
  → too much variance relative to what the model expects



Family (variance function)	Pearson residual (r_P)
Gaussian	$y_i - \widehat{\mu}_i$
Bernoulli	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i(1-\widehat{\mu}_i)}}$
Binomial(k)	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i(1-\widehat{\mu}_i/k_i)}}$
Poisson	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i}}$
Gamma	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i^2}}$
Inverse Gaussian	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i^3}}$
Negative binomial(α)	$\dfrac{y_i - \hat{\mu}_i}{\sqrt{\widehat{\mu}_i + \alpha\widehat{\mu}_i^2}}$
Power(k)	$\dfrac{y_i - \widehat{\mu}_i}{\sqrt{\widehat{\mu}_i^k}}$

Hardin, J.W. & Hilbe, J. M. (2018). Generalized linear models and extensions (4th ed.). STATA Press.

Problem #2: There are **no** 0 values

- "**Zero-Altered**" = "**Zero-Truncated**" Poisson or Negative Binomial: ZAP/ZANB or ZTP/ZTNB (used in hurdle models)
  - Is usual count distribution, just not including any 0 values
  - In SAS PROC FMM using DIST=TRUNCPOISSON for ZTP or DIST=TRUNCNEGBIN for ZTNB
  - In STATA as TPOISSON (for ZTP) and TNBREG (for ZTNB)
  - R package VGAM option "za__" for Poisson and Negative Binomial
  - Multivariate versions could be fitted in SAS NLMIXED or Mplus, too

- e.g., Poisson PDF: $\mathrm{prob}(y_i = y) = \frac{\mu^y * \exp(-\mu)}{y!}$ $\boxed{\mathrm{Log}(\mu) = \hat{y}}$

- e.g., Zero-Truncated Poisson PDF: $\mathrm{prob}(y_i = y \,|y_i > 0) = \frac{\mu^y * \exp(-\mu)}{y![1-\exp(-\mu)]}$
  - $\mathrm{prob}(y_i = 0) = \exp(-\mu)$, so $\mathrm{prob}(y_i > 0) = 1 - \exp(-\mu)$
  - Divides by probability of non-0 outcomes so probability still sums to 1

Problem #3: **Too many** 0 values, Option #1

- "**Zero-Inflated**" Poisson (ZIP) or NB (ZINB) in SAS GENMOD or COUNTREG; STATA ZIP/ZINB/ZIGP; R VGAM or PSCL; also Mplus
  - Distinguishes **two kinds of 0 values**: **expected** and **inflated/structural** (extra) through a mixture of Bernoulli + Poisson/NegBin/GenPoisson)
  - Creates two submodels to predict "if **extra** 0" and "if not, how much"?
    - The "zero-model" does not readily map onto most hypotheses (in my opinion)
    - But a ZIP example would look like this… (ZINB would add k dispersion, too)

- Submodel 1: $\mathrm{Log}[\mathrm{E}(y_i)] = \mathrm{Log}(\mu_i) = \beta_{0c} + \beta_{1c}(x_i)$
  - Predict **all counts (including expected 0 values)** using Link = Log, Distribution = Poisson/Negative Binomial/Generalized Poisson

- Submodel 2: $\mathrm{Logit}[\mathrm{prob}(y_i = \mathrm{extra}\ 0)] = \beta_{0z} + \beta_{1z}(x_i)$
  - Predict probability of **being an extra 0** using Link = Logit, Distribution = Bernoulli
  - Don't need predictors for this part, can just estimate a fixed intercept (i.e., an empty model) to see if a zero-inflation model is needed at all (see Example 3)

Problem #3: **Too many** 0 values, Option #1

- The predicted outcomes across the zero-Inflated submodels get put back together into one predicted count as follows:
  - ω_i ("omega") is the model-predicted probability of being an extra 0:

$$\omega_i = \frac{\exp[\text{Logit}[\text{prob}(y_i = extra\ 0)]]}{1 + \exp[\text{Logit}[\text{prob}(y_i = extra\ 0)]]}$$

  - μ_i is model-predicted data-scale expected count (including 0 values)
  - ZIP: $\text{Mean (original } y_i) = (1 - \omega_i)\mu_i$
  - ZIP: $\text{Variance (original } y_i) = \mu_i + \frac{\omega_i}{(1-\omega_i)}\mu_i^2$
  - ZINB: $\text{Mean (original } y_i) = (1 - \omega_i)\mu_i$
  - ZINB: $\text{Variance (original } y_i) = \mu_i + \left[\frac{\omega_i}{(1-\omega_i)} + \frac{k}{1-\omega_i}\right]\mu_i^2$

Problem #3: **Too many** 0 values, Option #2

- "**Hurdle**" models for Poisson or Negative Binomial
  - PH or NBH: Explicitly **separates 0 from non-0 values** through two distinct outcome distributions (Bernoulli + Zero-Altered Poisson/NB)
  - Creates two submodels to predict "if any 0" and "if not 0, how much"?
    - Easier to think about in terms of predicting the 0 values (in my opinion)

- Submodel 1: $\text{Log}[\text{E}(y_i)|y_i > 0] = \text{Log}(\mu_i) = \beta_{0c} + \beta_{1c}(x_i)$
  - Predict **only counts > 0** using Link = Log, Distribution = ZAP/ZANB

- Submodel 2: $\text{Logit}[\text{prob}(y_i = \text{any } 0)] = \beta_{0h} + \beta_{1h}(x_i)$
  - Predict **probability of any 0** using Link = Logit, Distribution = Bernoulli

- SAS NLMIXED; STATA CHURDLE, ZTPNM or ZTNB; R VGAM or PSCL; Mplus
  - Can also split the outcome explicitly and estimate each submodel separately, but then you lose the ability for multivariate test of a predictor effect across submodels (which may be an acceptable limitation)

Zero-Inflated Models for Counts

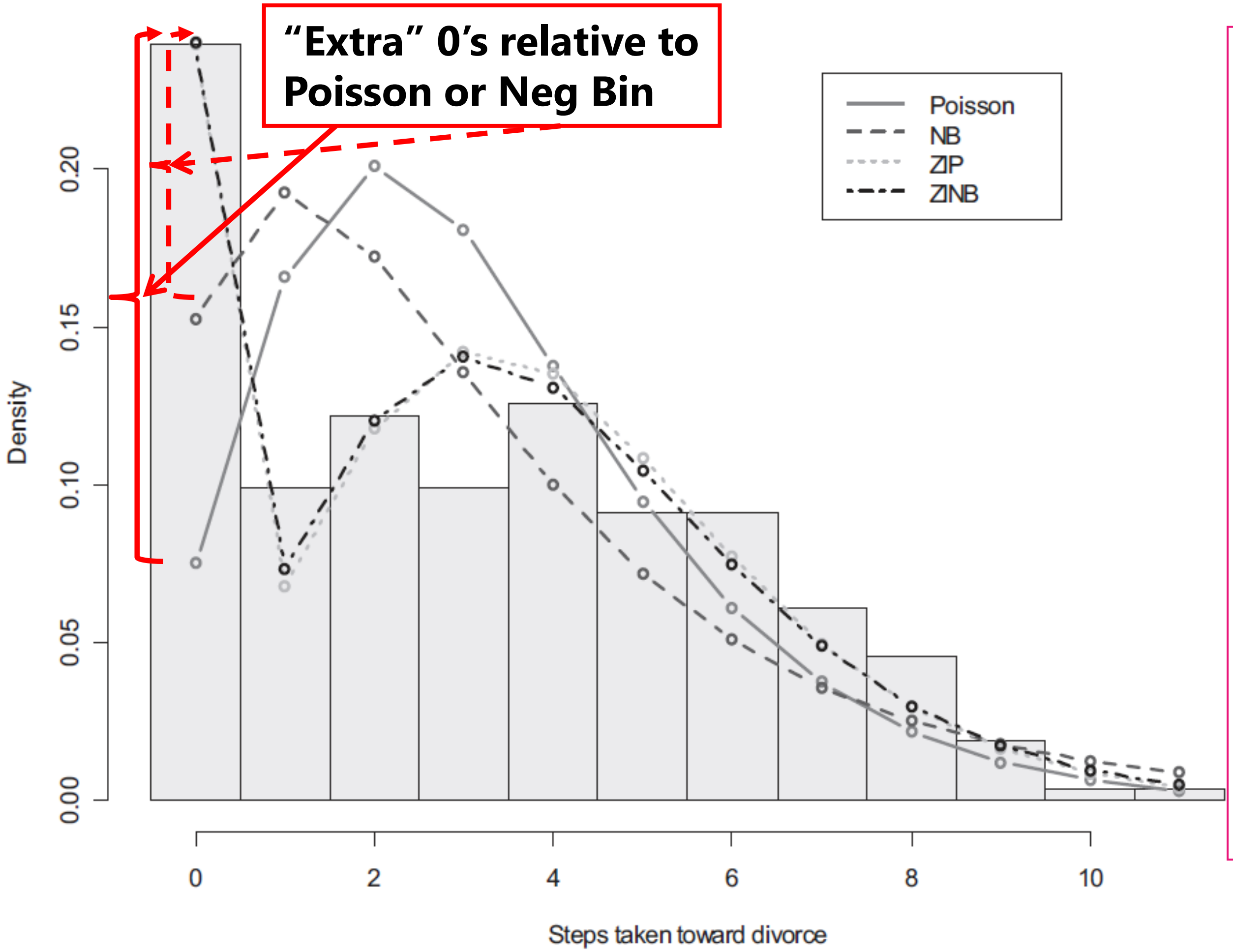


Figure 1. Histogram of Marital Status Inventory with predicted probabilities from regressions. NB = negative binomial; ZIP = zero-inflated Poisson; ZINB = zero-inflated negative binomial.

Zero-inflated distributions come in many forms: Poisson (mean = variance) and Negative Binomial (variance > mean).

An alternative is to think of them as **semi-continuous** in an "**if and how much**" model (my own term for **hurdle models**):

Pred1: =0 if x=0, 1 if x > 0
= Pred1 is binary
Pred2: =how much if x > 0
= Pred2 is quantitative

Image borrowed and doctored from: Atkins, D. C., & Gallop, R. J. (2007). Rethinking how family researchers model infrequent outcomes: A tutorial on count regression and zero-inflated models. *Journal of Family Psychology, 21*(4), 726-735.

More on Comparing Count Models

- Whether or not a dispersion scale parameter is needed (to distinguish Poisson and NB) can be answered via a likelihood ratio test (LRT, or −2ΔLL)
  - For the fairest comparison, keep the linear predictor model the same
  - LRTs should use a χ^2 with a mixture of DF=0 and DF=1 ("chibar" in STATA)

- Whether or not a zero-inflation model is needed should, in theory, also be answerable via a likelihood ratio test… But people disagree about this
  - Problem? Zero-inflation probability can't be negative, so is bounded at 0
  - Solution? Use a χ^2 with a mixture of DF=0 and DF=1 ("chibar" in STATA)
  - Vuong test *had* been used for this, but is currently not recommended
  - Can always compare AIC and BIC (smaller is better)

- In general, models with the same distribution and different links can be compared via AIC and BIC, but one cannot use AIC and BIC to compare across alternative distributions (e.g., normal vs count?)
  - Log-likelihoods are not on the same scale due to using different PDFs
  - Pearson χ^2 / DF provides some guidance as to fit of conditional distribution

Summary: Predicting Counts

- A **count** is a discrete outcome that:
  - Is quantitative (numbers are really numbers)
  - Ranges from 0 (or 1) to positive infinity
    - Don’t have any zeros? Need “zero-truncated/altered” distribution
  - Is predicted using a **log link function** to ensure predicted counts > 0

- Determining the “right” distribution for a count outcome is aided by examining **conditional distribution fit**: Pearson $\chi^2/\mathrm{DF} \approx 1$
  - Counts often have more variance (because of positive skewness) than expected by Poisson (in which mean = variance)—this “over-dispersion” can be fixed by adding a “**scale**” parameter by which variance > mean
  - If you have **more zero values than expected**, may need to add a “zero-inflation” submodel or switch to a “hurdle” two-submodel variant
  - But both dispersion parameters and zero-inflation models are trying to accommodate for skewness, so you may not need both (check fit to see)

SAS for Discrete Outcomes

- There are many choices for modeling not-normal **discrete** outcomes (that include integer values only); most use either an identity or log link
- **Single-level, univariate generalized models by PROC:**
  - GENMOD: DIST= (and default link): Binomial (Logit), Poisson (Log), Zero-Inflated Poisson (Log), Negative Binomial (Log), Zero-Inflated Negative Binomial (Log)
  - FMM: DIST= (and default link): Binomial (Logit), Poisson (Log), Generalized Poisson (Log), Truncated Poisson (Log), Negative Binomial (Log), Uniform
- **Multilevel or multivariate generalized models through GLIMMIX:**
  - Binomial (Logit), Poisson (Log), Negative Binomial (Log)
  - BYOBS, which allows multivariate models by which you specify outcome-specific (or submodel specific) link functions and distributions estimated simultaneously
  - User-defined variance functions for special cases (e.g., generalized Poisson)
  - NLMIXED can also be used to fit any user-defined model

STATA and R for Discrete Outcomes

- **STATA for single-level, univariate generalized models:**
  - glm for multiple options, logit, probit, or cloglog for binary, ologit or oprobit for ordinal, poisson, nbreg, or gnbreg for counts, and many more options
  - Most of these allow cluster-corrected or robust standard errors (stay tuned)
- **R for single-level, univariate generalized models:**
  - VGAM seems to have everything, but it is harder to use than other packages that have fewer options (IMHO), like GLM, MASS, or PSCL

- **STATA for multilevel or multivariate generalized models:**
  - meglm for multiple options, melogit, meprobit, or mecloglog for binary, meologit or meoprobit for ordinal, mepoisson or menbreg for counts
  - menl can also be used to fit any user-defined model (haven't tried that yet)
- **R for multilevel or multivariate generalized models:**
  - glmer (I've used in PSQF 6272 MLM); also nlmer, glmmML (I haven't tried these)