Linear & Generalized Linear Models/ Advanced Regression for Independent Data

Andrew J. Spieker, Ph.D.

Associate Professor of Biostatistics
Vanderbilt University

Set 10: Overdispersion, quasi-likelihood, and optimality

Version: 07/03/2026

Table of Contents

Variance estimators (so far)

Recall: Likelihood-based variance

- We estimate the variance as:

$$\widehat{\text{Cov}}[\widehat{\boldsymbol{\beta}}] = \widehat{\phi}(\mathbb{A}_N(\widehat{\boldsymbol{\beta}}))^{-1},$$

  where $\widehat{\phi} = 1$ if there "is no nuisance" associated with the likelihood, and

$$\widehat{\phi} = \left(\frac{1}{N-K} \sum_{i=1}^{N} \frac{(y_i - \widehat{\mu}_i)^2}{V(\widehat{\mu}_i)} \right).$$

  otherwise.
- $\mathbb{A}_n(\widehat{\boldsymbol{\beta}})$ is proportional to the (expected) Fisher information (for $\boldsymbol{\beta}$).
- These estimators are valid if the likelihood is completely correct.
- We argued previously that these estimators are valid if both the mean model and the mean-variance relationship implied by the GLM are correctly specified—which is to say that the higher-order moments did not come into play.

Variance estimators (so far)

Recall: Sandwich-based variance (version 1)

- The following estimator protects against both mean model and mean-variance misspecification:

$$\widehat{\text{Cov}}[\widehat{\boldsymbol{\beta}}] = (\mathbb{A}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}}))^{-1}(\mathbb{B}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}}))(\mathbb{A}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}}))^{-1}.$$

- If the mean model is incorrect, this estimator will not be valid if $\mathbf{X}$ is fixed (i.e., does not vary by study replicate).

Variance estimators (so far)

Recall: Sandwich-based variance (version 2)

- The following estimator protects against misspecification of the mean-variance relationship:

$$\widehat{\text{Cov}}[\widehat{\boldsymbol{\beta}}] = (\mathbb{A}_N(\widehat{\boldsymbol{\beta}}))^{-1}(\mathbb{B}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}}))(\mathbb{A}_N(\widehat{\boldsymbol{\beta}}))^{-1}.$$

- If the mean model is incorrect, this estimator will not be valid if $\mathbf{X}$ is fixed (i.e., does not vary by study replicate).
- When the canonical link is used, $\mathbb{A}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}})$ and $\mathbb{A}_N(\widehat{\boldsymbol{\beta}})$ are the same; therefore, this estimator would be valid under mean model misspecification under the canonical link.

Variance estimators (so far)

Recall: Bootstrap variance (version 1)

- We discussed a flavor of the bootstrap procedure that re-samples "independent rows" from the original data (with replacement).
- This is consistent with a sampling mechanism in which the design matrix changes from replicate to replicate.
- If the mean model is incorrect and $\mathbf{X}$ is really fixed by design, this will tend to overstate the variance.

Variance estimators (so far)

Recall: Bootstrap variance (version 2)

- We discussed a flavor of the bootstrap procedure that holds the design matrix fixed and re-samples (or re-generates) outcomes only.
- This is consistent with a sampling mechanism in which the design matrix is fixed by design.
- If the mean model is incorrect and $\mathbf{X}$ is really random by design, this will tend to understate the variance.

Variance estimators (so far)

Next step: Quasi-likelihood

- These notes introduce quasi-likelihood; in my mind, it is easier to first discuss a specific application of quasi-likelihood (overdispersion, in particular) and then go through the framework a little more thoroughly.

Table of Contents

Overdispersion

Motivation: Poisson regression

- Imagine we are modeling a count outcome, Y_i, as a function of $\mathbf{x}_i$.
- A natural GLM: $Y_i \sim \text{Poisson}(\mu_i = \exp(\mathbf{x}_i^\intercal \boldsymbol{\beta}))$.
- Here is the salient information implied by the GLM:
  - Mean model: $\log(\text{E}[Y|\mathbf{x}]) = \mathbf{x}^\intercal \boldsymbol{\beta}$.
  - Mean-variance relationship: $\text{Var}[\mathbf{y}] = \boldsymbol{\mu}$.
- The Poisson family is a single-parameter family; it does not have a nuisance parameter. The mean-variance relationship described above is one of strict *equality* if you believe the full likelihood.
- What if I believe the mean model to be correct, but *not* the mean-variance relationship?
  - Any of the many variance estimators we have discussed so far should be valid *except* the likelihood-based one. Why?

Overdispersion

Example 10.1: Overdispersed/underdispersed Poisson

- What if I believe the mean model, and that $\text{Var}[Y_i|\mathbf{x}_i] = \varphi\mu_i$?
- We refer to φ as the *dispersion* parameter:
  - If $0 < \varphi < 1$, we say there is underdispersion.
  - If $1 < \varphi < \infty$, we say there is overdispersion.
- Remember: no nuisance parameter in the Poisson family. Believing the mean-variance relationship implied by the Poisson family means you believe that $\text{Var}[Y_i|\mathbf{x}_i] = \mu_i$ exactly.
- Believing the mean-variance relationship "up to a fixed, unknown proportionality constant" effectively introduces (enforces) a nuisance where it doesn't exist in the single-parameter likelihood.
  - This is, as we will see, an application of a broader framework referred to as quasi-likelihood.
- Appropriate variance estimator in this case:

$$\widehat{\text{Cov}}[\widehat{\boldsymbol{\beta}}] = \widehat{\varphi}(\mathbb{A}_N(\widehat{\boldsymbol{\beta}}))^{-1} = \left(\frac{1}{N-K}\sum_{i=1}^{N}\frac{(y_i-\widehat{\mu}_i)^2}{V(\widehat{\mu}_i)}\right)(\mathbb{A}_N(\widehat{\boldsymbol{\beta}}))^{-1}.$$

Example 10.1: Overdispersed/underdispersed Poisson

- I leave it to you to verify the following estimating equations for Poisson regression (log-link):

$$\mathbf{X}^\mathsf{T}(\mathbf{y} - \text{vec}(\exp(\mathbf{x}_i^\mathsf{T}\boldsymbol{\beta}))) = \mathbf{0}.$$

- I leave it to you to verify the following components of the sandwich:

$$\begin{aligned} \mathbb{A}_N(\widehat{\boldsymbol{\beta}}) &= \mathbf{X}^\mathsf{T}\text{diag}(\exp(\mathbf{x}_i^\mathsf{T}\widehat{\boldsymbol{\beta}}))\mathbf{X} \qquad (= \mathbb{A}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}})\ldots). \\ \mathbb{B}_N^{\text{obs}}(\widehat{\boldsymbol{\beta}}) &= \mathbf{X}^\mathsf{T}\text{diag}(y_i - \exp(\mathbf{x}_i^\mathsf{T}\widehat{\boldsymbol{\beta}}))^2\mathbf{X}. \end{aligned}$$

- I leave it to you to verify the estimator of the dispersion parameter:

$$\widehat{\varphi} = \frac{1}{N-K}\sum_{i=1}^{N}\frac{(y_i - \exp(\mathbf{x}_i^\mathsf{T}\widehat{\boldsymbol{\beta}}))^2}{\exp(\mathbf{x}_i^\mathsf{T}\widehat{\boldsymbol{\beta}})}.$$

Example 10.1: Overdispersed Poisson

- Consider generating $Y = 2Y^*$, where $Y^* \sim \text{Poisson}(\exp(\mathbf{x}^\top \boldsymbol{\beta}^*))$.
- Then, $\text{E}[Y|\mathbf{x}] = \exp(\mathbf{x}^\top \boldsymbol{\beta})$, where $\boldsymbol{\beta} = (\beta_0^* + \log(2), \beta_1^*, \ldots, \beta_{K-1}^*)$.
  - The Poisson mean model is correctly specified.
- Further, $\text{Var}[Y|\mathbf{x}] = 4\text{Var}[Y^*|\mathbf{x}] = 2\text{E}[Y|\mathbf{x}] \Rightarrow \varphi = 2$. The Poisson mean-variance relationship is not correctly specified, but it *is* correct up to a proportionality constant that can be estimated and taken into account.

Example 10.1: Overdispersed Poisson

```
## Set seed for reproducibility
set.seed(7345)

## Set "true" value of beta
beta <- matrix(c(1, 1), nrow = 2)

## Generate data
n <- 40
X <- matrix(cbind(1, runif(n, 0, 2)), ncol = 2)
y <- 2*rpois(n, exp(X %*% beta))

## The true value of beta is really (1 + log(2), 1).
```

Overdispersion

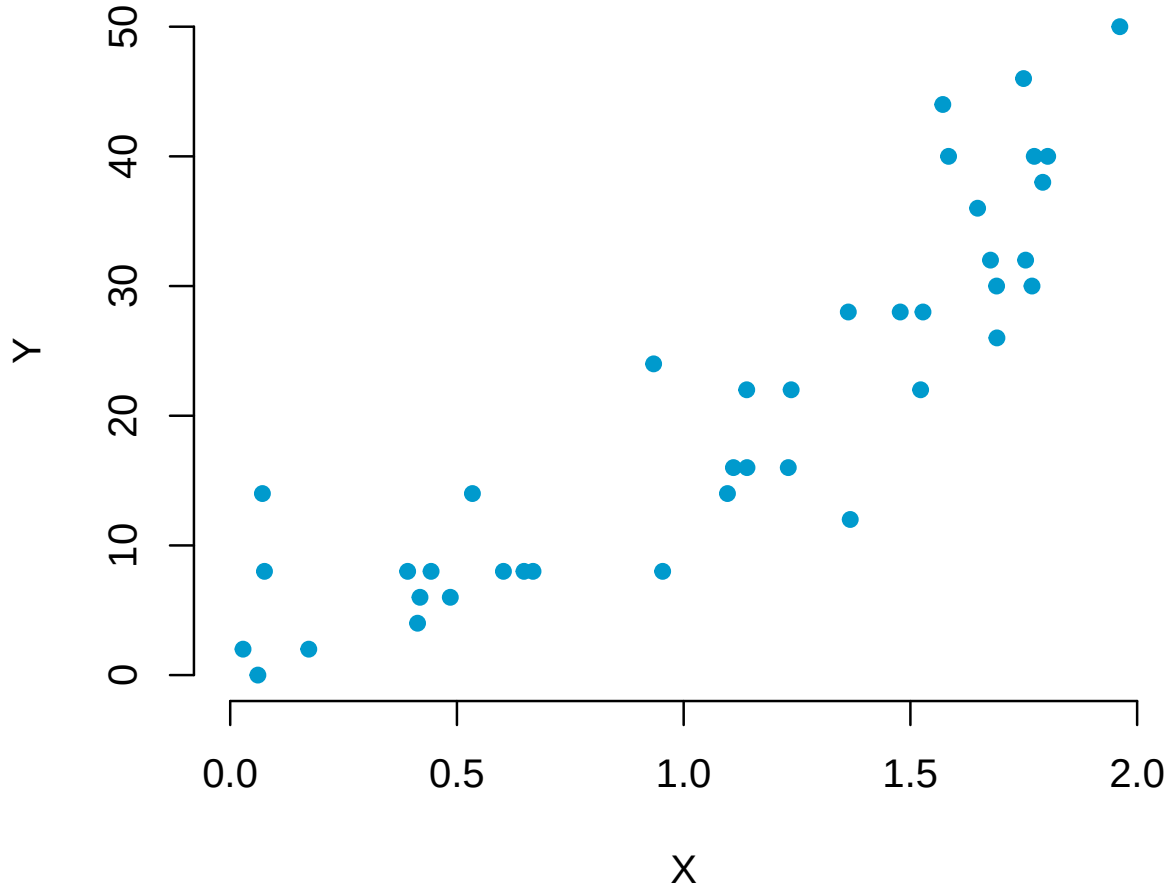


Figure 10.1: Overdispersed Poisson.

Example 10.1: Overdispersed Poisson

- I leave it to you to verify that $\widehat{\beta}_1 = 1.22$.
- I leave it to you to verify that $\widehat{\varphi} = 1.94$.
- I leave it to you to verify the following standard errors:
  1. Likelihood-based (invalid): $\widehat{\text{SE}}[\widehat{\beta}_1] = 0.0764$.
  2. Sandwich-based (valid): $\widehat{\text{SE}}[\widehat{\beta}_1] = 0.112$.
  3. Non-parametric bootstrap (valid): $\widehat{\text{SE}}[\widehat{\beta}_1] \approx 0.115$.
  4. Quasi-Poisson (valid): $\widehat{\text{SE}}[\widehat{\beta}_1] = 0.106$.
- Once you know how to program a GLM in R, going the extra step to estimate the dispersion parameter is straightforward.
- In R, you can use the following family specification to account for overdispersion (or underdispersion):

```
glm(..., family = quasipoisson(link = "log"))
```

Overdispersion

Example 10.1: Overdispersed Poisson

- With the previous example, we just motivated (and provided a *very* common application of) quasi-likelihood theory.
- We previously made the argument that the score equations for GLMs are uniquely determined by the first and second moments of the corresponding likelihood; even then, there is typically an asymptotically valid estimation procedure for the variance even if neither moment is correct (there are many considerations that go into selecting the right variance estimator).
- We'll make this argument again within the quasi-likelihood framework, and discuss the implications in a little more detail.

Table of Contents

Quasi-score:

- Consider the fact that the quantity

$$U(\mu; Y) = \frac{Y - \mu}{\varphi V(\mu)}$$

  behaves in many ways like a score function.
- The function $V(\cdot)$ specifies the mean-variance relationship.
- I leave it to you to verify that:
  - $\mathrm{E}[U] = 0$.
  - $\mathrm{Var}[U] = (\varphi V(\mu))^{-1}$.
  - $-\mathrm{E}[\partial U/\partial \mu] = (\varphi V(\mu))^{-1}$.

Quasi-(log)likelihood:

- Therefore, it stands to reason that the quantity

$$\sum_{i=1}^{N} Q(\mu_i, y_i) = \sum_{i=1}^{n} \int_{y_i}^{\mu_i} \frac{y_i - t}{\varphi V_i(t)} dt,$$

  when it exists, behaves much like a log-likelihood function.
- If it does exist, its derivative (with respect to $\boldsymbol{\beta}$) takes the form of the estimating equations we have been considering:

$$\mathbf{D}^\top \mathbf{V}^{-1}(\mathbf{y} - \boldsymbol{\mu})/\varphi = \mathbf{0}.$$

Quasi-(log)likelihood:

- The quasi-(log)likelihood looks something like the log-likelihood of the exponential families we've been discussing:

$$\sum_{i=1}^{N} Q(\mu_i, y_i) = \sum_{i=1}^{n} \int_{y_i}^{\mu_i} \frac{y_i - t}{\varphi V_i(t)} dt.$$

- The specific form of the quasi-(log)likelihood (as a function of $\boldsymbol{\mu}$) is determined by specifying a function, $V(\cdot)$, that corresponds to a mean-variance relationship.
- There are certain things we might expect if we were to choose specific forms of $V(\cdot)$. Let's try a couple!

Quasi-(log)likelihood: Normal

- If we choose $V(\mu) = 1$, we expect the quasi-(log)likelihood to resemble the log-likelihood of a normal distribution.

$$\begin{aligned} Q(\mu, y) &= \int_y^\mu \frac{y-t}{\varphi} dt \\ &= \frac{1}{\varphi} \int_y^\mu (y-t)\, dt \\ &= -\frac{1}{2\varphi}(y-\mu)^2. \end{aligned}$$

- Important notes:
  - This contains only the salient information from the normal likelihood.
  - The dispersion parameter, φ, serves the role of σ^2.

Quasi-(log)likelihood: Poisson

- If we choose $V(\mu) = \mu$, we expect the quasi-(log)likelihood to resemble the log-likelihood of a Poisson distribution.

$$\begin{aligned} Q(\mu, y) &= \int_y^\mu \frac{y-t}{\varphi t} dt \\ &= \frac{1}{\varphi} \int_y^\mu \left(y t^{-1} - 1\right) dt \\ &= \frac{1}{\varphi} \left(y \log(\mu) - \mu - y(1 + \log(y))\right). \end{aligned}$$

- Important notes:
  - Restrictions: $\mu > 0$; $y > 0$.
  - $c(y) = -y(1 + \log(y))$ is constant with respect to μ and is irrelevant in deriving the quasi-score for $\boldsymbol{\beta}$.
  - The Poisson distribution does not have a nuisance parameter; this framework introduces overdispersion and underdispersion to a model in which we believe the mean and variance are directly proportional.

Quasi-(log)likelihood: Bernoulli

- If we choose $V(\mu) = \mu(1-\mu)$, we expect the quasi-(log)likelihood to resemble the log-likelihood of a Bernoulli distribution.

$$\begin{aligned} Q(\mu, y) &= \int_y^\mu \frac{y-t}{\varphi t(1-t)} dt \\ &= \ldots \text{Informal homework!} \end{aligned}$$

- Important notes:
  - Be mindful of restrictions on μ and y.
  - If Bernoulli observations are independent, it is theoretically impossible to have underdispersion or overdispersion under a correct mean model.
  - If there *is* underdispersion or overdispersion in this case, it must originate from correlation between the observations (sneak preview of longitudinal material).

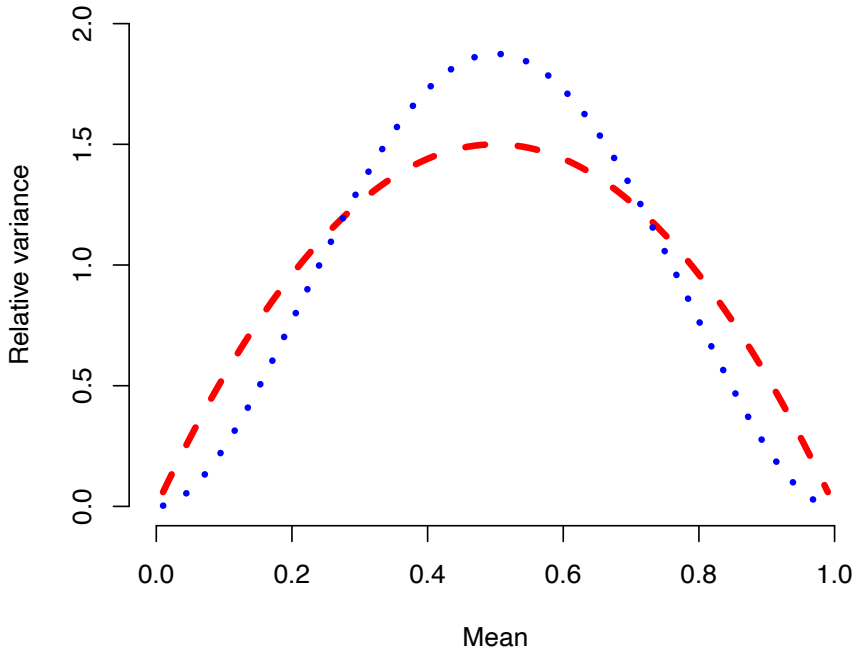
Quasi-(log)likelihood: ???

- If we choose $V(\mu) = \mu^2(1-\mu)^2$, we expect the quasi-(log)likelihood to resemble the log-likelihood of a... ???

$$Q(\mu, y) = \int_y^\mu \frac{y-t}{\varphi t^2(1-t)^2} dt.$$

- Important notes:
  - This choice of $V(\mu)$ does not originate from a probability function.
  - "Quasi-likelihood" is so termed because the resulting estimating equations need not originate from an actual likelihood.
- What might be a motivation to use this choice of a variance function for an outcome bounded between zero and one?

Quasi-likelihood



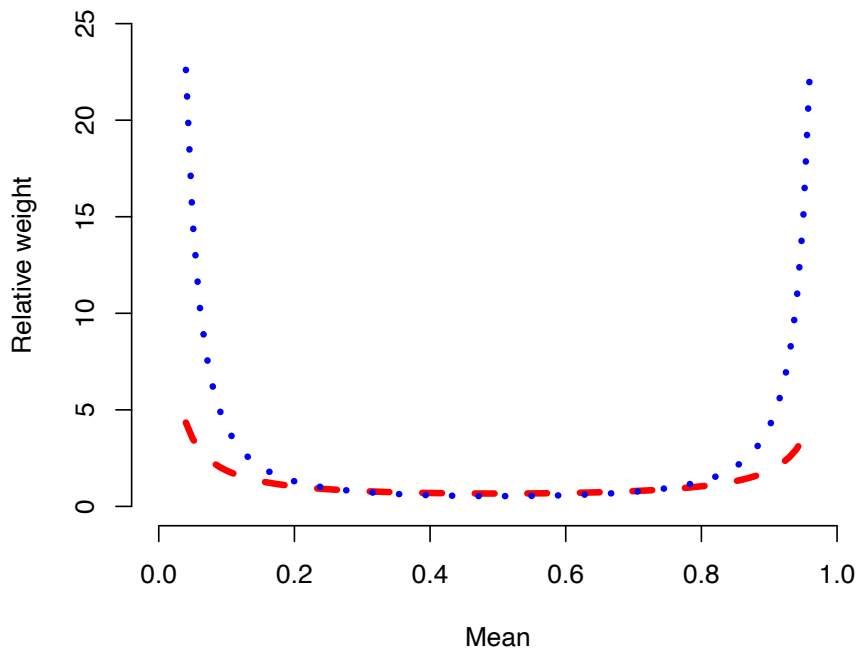


Figure 10.2: Implied weights ($V_1(\mu) \propto \mu(1-\mu)$ vs. $V_2(\mu) \propto \mu^2(1-\mu)^2$).

Quasi-likelihood

GLMs with a nuisance parameter: Really just quasi-likelihood

- Thinking about GLMs under the quasi-likelihood framework involves the following steps:
  1. Specify a mean model, $g(\mu) = \mathbf{x}^\mathsf{T}\boldsymbol{\beta}$.
  2. Specify a weighting scheme $W(\mu)$ (often by inverting a mean-variance relationship, $V(\mu)$—this comes with advantages that I've been hinting toward and which we will discuss shortly).
  3. Select an appropriate variance estimation procedure based on the following considerations (some of which interact with one another).
     - ⋆ Do you believe the mean model and/or mean-variance relationship?
     - ⋆ Might there by overdispersion or underdispersion?
     - ⋆ Is the link function "canonical" given the mean-variance relationship?
     - ⋆ Is $\mathbf{X}$ fixed by design or random by design?
     - ⋆ What is your sample size? What is your computing power?
- This gives you some flexibility. For instance, when thinking within this framework, you don't need to have count data to justify fitting "Poisson regression," if you believe the mean model and mean-variance relationship *implied* by a Poisson regression model up to a proportionality constant that can be estimated.

Another application: Robust estimation

- Quasi-likelihood has other applications that are outside the scope of this course.
- One other popular application is to robust estimation, in which you select the weights of a model in a way that *bounds* the influence of an observation (the influence of an observation grows without bound in a regression model when they are equally weighted).

Table of Contents

Negative binomial regression

Poisson regression: Variance choices

- Let's pivot to another way to deal with overdispersion (overdispersed count data, in this case).
- As we know, count data are often analyzed using Poisson regression.
  - If we believe the mean model and that $\text{Var}[Y_i|\mathbf{x}_i] = \text{E}[Y_i|\mathbf{x}_i]$, we should be justified in the likelihood-based approach.
  - If we believe the mean model and that $\text{Var}[Y_i|\mathbf{x}_i] = \varphi\text{E}[Y_i|\mathbf{x}_i]$ for some unknown φ, we can use quasi-Poisson regression.
  - If we believe the mean model but not that $\text{Var}[Y_i|\mathbf{x}_i] = \varphi\text{E}[Y_i|\mathbf{x}_i]$, we can employ bootstrap or sandwich-based techniques.
- Even the condition $\text{Var}[Y_i|\mathbf{x}_i] = \varphi\text{E}[Y_i|\mathbf{x}_i]$ is not maximally flexible. What if I wanted to impose a different structure on the mean variance relationship, such as $\text{Var}[Y_i|\mathbf{x}_i] = \text{E}[Y_i|\mathbf{x}_i] + \text{E}[Y_i|\mathbf{x}_i]^2/\varphi$ for an unknown φ.
- This doesn't quite fit into our quasi-likelihood framework. Why?

Negative binomial regression

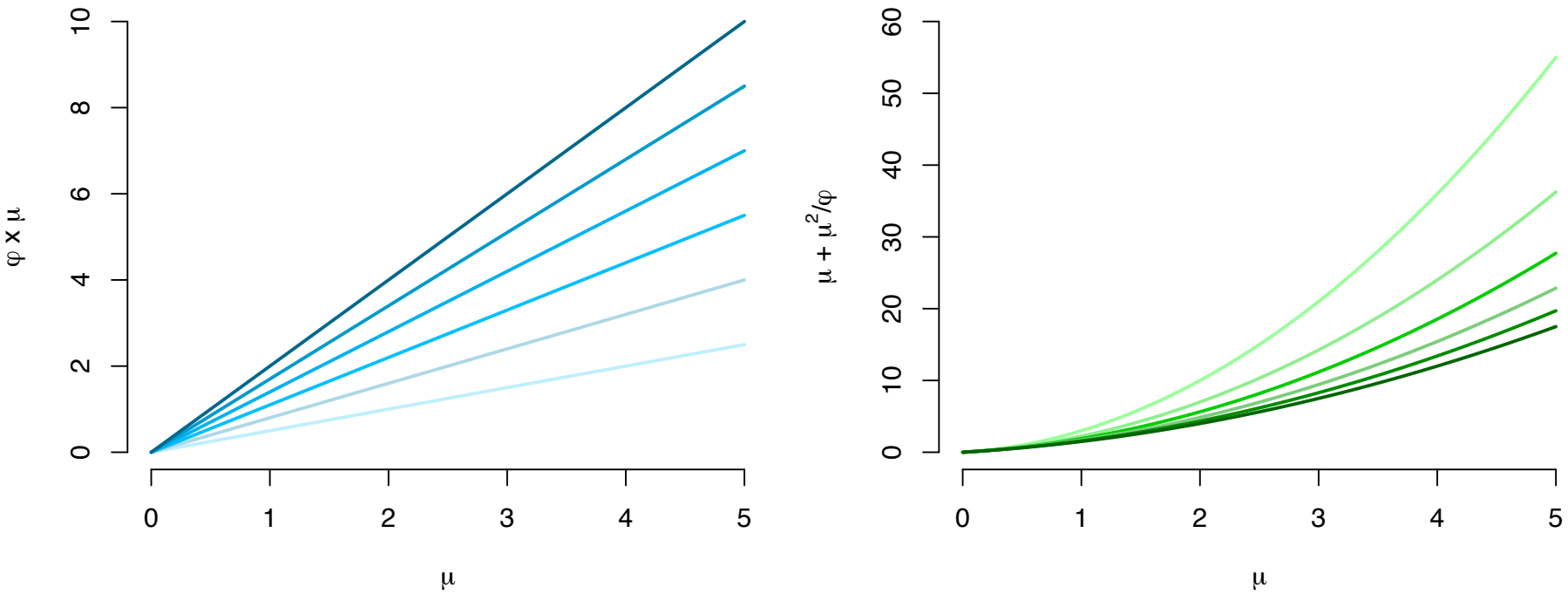


Figure 10.3: Modeling overdispersion (different choices of φ).

Poisson regression: Variance choices

- Both the parametric and the quasi-likelihood formulation of a GLM result in estimating equations that take the following form:

$$\begin{aligned} \mathbf{D}^\top(\boldsymbol{\beta})\mathbf{V}^{-1}(\boldsymbol{\beta})(\mathbf{y}-\boldsymbol{\mu}(\boldsymbol{\beta}))/\phi &= \mathbf{0} \quad \text{(likelihood);} \\ \mathbf{D}^\top(\boldsymbol{\beta})\mathbf{V}^{-1}(\boldsymbol{\beta})(\mathbf{y}-\boldsymbol{\mu}(\boldsymbol{\beta}))/\varphi &= \mathbf{0} \quad \text{(quasi-likelihood).} \end{aligned}$$

- The value of the nuisance/dispersion parameter does not have any influence on the solution to the estimating equations $\widehat{\boldsymbol{\beta}}$. Why?
  - For the special exponential family we've been considering for GLMs, recall that $\text{Var}[Y_i|\mathbf{x}_i] = \phi b''(\theta_i)$ is linear in the nuisance parameter, ϕ.
  - For quasi-likelihood, we assume that $\text{Var}[Y_i|\mathbf{x}_i] = \varphi V(\mu_i)$ is linear in the dispersion parameter, φ.
- If we assume $\text{Var}[Y_i|\mathbf{x}_i] = \mu_i + \mu_i^2/\varphi$, then the estimating equations take the form: $\mathbf{D}^\top(\boldsymbol{\beta})\mathbf{V}^{-1}(\boldsymbol{\beta},\varphi)(\mathbf{y}-\boldsymbol{\mu}(\boldsymbol{\beta})) = \mathbf{0}$, whereby the solution to the estimating equations, $\widehat{\boldsymbol{\beta}}$, *does* depend upon φ.

Negative binomial regression

Variance function: $\text{Var}[Y_i|\mathbf{x}_i] = \mu_i + \mu_i^2/\varphi$

- Natural question: what are the origins of this variance function?
- I confess: I answer this begrudgingly because the answer is a parametric model, in part undermining the point I'm trying to make in this set of notes, which is that we don't need a likelihood to motivate a suitable estimating equation.
- Nevertheless, context can be helpful, so here's the scoop.
- Suppose $Z \sim \text{Gamma}(\varphi, \varphi)$ is a latent variable (so that $\text{E}[Z] = 1$ and $\text{Var}[Z] = 1/\varphi$), and let $Y|Z \sim \text{Poisson}(\mu Z)$. It is straightforward to show that the *marginal* probability function for Y is given by:

$$p_Y(y; \mu, \varphi) = \frac{\Gamma(\varphi + y)}{\Gamma(\varphi) y!} \frac{\varphi^\varphi \mu^y}{(\mu + \varphi)^{\varphi + y}},$$

such that $Y \sim \text{NegativeBinomial}(k = \varphi, p = \mu/(\mu + \varphi))$.

Negative binomial regression

Negative binomial distribution:

- Let's investigate $Y \sim \text{NegativeBinomial}(k = \varphi, p = \mu/(\mu + \varphi))$, which originates from counting the number of failures in a sequence of i.i.d. Bernoulli trials until the first φ successes are achieved.
  - We're not using this parametric family because of its interpretation in this way, but because of the implied mean-variance relationship that we seek to derive.
- If φ is known, this takes the exponential form appropriate for a GLM.

$$p_Y(y; \mu, \varphi) = \exp\left(y \log\left(\frac{\mu}{\mu + \varphi} \right) - \varphi \log(\mu + \varphi) + c(y, \varphi) \right).$$

- If φ is unknown, this is not suitable for the GLM framework as we have discussed it. Let's start by treating φ as known.

Negative binomial regression

Negative binomial distribution:

- Mass function:

$$p_Y(y;\mu,\varphi) = \exp\left(y\log\left(\frac{\mu}{\mu+\varphi}\right) - \varphi\log(\mu+\varphi) + c(y,\varphi)\right).$$

- I leave it to you to show the following:
  - $\theta = \log(\mu/(\mu+\varphi))$ (canonical parameter).
  - $b(\theta) = \varphi\log(\varphi/(1-\exp(\theta)))$.
  - $b'(\theta) = \varphi\exp(\theta)/(1-\exp(\theta)) = \mu$.
  - $V(\mu) = \mu + \mu^2/\varphi$ (this is the punchline).
- We rarely use the canonical link for negative binomial regression.
- Rather, this model is a convenient way of modeling count data in which there is overdispersion with respect to the Poisson model.
- Note that the negative binomial distribution resembles a Poisson distribution as φ grows (reflected by the mean-variance relationship).

Negative binomial regression

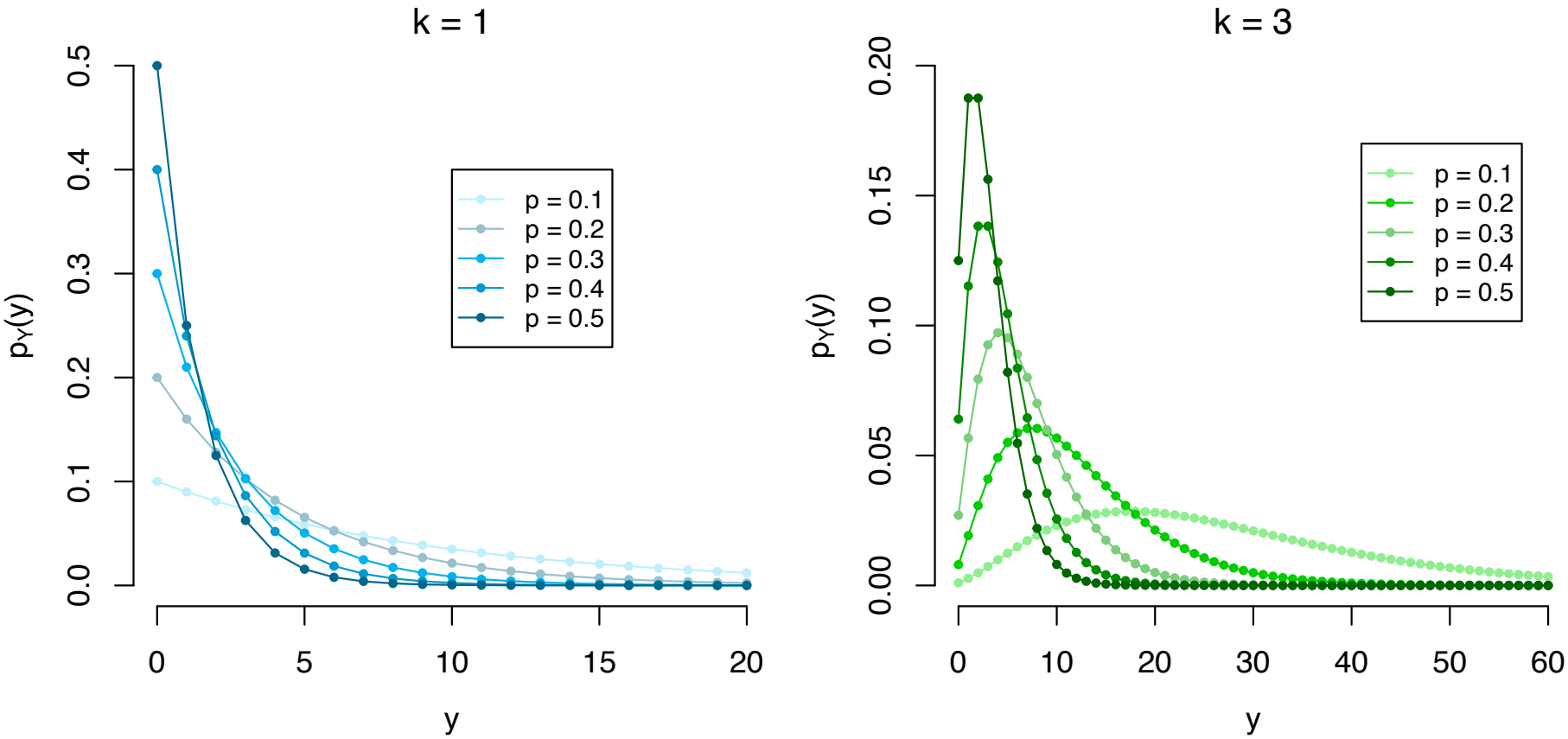


Figure 10.4: The negative binomial distribution.

Negative binomial regression

Negative binomial regression: Unknown φ

- If φ is known, estimation of $\boldsymbol{\beta}$ is straightforward under the likelihood framework or the framework of estimating equations.
- The estimating equations for $\boldsymbol{\beta}$ unavoidably depend upon φ, but we can estimate it iteratively with $\boldsymbol{\beta}$.
- I leave it to you to verify the following estimating equations for negative binomial regression under the log link:

$$\mathbf{X}^{\mathsf{T}}\text{diag}\left(\frac{\exp(\mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta})}{\exp(\mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta}) + \exp(\mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta})^2/\varphi}\right)(\mathbf{y} - \text{vec}(\exp(\mathbf{x}_i^{\mathsf{T}}\boldsymbol{\beta}))) = \mathbf{0}.$$

- I also leave it to you to verify the following method-of-moments estimator for φ given $\widehat{\boldsymbol{\beta}}$:

$$\widehat{\varphi} = \left[\frac{1}{N-K}\sum_{i=1}^{N}\frac{(y_i - \exp(\mathbf{x}_i^{\mathsf{T}}\widehat{\boldsymbol{\beta}}))^2 - \exp(\mathbf{x}_i^{\mathsf{T}}\widehat{\boldsymbol{\beta}}))}{\exp(\mathbf{x}_i^{\mathsf{T}}\widehat{\boldsymbol{\beta}})^2}\right]^{-1}$$

Negative binomial regression

Negative binomial regression: Unknown φ

- This suggests iterating estimation of $\boldsymbol{\beta}$ and φ using, e.g., the Gauss-Newton algorithm. There are technical/theoretical details surrounding why you can do this; I am putting those aside for the sake of time.
- The function `glm.nb()` in R (from the `MASS` library) will iterate estimation of $\boldsymbol{\beta}$ but will not iterate estimation of φ; it will initialize $\boldsymbol{\beta}$ from a Poisson fit, and φ as fixed following a single step.
  - The single-step method *only* works if it is based on a consistent estimator, $\widehat{\boldsymbol{\beta}}$.
  - Theory surrounding "one-step" estimators can be used to justify this.
  - Absent a consistent initializer, we *must* iterate estimation of φ until convergence (alongside $\boldsymbol{\beta}$).

Example 10.2: Negative binomial regression

```r
## Set seed for reproducibility
set.seed(7345)

## Set sample size
n <- 60

## Set true parameters
beta <- c(2, 1)
phi <- 3

## Generate data
X <- cbind(1, runif(n, 1, 5))
prob <- phi/(phi + exp(X %*% beta))
y <- rnbinom(n, size = phi, prob = prob)

## Initialize beta and phij
betaj <- as.numeric(coef(glm(y ~ X[,2], family = poisson(link = "log"))))
muj <- c(exp(X %*% betaj))
phij <- ((1/(n - 2))*sum(((y - muj)^2 - muj)/(muj^2)))^(-1)

## Set tolerance and count iterations
tol <- 1
iter <- 1
```

Negative binomial regression

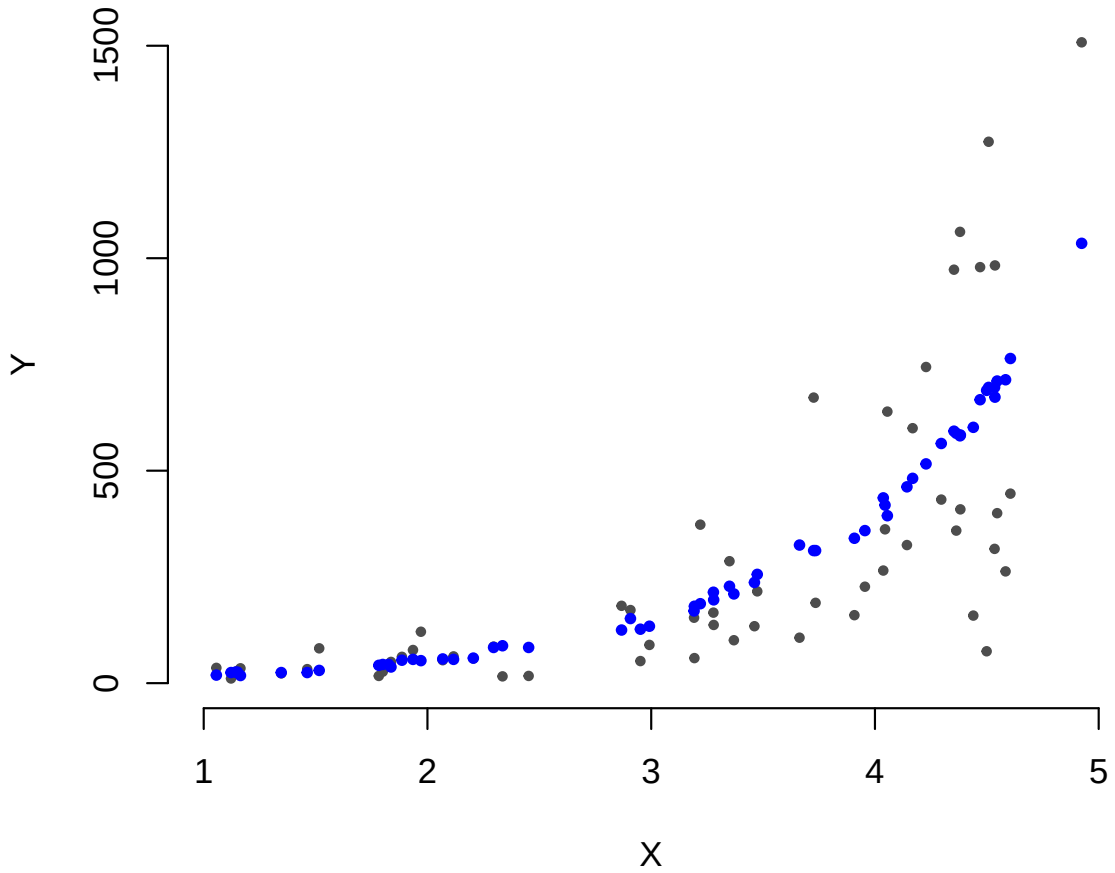


Figure 10.5: Negative binomial with to Poisson (`rpois()`).

Example 10.2: Negative binomial regression (implementation)

```
while(tol > 1e-12 & iter < 50)
{
  ## Store prior iteration
  betaj.prior <- betaj

  ## Mean and mean-variance
  muj <- c(exp(X %*% betaj))
  V <- c(muj + muj^2/phij)

  ## Estimating function
  Gn <- t(X * muj * (1/V)) %*% (y - muj)

  ## An
  An <- t(X * (muj^2) * (1/V)) %*% X

  ## Update to next iteration
  betaj <- betaj + solve(An) %*% Gn

  ## Nuisance
  phij <- ((1/(n - 2))*sum(((y - muj)^2 - muj)/(muj^2)))^(-1)

  ## Determine if conditions are met
  tol <- sum((betaj - betaj.prior)^2)
  iter <- iter + 1
}
```

Example 10.2: Negative binomial regression (results)

```
> iter
[1] 7

> tol
[1] 1.239706e-13

## Estimate
> betaj
            [,1]
[1,] 2.1642150
[2,] 0.9429575

## Standard errors
muj <- c(exp(X %*% betaj))
phij <- ((1/(n - 2))*sum(((y - muj)^2 - muj)/(muj^2)))^(-1)
V <- c(muj + muj^2/phij)
An <- t(X * (muj^2) * (1/V)) %*% X

## Estimated SEs from negative binomial model (valid)
> sqrt(diag(solve(An)))
[1] 0.21660667 0.06387519
```

Example 10.2: Negative binomial (compare to Posson)

```
zz2 <- glm(y ~ X[,2], family = poisson(link = "log"))

## Estimate (consistent)
> summary(zz2)$coef[,1]
(Intercept)        X[, 2]
   1.892569    1.015723

## Model-based standard errors (invalid!)
> sqrt(diag(vcov(zz2)))
(Intercept)        X[, 2]
 0.04549637  0.01097969

## Sandwich standard errors (valid!)
> sqrt(diag(sandwich(zz2)))
(Intercept)        X[, 2]
  0.3621749   0.1013718
```

Example 10.2: Negative binomial (compare to quasi-Poisson)

```
zz3 <- glm(y ~ X[,2], family = quasipoisson(link = "log"))

## Estimate (consistent; same as Poisson GLM)
 > summary(zz3)$coef[,1]
(Intercept)        X[, 2]
   1.892569    1.015723

## Dispersion parameter (not consistent for phi; different model!)
> summary(zz3)$dispersion
[1] 98.56069

## Quasi-Poisson (invalid; "better than model-based.")
> sqrt(diag(vcov(zz3)))
(Intercept)        X[, 2]
  0.4516777   0.1090039

## ...But keep in mind that this is based on one replicate only...
```

Table of Contents

Optimality

Linear unbiased estimating equations:

- Although negative binomial regression does not quite fall under the umbrella of quasi-likelihood, it *does* fall under the unifying framework of linear unbiased estimating equations. Let's tie this up with a discussion of optimality.
- Consider the mean model $g(\mu) = \mathbf{x}^\top \boldsymbol{\beta}$ (assume correctly specified), and consider the following equations to estimate $\boldsymbol{\beta}$:

$$\mathbb{G}_{N,\mathbf{W}}(\boldsymbol{\beta}) = \mathbf{D}^\top \mathbf{W}(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0},$$

  where $\mathbf{W}$ is a diagonal weight matrix that may depend upon $\boldsymbol{\beta}$.
- These are all linear unbiased estimating equations.
- Question: What choice of $\mathbf{W}$ is "optimal" and in what sense?

Optimality

Linear unbiased estimating equations:

- The estimating equations $\mathbb{G}_{N,\mathbf{W}}(\boldsymbol{\beta}) = \mathbf{D}^\top \mathbf{W}(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0}$, produce solutions $\widehat{\boldsymbol{\beta}}_{\mathbf{W}}$.
  - Let us restrict our attention to "regular" choices of $\mathbf{W}$.
- Key point: Asymptotically, $\text{Var}[\mathbf{c}^\top \widehat{\boldsymbol{\beta}}_{\mathbf{V}^{-1}}] \leq \text{Var}[\mathbf{c}^\top \widehat{\boldsymbol{\beta}}_{\mathbf{W}}]$.
- This is a Gauss-Markov style theorem as for linear unbiased estimating equations (including GLMs). You've already seen several variants of this theorem for linear regression.
  - The key to proving this is to show that, at least asymptotically, $\text{Cov}[\widehat{\boldsymbol{\beta}}_{\mathbf{W}}] \succ \text{Cov}[\widehat{\boldsymbol{\beta}}_{\mathbf{V}^{-1}}]$ when $\mathbf{W} \not\propto \mathbf{V}^{-1}$.
  - For OLS and WLS with known weights, this held in finite samples.

Example 10.3: Simulation study

- Consider the following data generating mechanism:
  - $X \sim \text{Uniform}(0, 10)$.
  - $Y \sim \text{Poisson}(\lambda = 10 + 3X)$.
- Now, consider the following GLMs with identity link (conceptualized in the framework of linear unbiased estimating equations):
  1. Gaussian (that is, $\mathbf{W} = \mathbf{I}$).
  2. Inverse-Gaussian (that is, $\mathbf{W} = \text{diag}(1/\mu_i^3)$).
  3. Gamma (that is, $\mathbf{W} = \text{diag}(1/\mu_i^2)$).
  4. Poisson (that is, $\mathbf{W} = \text{diag}(1/\mu_i)$).
- Based on the theory, we expect the Poisson-based estimating equations to provide the most efficient solutions among these four. Let's try a simulation with $n = 100$.

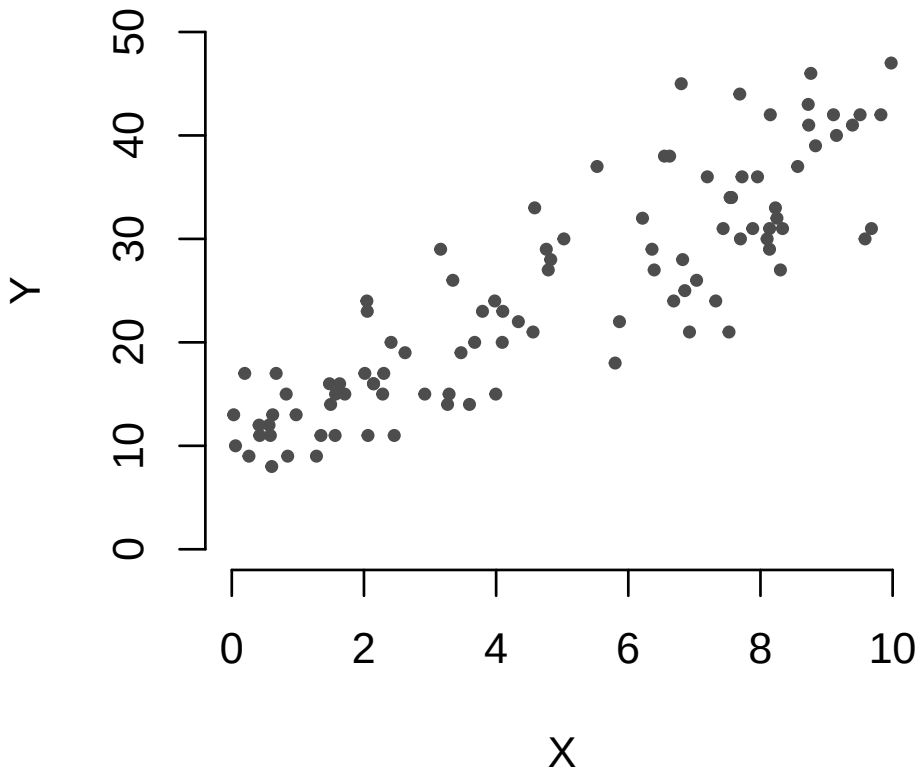


Figure 10.6: Example data set.

Example 10.3: Simulation study

```
## Set seed for reproducibility
set.seed(7345)

## Sample size
n <- 100

## Number of simulations
nsim <- 5000

## True value of beta
beta0 <- 10
beta1 <- 3

## Store results
res <- matrix(0, nrow = nsim, ncol = 4)
```

Example 10.3: Simulation study

```
## Run simulation
for (j in 1:nsim)
{
  ## Generate data
  x <- runif(n, 0, 10)
  y <- rpois(n, lambda = beta0 + beta1 * x)

  ## OLS (incorrect mean-variance)
  zz1 <- glm(y ~ x, family = gaussian(link = "identity"))

  ## Inverse-Gaussian (incorrect mean-variance)
  zz2 <- glm(y ~ x, family = inverse.gaussian(link = "identity"), start = coef(zz1))

  ## Gamma (incorrect mean-variance)
  zz3 <- glm(y ~ x, family = Gamma(link = "identity"), start = coef(zz1))

  ## Poisson with identity link (correct mean-variance)
  zz4 <- glm(y ~ x, family = poisson(link = "identity"), start = coef(zz1))
  res[j,] <- c(coef(zz1)[2], coef(zz2)[2], coef(zz3)[2], coef(zz4)[2])
}

## Results square with the theory!
> apply(res, 2, mean)
[1] 3.00484 3.01779 3.00778 3.00447

> apply(res, 2, sd)
[1] 0.172560 0.214435 0.175121 0.163660
```

Example 10.4: Justifying a GLM

- Suppose $Y_i \sim \text{Gamma}(\alpha_i = \eta_i^2, \beta_i = \eta_i)$.
- It is readily shown that $\text{E}[Y_i] = \eta_i$ and $\text{Var}[Y_i] = 1$.
- Under this data generating mechanism, the theory says that the Gaussian family with identity link gives rise to linear unbiased estimating equations that are asymptotically efficient (for $\boldsymbol{\beta}$) within that subclass of estimators.
- One could, in this bizarre example, treat $\sigma^2 = 1$ as fixed and known to squeeze out some efficiency in estimation of $\text{Cov}[\widehat{\boldsymbol{\beta}}]$.
  - This example is purely didactic.

Optimality

Example 10.4: Justifying a GLM

- $Y_i \sim \text{Gamma}(\alpha_i = \eta_i^2, \beta_i = \eta_i)$.
- As just stated GLM's optimality is restricted to the very specific class of estimating equations given by:

$$\mathbb{G}_{N,\mathbf{W}}(\beta) = \mathbf{X}^\intercal \mathbf{W}(\mathbf{y} - \boldsymbol{\eta}) = \mathbf{0}.$$

  (. . . note that there is a hidden matrix in here—$\partial \mu_i / \partial \eta_i = 1$).
- Be sure you know where these estimating equations came from. The only thing that is permitted to vary in the optimality statement is $\mathbf{W}$.
- The choice $\mathbf{W} \propto \mathbf{I}$ possesses the Gauss-Markov optimality property.

Example 10.4: Simulation study

```r
## Let's code the (negative) log-likelihood taking the full likelihood into account
negll <- function(beta, X, Y) {
  beta0 <- beta[1]
  beta1 <- beta[2]
  eta <- beta0 + beta1*X
  out <- -sum(log(dgamma(Y, shape = eta^2, rate = eta)))
  return(out)
}

## Set seed for reproducibility
set.seed(7345)

## Number of simulations
nsim <- 10000

## Sample size
n <- 100

## Coefficients for data generating mechanism
beta0 <- 1
beta1 <- 1

## Space to store results
results <- matrix(0, nrow = nsim, ncol = 2)
```

Example 10.4: Simulation study

```r
## Conduct simultion
for (j in 1:nsim) {
  X <- runif(n, 1, 5)
  eta <- beta0 + beta1*X
  Y <- rgamma(n, shape = eta^2, rate = eta)
  ols.est <- lm(Y ~ X)
  mle.est <- optim(par = coef(ols.est), negll, X = X, Y = Y)
  results[j,1] <- coef(ols.est)[2]
  results[j,2] <- mle.est$par[2]
  if (round(j/200) == (j/200)) {print(paste(j, "iterations complete!"))}
}

## How did the methods do (on average)?
# > colMeans(results)
# [1] 1.0015568 0.9993384
## Sweet!!

## How do they compare in terms of efficiency?
# > apply(results, 2, sd)
# [1] 0.08605330 0.07879166
## Hmmmmm......
```

Example 10.4: Justifying a GLM

- This does *not* contradict Gauss-Markov logic.
- The award for "most asymptotically efficient overall" still goes to the MLE, which is not generally the same.
  - Exceptions include OLS/Gaussian likelihood.
- Linear unbiased estimating equations are sometimes easier, faster, and more convenient to solve.
- It may be reasonable when the likelihood function is unknown to "settle" for the most efficient within the restricted class based on a presumed mean model and mean-variance relationship.
- The full likelihood may not be known; even when one wants to posit a full likelihood, there is no guarantee that solving it will be easy. Linear unbiased estimating equations are very easy to solve overall.

Example 10.5: Justifying a GLM

- Suppose $Y_i \sim \text{Gamma}(\alpha_i = \eta_i, \beta_i = \eta_i^2)$.
- It is readily shown that $\text{E}[Y_i] = 1/\eta_i$ and $\text{Var}[Y_i] = 1/\eta_i^3 = \mu_i^3$.
- Under this data generating mechanism, the theory says that the inverse Gaussian family with inverse link gives rise to linear unbiased estimating equations that are asymptotically efficient within that subclass of estimators.
- One could, in this bizarre example, treat $\phi = 1$ as fixed and known to squeeze out some efficiency in estimation of $\text{Cov}[\widehat{\boldsymbol{\beta}}]$.

Example 10.6: Justifying a GLM

- Suppose $Y_i \sim \text{Uniform}(0, \theta_i = 2\eta_i)$.
- It is readily shown that $\text{E}[Y_i] = \eta_i$ and $\text{Var}[Y_i] \propto \eta_i^2 = \mu_i^2$.
- Under this data generating mechanism, the theory says that the Gamma family with identity link gives rise to linear unbiased estimating equations that are asymptotically efficient within that subclass of estimators.

Example 10.7: Justifying a GLM

- Suppose $Y_i \sim \text{Borel}(\mu_i = 1 - \eta_i^{-1})$.
  - This is a good (albeit tricky) example because $\text{E}[Y_i] \neq \mu_i$; we need to mind the parameterization.
- Consulting proper resources reveals that $\text{E}[Y_i] = 1/(1-\mu_i) = \eta_i$ and $\text{Var}[Y_i] = \mu_i/(1-\mu_i)^3 = \eta_i^2(\eta_i - 1)$.
- This corresponds to no named GLM of which I am aware.
- The theory of optimally weighted linear unbiased estimating equations nevertheless suggests the following estimating equations:

$$\mathbf{X}^\mathsf{T}\text{diag}(1/(\eta_i^2(\eta_i - 1)))(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{0}.$$

- The previous examples just "happened" to involve mean-variance relationships with families we could name.

Optimality

Further thoughts:

- The word *assumption* is not applied consistently and can refer to different things. An assumption could either be a *basis* for a procedure or a *requirement* for its validity. The theory of generalized estimating equations (BIOS 7346/8346) uses language that reflects this difference very well ("working" correlation).
- Consider the question: "Does OLS assume homoscedasticity?"
- If you mean "homoscedasticity needs to hold in order...
  - ...for the estimate to be unbiased," the answer is "no."
  - ...to achieve Gauss-Markov optimality," the answer is "yes."
  - ...for the model-based variance to be valid," the answer is "yes."
  - ...to have a hope of getting valid inference," the answer is "no."
- Always remember what you're estimating and what you're trying to accomplish. Keep your procedures calibrated accordingly.
  - If your focus is on $\boldsymbol{\beta}$, thinking of GLMs through the framework of optimal estimating equations is generally a good idea.
  - If your focus is on $f_{Y|\mathbf{X}}(y)$, the theory of estimating equations won't give you everything you need.

Table of Contents

The validity/stability trade-off

Discussion point: Stability

- Given that:
  - The solution to the estimating equations does not depend upon the choice of a variance estimator, and
  - The sandwich (and bootstrap) are robust to model misspecification,

  . . . what exactly do we stand to *gain* from something like quasi-likelihood?
- To answer this, think about $\text{Var}[\widehat{\text{Var}}[\mathbf{c}^\top \widehat{\boldsymbol{\beta}}]]$.
  - The variance associated with the model-based standard error will tend to be smaller than the variance of the sandwich-based standard error when the model-based assumptions are correct.
  - When the mean-variance relationship is not reflected by the choice of $\mathbf{W}$, the sandwich-based variance estimator will be *valid* where the model-based variance estimator will not.
- This is essentially the bias-variance trade-off applied to variance estimation instead of mean estimation.
- This phenomenon can be confirmed by simulation.

Summary

This unit:

- Over- and under-dispersion.
- Quasi-likelihood.
- Negative binomial regression.
- Optimality.
- Stability.

Summary: So far

- Random vectors and matrices; multivariate normal theory.
- Ordinary least squares.
- Hypothesis testing and ANOVA.
- Weighted least squares.
- Confidence regions and prediction.
- Diagnostics.
- Exponential families.
- Generalized linear models.
- Sandwich and bootstrap.
- Quasi-likelihood.

Summary: Coming up

- Hypothesis testing for GLMs.
- Diagnostics for GLMs.
- Further considerations for binary outcomes.
- Nonlinear least squares.