NB10 Case Study

STAT 206
23 Feb 21

Lecture

1

The Gaussian sampling model (in cheating mode, in which we (a) look at histograms & Normal qqplots of the entire dataset to find a reasonable sampling ~~distribution~~ model (SM), and (b) pretend that we knew all along that the SM from (a) was a good choice) is not a good fit to the data, because the data vector $y = (y_1, \dots, y_n)$ has outliers in both tails; the Normal qqplot showed that a symmetric SM is reasonable. R code (parts 1) This suggests

a different use of the family of t-distributions than that made by Mr. Gosset in his small-sample adjustment to Mr. Neyman's confidence intervals: let's use the t family as a SM:

new model for NB10 data

$(i = 1, \dots, n)$

$$(y_i \mid \mu\, \sigma\, \nu\, \mathbb{B}) \overset{IID}{\sim} t_\nu(\mu, \sigma^2)$$

$\leftarrow (t\ SM)$

the $t_\nu(\mu, \sigma^2)$ PDF (Gelman et al. Appendix A) is

$$p(y_i \mid \mu\, \sigma\, \nu\, \mathbb{B})$$

looks complicated, but actually isn't too bad

R code

$$= \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma \cdot \Gamma\left(\frac{\nu}{2}\right)\sqrt{\pi \nu}} \left[1 + \frac{1}{\nu}\left(\frac{y_i - \mu}{\sigma}\right)^2\right]^{-\frac{\nu+1}{2}}$$

here $\underset{\sim}{\theta} = (\mu, \sigma, \nu)$ $(k = 3)$

here $\Gamma(x)$ is the Gamma function $\Gamma(x) \doteq \int_0^\infty t^{x-1} e^{-t}\, dt$,

defined as an integral because $g(t) = t^{x-1} e^{-t}$ has no anti-derivative in closed form

for any positive integer n, $\Gamma(n) = (n-1)!$, so $\Gamma(x)$ is a continuous generalization of the factorial function

$\Gamma(x) \uparrow +\infty$ really fast as $x \uparrow \infty$, so we're also interested in $\log \Gamma(x)$

Likelihood analysis of t SM first

joint sampling distribution

$\underset{\sim}{\theta} = (\mu, \sigma, \nu)$ $(k=3)$

$-\infty < \mu < \infty$

$\sigma > 0$

$\nu > 0$

$$p(\underset{\sim}{y} \mid \mu\,\sigma\,\nu\,\mathbb{T}\,\mathcal{B}) = \prod_{i=1}^{n} p(y_i \mid \mu\,\sigma\,\nu\,\mathbb{T}\,\mathcal{B})$$

$$= \prod_{i=1}^{n} \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\,\Gamma\left(\frac{\nu}{2}\right)\sqrt{\pi\nu}} \left[1 + \frac{1}{\nu}\left(\frac{y_i - \mu}{\sigma}\right)^2\right]^{-\frac{\nu+1}{2}}$$

$$= \frac{\left[\Gamma\left(\frac{\nu+1}{2}\right)\right]^n \left\{\prod_{i=1}^{n}\left[1 + \frac{1}{\nu}\left(\frac{y_i - \mu}{\sigma}\right)^2\right]\right\}^{-\frac{\nu+1}{2}}}{\sigma^n \left[\Gamma\left(\frac{\nu}{2}\right)\right]^n (\pi\nu)^{n/2}} = \ell(\mu\,\sigma\,\nu \mid \underset{\sim}{y}\,\mathbb{T}\,\mathcal{B})$$

(likelihood)

log likelihood

$$\ell\ell(\mu\,\sigma\,\nu \mid \underset{\sim}{y}\,\mathbb{T}\,\mathcal{B}) =$$

$$n \log \Gamma\left(\frac{\nu+1}{2}\right) - n \log \sigma - n \log \Gamma\left(\frac{\nu}{2}\right)$$

$$- \frac{n}{2} \log \nu - \left(\frac{\nu+1}{2}\right) \sum_{i=1}^{n} \log\left[1 + \frac{1}{\nu}\left(\frac{y_i - \mu}{\sigma}\right)^2\right]$$

(~~minimal~~) sufficient statistics?

a minimal sufficient statistic is $\underset{\sim}{y} = (y_1, \dots, y_n)$ (!)

how to find the MLEs? | you can try the calculus approach:

solve this system of 3

$$\left\{ \begin{array}{l} \frac{\partial}{\partial \mu} \ell\ell(\mu\,\sigma\,\nu \mid y\,\mathbb{T}\,\mathcal{B}) = 0 \\ \frac{\partial}{\partial \sigma} \ell\ell(\mu\,\sigma\,\nu \mid y\,\mathbb{T}\,\mathcal{B}) = 0 \\ \frac{\partial}{\partial \nu} \ell\ell(\mu\,\sigma\,\nu \mid y\,\mathbb{T}\,\mathcal{B}) = 0 \end{array} \right\} \quad (*)$$

equations in the 3 unknowns (μ, σ, ν)

this will work, but no closed-form solutions exist (mainly because of the $\log \Gamma(\cdot)$ functions)

you can solve the system $(*)$ numerically, but if we're going to go numeric on this problem there's an easier way: let's numerically <u>maximize</u>

$$\ell\ell(\mu\,\sigma\,\nu \mid y\,\mathbb{T}\,\mathcal{B}) : \mathbb{R}^{3^{\kappa k}} \to \mathbb{R}^1$$

R has a general-purpose optimization built-in function called optim, for maximizing or minimizing functions from $\mathbb{R}^k$ to $\mathbb{R}^1$ (R code).

Bayesian analysis of the t SM in the NB10 case study

Our Bayesian model would be

$$\left\{ \begin{array}{c} (\mu\, \sigma\, \nu \mid \mathcal{B}) \sim p(\mu\, \sigma\, \nu \mid \mathcal{B}) \\ (y_i \mid \mu\, \sigma\, \nu\, \mathbb{T}\, \mathcal{B}) \stackrel{IID}{\sim} t_\nu(\mu, \sigma^2) \\ (i = 1, \dots, n) \end{array} \right.$$

there is no conjugate prior in this SM; from context we want a LI prior, but conjugacy can't help to create one. So we need a new Bayesian computation method in the year 2021: 3 possibilities: ① Laplace approximation

② simulation-based numerical approximation of posterior distributions

---

③ large-sample Gaussian approximations

---

if time permits I'll sketch the Laplace approximation story in a discussion section

---

③ is simple to state

by Bernstein-von Mises, for large n and a LI prior

$$(\underset{\sim}{\theta} \mid \mathcal{D}\,[SM]\,[PM]\,\mathcal{B}) \,\dot\sim\, N_k\left(\hat{\theta}_{MLE}, \hat{I}^{-1}\right)$$

---

this produces exactly the same posterior (credible) intervals for $(\theta_1, \dots, \theta_k)$ as with the large-sample maximum-likelihood approach's confidence intervals data-driven

empirical CDF based on a data vector

$y = (y_1, \ldots, y_n)$: $\hat{F}_n(t) = \dfrac{\#(y_i \le t)}{n}$

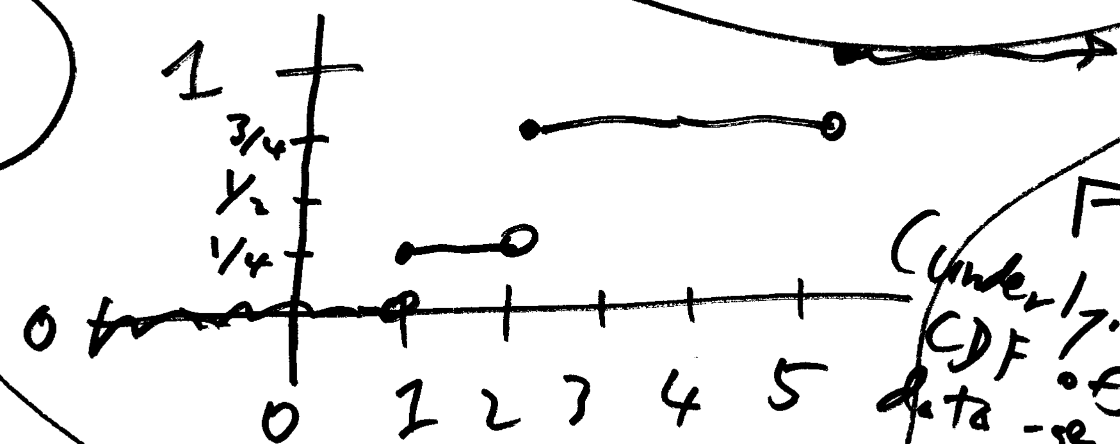
$= \dfrac{1}{n} \sum_{i=1}^{n} I(y_i \le t)$

CDF

$F_Y(y) = P(Y \le y)$

always piecewise constant

$\{1, 2, 2, 5\}$



F: (underlying CDF of data-generating process)

$\hat{F}_n(t)$, in absence of external information about F, is the absolutely best estimate of F ($\hat{F}_n(t)$ is the

non-parametric MLE of F)

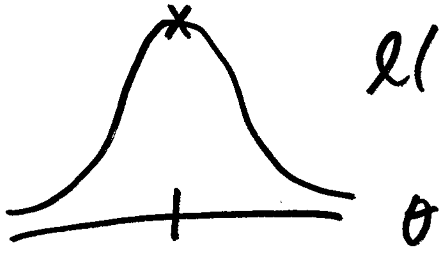
we're not assuming, e.g. $F \leftrightarrow N(\mu, \sigma^2)$

PDF

parametric (SM)

$k=1$

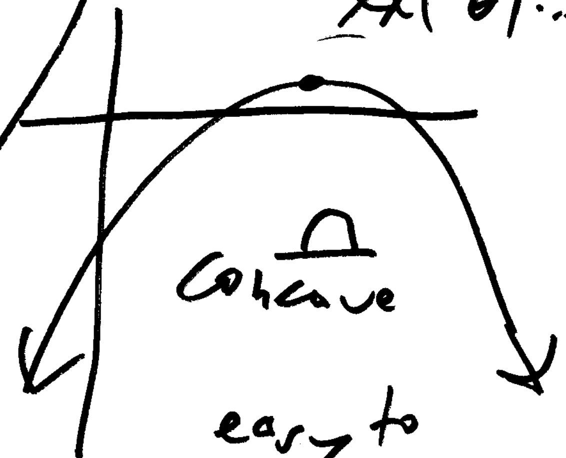
$n=$ large



$\ell(\theta \mid D \ldots B)$

$\hat{\theta}_{MLE}$ = easy to find max

(local max = global max)



$\ell\ell(\theta \mid \ldots)$

concave

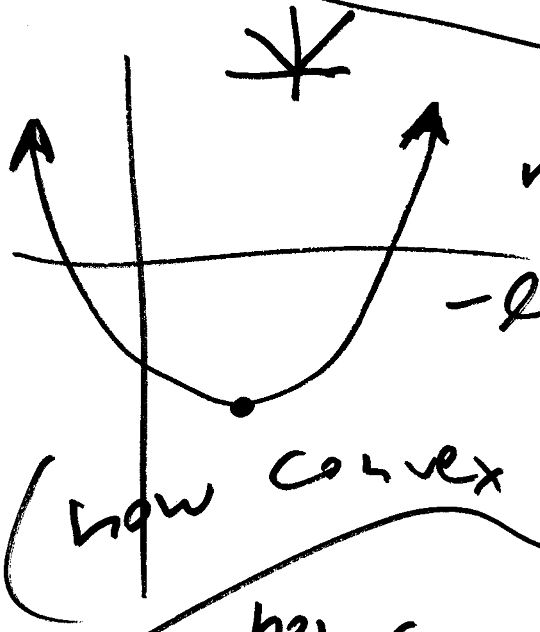
easy to find max

likelihood f'n is log concave

the $\hat{\theta}_{MLE}$ that maximizes $\ell\ell(\theta \mid \sim)$

also minimizes $-\ell\ell(\theta \mid \sim)$

"loss" function



minimize $-\ell\ell(\theta \mid \sim)$

(now convex)

this is called a convex optimization problem (machine learning)

(easy to solve)

non-convex optimization is (NP) hard

(non-polynomial time)