4.1 → Introduction

→ Goal is to get point + interval estimates for MLEs
→ sometimes explicit expressions can be found
→ but often numerical methods are needed
→ Typically these are iterative & based on the Newton-Raphson algorithm

4.2 → Example : Failure times

→ Weibull distribution
→ Commonly used to model failure (or survival) times
→ $f(y|\lambda,\theta) = \frac{\lambda y^{\lambda-1}}{\theta^\lambda} \exp\left\{-\left(\frac{y}{\theta}\right)^\lambda\right\}$
$y>0$, shape, scale

→ For this example, e-plot w/ $\lambda=2$ is good

⇒ want to estimate θ

→ exponential family ✓

$f(y|\lambda,\theta) = \exp\{\ln(\dots)\}$
$= \exp\{(\lambda-1)\ln(y) + \underbrace{\ln(\lambda) - \lambda\ln(\theta)}_{c(\theta)} - y^\lambda \frac{1}{\theta^\lambda}\}$
$d(y)$, λ = nuisance ↳ include w/ $c(\theta)$, $a(y)$, $b(\theta)$ ↳ include negative here

⇒ not canonical form ⇒ can't be used directly in specification of GLM
(unless $\lambda=1$ ⇒ $\exp(\theta)$)
→ but can still be used to show parameter estimation in the exponential family

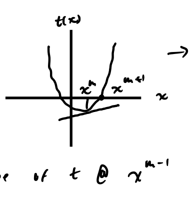
→ likelihood function
→ random sample → $L(\theta|y,\lambda) = \prod_{i=1}^{N} \frac{\lambda y_i^{\lambda-1}}{\theta^\lambda} \exp\left\{-\left(\frac{y_i}{\theta}\right)^\lambda\right\}$
$= \frac{\lambda^n}{\theta^\lambda} (\prod y_i)^{\lambda-1} \exp\{\Sigma(\frac{y_i}{\theta})^\lambda\}$

→ log likelihood → $\ell(\theta) = n\ln(\lambda) - \lambda\ln(\theta)\ (\lambda-1)\Sigma\ln(y_i) + \Sigma(\frac{y_i}{\theta})^\lambda$
→ yes → but use $\Sigma[a(y)b(\theta) + c(\theta) + d(y)]$ for to make easier for next steps
$= \sum_{i=1}^{N}[\ln(\lambda) + (\lambda-1)\ln(y) - \lambda\ln(\theta) - (\frac{y_i}{\theta})^\lambda]$

→ Score function → $U = \frac{d\ell}{d\theta} = \frac{d}{d\theta}[(\dots)]$ (keep in sum)
$= \sum_{i=1}^{N}\left[\frac{-\lambda}{\theta} + \frac{\lambda y_i^\lambda}{\theta^{\lambda+1}}\right]$

→ MLE → $\hat\theta$ such that $U(\theta) = 0$
→ In this case, can easily get expression for $\hat\theta$, but will use algorithm for demonstration purposes

→ Newton Raphson algorithm → Basic idea



→ Goal: Find x where crosses t axis ($t(x)=0$)

→ slope of t @ x^{m-1} is given by
$\left[\frac{dt}{dx}\right]_{x=x^{m-1}} = t'(x^{(m-1)}) = \frac{t(x^{(m)}) - t(x^{(m-1)})}{x^{(m)} - x^{(m-1)}} = \frac{\Delta t}{\Delta x}$ for small Δx

→ want next guess to be the solution ⇒ $t(x^{(m)}) = 0$
$t'(x^{(m-1)}) = (\dots)$ (rearrange)
$t'(x^{(m-1)})(x^{(m)} - x^{(m-1)}) = t(x^{(m)}) - t(x^{(m-1)})$ ($=0$ is goal)
$$\boxed{x^{(m)} = x^{(m-1)} - \frac{t(x^{(m-1)})}{t'(x^{(m-1)})}}$$ → Newton Raphson formula

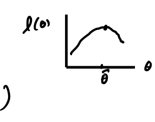
→ For MLE using the score function
⇒ $$\boxed{\theta^{(m)} = \theta^{(m-1)} - \frac{U^{(m-1)}}{U'^{(m-1)}}}$$

→ with weibull $\lambda=2$ → $U = \sum_{i=1}^{N}\left[\frac{-\lambda}{\theta} + \frac{\lambda y_i^\lambda}{\theta^{\lambda+1}}\right]$
$= \frac{-\lambda N}{\theta} + \frac{\lambda}{\theta^{\lambda+1}}\Sigma y_i^\lambda$
$= -\frac{2N}{\theta} + \frac{2}{\theta^3}\Sigma y_i^2$
→ this gets evaluated at successive $\theta^{(m)}$
→ $U' = \frac{dU}{d\theta} = \frac{2N}{\theta^2} - \frac{2(3)\Sigma y_i^2}{\theta^4}$

→ for MLE, it's common to approximate U' by its expected value $E(U')$
⇒ $\boxed{U' \approx E(U')}$
→ for exponential family, we can get this using the information $\mathcal{J}$
$\mathcal{J} = E(-U') = E\left[-\sum_{i=1}^{N} U_i\right] = \sum_{i=1}^{N} E(-U_i')$
individual scores for each obs
$= \sum_{i=1}^{N}\left[\frac{b''(\theta)c'(\theta)}{b'(\theta)} - c''(\theta)\right]$
$= (\dots)$ (trust it)
$= \frac{\lambda^2 N}{\theta^2}$

⇒ Alternative estimating equation → Method of Scoring
$$\boxed{\theta^{(m)} = \theta^{(m-1)} + \frac{U^{(m-1)}}{\mathcal{J}^{(m-1)}}}$$

→ Then iteratively estimate $\theta^{(m)}$, can just use $\bar{y}$ as $\theta^{(1)}$ if want
⇒ get a final estimate of $\hat\theta$ & can calculate U + ℓ at this point



→ The curvature of the function in the vicinity of the MLE determines the reliability of $\hat\theta$.
→ curvature of ℓ is defined by rate of change of U ($= U' \approx E(U')$)
→ If these values are small, then ℓ is flat & $U \approx 0$ for a wide range of values
⇒ $\hat\theta$ is not well determined & it's standard error is large
(derivative of ℓ) ⇒ inflection point
→ In this example → $\hat\theta = 9892.7$
$\mathcal{J} = -E(U') = 2\times10^{-5}$
⇒ can be shown later that $V(\hat\theta)$ is inversely related to $\mathcal{J} = E(-U)$
⇒ $\sigma_{\hat\theta} = \sqrt{1/\mathcal{J}}$ → $= \frac{1}{\sqrt{.00002}} = 707$
→ if sampling dist of $\hat\theta$ is approximately normal
95% CI ≈ $\hat\theta \pm z_\alpha\sigma_{\hat\theta} = 9892 \pm 1.96\times707 = [8506, 11278]$

4.3 → Maximum likelihood Estimation

→ Setup → $Y_1, \dots, Y_N$ are II RVs that satisfy properties of a GLM
→ Goal is to estimate parameters β which are related to the Y_i's through $E(Y_i) = \mu_i$ & $g(\mu_i) = x_i^T\beta$

→ log likelihood function → for each Y_i, we have
$\ell_i = Y_i b(\theta_i) + c(\theta_i) + d(Y_i)$
→ Also know → $E(Y_i) = \mu_i = \frac{-c'(\theta)}{b'(\theta)}$
$V(Y_i) = [b''(\theta)c'(\theta) - c''(\theta)b'(\theta)]/[b'(\theta_i)]^3$
$g(\mu_i) = x_i^T\beta = \eta_i$

→ For all Y_i's → $\ell = \sum_{i=1}^{N}\ell_i = \Sigma Y_i b(\theta_i) + \Sigma c(\theta_i) + \Sigma d(Y_i)$

→ MLE → To get MLE for β_j, need
$\frac{\partial\ell}{\partial\beta_j} = U_j = \sum_{i=1}^{N}\left[\frac{\partial\ell_i}{\partial\beta_j}\right] = \sum_{i=1}^{N}\left[\underbrace{\frac{\partial\ell_i}{\partial\theta_i}\cdot\frac{\partial\theta_i}{\partial\mu_i}}\cdot\frac{\partial\mu_i}{\partial\beta_j}\right]$, using chain rule

→ $\frac{\partial\ell_i}{\partial\theta_i} = \frac{\partial}{\partial\theta_i}[Y_i b(\theta_i) + c(\theta_i) + d(Y_i)]$
$= Y_i b'(\theta_i) + c'(\theta)$ ← $E(Y_i) = \frac{-c'(\theta)}{b'(\theta)}$ ⇒ $c'(\theta) = -E(Y_i)b'(\theta)$
$= Y_i b'(\theta) - b'(\theta)\mu_i$
$= b'(\theta)(Y_i - \mu_i)$

→ $\frac{\partial\theta_i}{\partial\mu_i} = \frac{1}{\partial\mu_i/\partial\theta_i}$ → $\frac{\partial\mu_i}{\partial\theta_i} = \frac{\partial}{\partial\theta}\left[-\frac{c'(\theta)}{b'(\theta)}\right] = \frac{-c''(\theta_i)}{b'(\theta_i)} + \frac{c'(\theta_i)b''(\theta_i)}{(b'(\theta_i))^2}$ (product / quotient rule)
$= b'(\theta_i)V(Y_i)$ (chose to ÷ by another $b'(\theta)$ to get $\frac{(\dots)}{(b'(\theta))^3}$ ⇒ multiply by $b'(\theta)$ to balance)

→ $\frac{\partial\mu_i}{\partial\beta_j} = \frac{\partial\mu_i}{\partial\eta_i}\cdot\frac{\partial\eta_i}{\partial\beta_j} = \frac{\partial\mu_i}{\partial\eta_i}x_{ij}$, $\eta_i = x_i^T\beta$
↳ just invert link function to get $\mu_i = f(\underbrace{x_i^T\beta}_{=\eta_i})$

Score ⇒ $U_j = \sum_{i=1}^{N}\left[\frac{(Y_i-\mu_i)}{V(Y_i)}x_{ij}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)\right]$

→ var-cov matrix of U_j's has terms $\mathcal{J}_{jk} = E[U_jU_k]$
→ These form the information matrix $\mathcal{J}$
$\mathcal{J}_{jk} = E\left\{\sum_{i=1}^{N}\left[\frac{(Y_i-\mu_i)}{V(Y_i)}x_{ij}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)\right]\sum_{\ell=1}^{N}\left[\frac{(Y_\ell-\mu_\ell)}{V(Y_\ell)}x_{\ell k}\left(\frac{\partial\mu_\ell}{\partial\eta_\ell}\right)\right]\right\} = \sum_{i=1}^{N}\frac{x_{ij}x_{ik}}{V(Y_i)}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)^2$
$= \sum_{i=1}^{N}\frac{E[(Y_i-\mu_i)^2]x_{ij}x_{ik}}{[V(Y_i)]^2}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)^2$, b/c $Cov(Y_i,Y_\ell) = E[(Y_i-\mu_i)(Y_\ell-\mu_\ell)] = 0$ for $i\neq\ell$

⇒ Method of scoring $\left(\theta^{(m)} = \theta^{(m-1)} + \frac{U^{(m-1)}}{\mathcal{J}^{(m-1)}}\right)$ generalizes to
$b^{(m)} = b^{(m-1)} + [\mathcal{J}^{(m-1)}]^{-1}U^{(m-1)}$
↳ $= \begin{bmatrix}\beta_1\\ \vdots\\ \beta_p\end{bmatrix}$ @ mth iteration; vector of elements U_j evaluated at $b^{(m)}$

$\mathcal{J}^{(m-1)}b^{(m)} = \mathcal{J}^{(m-1)}b^{(m-1)} + U^{(m-1)}$ (multiply both sides by $\mathcal{J}^{(m-1)}$)
$= \sum_{k=1}^{p}\sum_{i=1}^{N}\frac{x_{ij}x_{ik}}{V(Y_i)}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)^2 b_k^{(m-1)} + \sum_{i=1}^{N}\frac{(Y_i-\mu_i)x_{ij}}{V(Y_i)}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)$
$= X^TWz$
↳ has elements $z_i = \sum_{k=1}^{p}x_{ik}b_k^{(m-1)} + (Y_i-\mu_i)\underbrace{\left(\frac{\partial\eta_i}{\partial\mu_i}\right)}$ ↳ evaluated at $b^{(m-1)}$

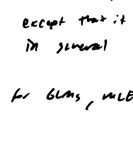
→ rewriting $\mathcal{J}$ → $\mathcal{J} = X^TWX$
↳ $N\times N$ diagonal matrix w/ elements $w_{ii} = \frac{1}{V(Y_i)}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)^2$

$= X^TWXb^{(m)} = X^TWz$
⇒ Same form as normal equations for a linear model obtained by weighted least squares, except that it has to be solved iteratively b/c in general W + z depend on b

→ Thus for GLMs, MLEs are obtained by an iterative weighted least squares procedure
key result
? on the specifics of the derivation though

4.4 → Poisson regression example

→ Setup → $Y_i \overset{ii}{\sim} \text{Poisson}(\mu_i)$



→ can ... wth ...
$g(\mu_i) = \mu_i = \beta_1 + \beta_2 x_i = x_i^T\beta = \eta_i$
→ $\beta = [\beta_1\ \beta_2]^T$

→ Calculations
$w_{ii} = \frac{1}{V(Y_i)}\left(\frac{\partial\mu_i}{\partial\eta_i}\right)^2$
$= \frac{1}{x_i^T\beta}(1)^2$
$= (\dots)^{-1}$

→ $z_i = x_i^T b + (y_i - x_i^T b)(1)$
$= y_i$

→ $\mathcal{J} = X^TWX = \dots$

→ $X^TWz = (\dots)$

→ $(X^TWX)^{(m-1)}b^{(m)} = X^TWz^{(m-1)}$
$b^{(m)} = (X^TWX)^{-1}X^TWz^{(m-1)}$

→ Pick initial estimates $b_1^{(1)}$ + $b_2^{(1)}$, then solve until algorithm converges

→ At end, get final estimates $\hat\beta$ &
$\mathcal{J}^{-1} = (X^TWX)^{-1}$ = var-cov matrix of $\hat\beta$
⇒ s.e. of $\hat\beta$: $\hat\beta_k \pm 1.96\sqrt{\mathcal{J}^{kk}}$