Discretionary Note

Anish Krishna Lakkapragada

IF YOU USE THIS FILE TO CHEAT, YOU ARE NOT ONLY STUPID BUT YOU ARE CHEATING YOURSELF OUT OF THE ABILITY TO FALL IN LOVE WITH MATH. Furthermore, I am not smarter than you and my solutions did not always get a perfect score.

CONTENT STARTS ON NEXT PAGE.

To access the general instructions for this repository head here.

STATS 242 HW 6

March 5, 2025

Number of late days: 0; Collaborators: Matt Sprintson

1.

We first compute the first moment μ_1 of $X \sim \text{Geom}(p)$, which is given by $\mu_1 = \mathbb{E}[X] = \frac{1}{p}$. We now solve for the MoM estimator $\hat{p}$ where the observed sample moment $\hat{\mu_1} = \frac{1}{n}(X_1 + \cdots + X_n) = \bar{X}$ is equal to theoretical moment μ_1 with the estimated $\hat{p}$ parameter:

$$\hat{\mu_1} = \bar{X} = \frac{1}{\hat{p}}$$
$$\hat{p} = \frac{1}{\bar{X}}$$

Thus, $\hat{p} = \frac{1}{\bar{X}}$ is the MoM estimator $\hat{p}$ for p. We now compute the MLE of p. We first start by finding the log-likelihood function $\ell_n(p)$:

$$\begin{aligned}\ell_n(p) = \log[\text{lik}(p)] = \Sigma_{i=1}^n \log\, f(X_i|p) = \Sigma_{i=1}^n \log(p(1-p)^{x_i-1}) &= \Sigma_{i=1}^n \log(p) + (x_i - 1)\log(1-p) \\ &= n\log(p) + \Sigma_{i=1}^n (x_i - 1)\log(1-p)\end{aligned}$$

We now solve for the MLE $\hat{p}$ by solving $\ell_n'(\hat{p}) = 0$:

$$\ell_n'(\hat{p}) = 0$$
$$(n\log(\hat{p}) + \Sigma_{i=1}^n (x_i - 1)\log(1-\hat{p}))' = 0$$
$$\frac{n}{\hat{p}} - \Sigma_{i=1}^n \frac{(x_i - 1)}{1-\hat{p}} = 0$$
$$\frac{n}{\hat{p}} - \frac{1}{1-\hat{p}}(n\bar{X} - n) = 0$$
$$\frac{n(1-\hat{p}) - \hat{p}(n\bar{X} - n)}{\hat{p}(1-\hat{p})} = 0$$
$$n(1-\hat{p}) - \hat{p}(n\bar{X} - n) = 0$$
$$n - n\hat{p} - \hat{p}n\bar{X} + n\hat{p} = 0$$
$$n = n\hat{p}\bar{X}$$
$$1 = \hat{p}\bar{X}$$
$$\hat{p} = \frac{1}{\bar{X}}$$

Thus the MLE of p is $\hat{p} = \frac{1}{\bar{X}}$. We now find the sampling distribution of the MLE of p, under large n. We begin by computing the Fisher Information $I(p)$:

$$I(p) = -\mathbb{E}_p[\frac{\partial^2}{\partial p^2}\log f(X|p)] = -\mathbb{E}_p[\frac{\partial^2}{\partial p^2}(\log(p) + (x-1)\log(1-p))]$$
$$= -\mathbb{E}_p[\frac{\partial}{\partial p}(\frac{1}{p} + (1-x)\frac{1}{1-p})] = -\mathbb{E}_p[-\frac{1}{p^2} + \frac{(1-x)}{(1-p)^2}] = -(-\frac{1}{p^2} + \frac{1}{(1-p)^2} - \frac{\mathbb{E}_p[x]}{(1-p)^2})$$
$$= \frac{1}{p^2} - \frac{1}{(1-p)^2} + \frac{1}{p(1-p)^2} = \frac{(1-p)^2 - p^2 + p}{p^2(1-p)^2} = \frac{1-p}{p^2(1-p)^2} = \frac{1}{p^2(1-p)}$$

So this means that for the MLE $\hat{p}$, $\sqrt{n}(\hat{p}-p) \to \mathcal{N}(0, \frac{1}{I(p)})$ or $\sqrt{n}(\hat{p}-p) \to \mathcal{N}(0, p^2(1-p))$ meaning $\hat{p} \to \mathcal{N}(p, \frac{p^2(1-p)}{n})$ as $n \to \infty$.

2.

We first compute the MoM estimate of p. To do so, we compute the first moment μ_1 of $X \sim \text{NegBinom}(r,p)$ which is given by $\mu_1 = \mathbb{E}[X] = \frac{pr}{1-p}$. We now solve for the MoM estimator $\hat{p}$ where the observed sample moment $\hat{\mu_1} = \frac{1}{n}(X_1 + \cdots + X_n) = \bar{X}$ is equal to theoretical moment μ_1 with the estimated $\hat{p}$ parameter:

$$\hat{\mu_1} = \bar{X} = \frac{\hat{p}r}{1-\hat{p}}$$
$$\bar{X}(1-\hat{p}) = \hat{p}r$$
$$\bar{X} = \hat{p}\bar{X} + \hat{p}r$$
$$\bar{X} = \hat{p}(\bar{X}+r)$$
$$\hat{p} = \frac{\bar{X}}{\bar{X}+r}$$

Thus the MoM estimator of p is given by $\hat{p} = \frac{\bar{X}}{\bar{X}+r}$. We now compute the MLE of p. We first start by finding the log-likelihood function $\ell_n(p)$:

$$\ell_n(p) = \log[\mathrm{lik}(p)] = \Sigma_{i=1}^n \log\, f(x_i|p) = \Sigma_{i=1}^n \log(\binom{x+r-1}{x_i}(1-p)^r p^x)$$
$$= \Sigma_{i=1}^n \log(\binom{x_i+r-1}{x_i}) + r\log(1-p) + x_i\log(p) = nr\log(1-p) + \Sigma_{i=1}^n \log(\binom{x_i+r-1}{x_i}) + x_i\log(p)$$

We now solve for the MLE $\hat{p}$ by solving $\ell_n'(\hat{p}) = 0$:

$$\ell_n'(\hat{p}) = 0$$
$$(nr\log(1-\hat{p}) + \Sigma_{i=1}^n \log(\binom{x_i+r-1}{x_i}) + x_i\log(\hat{p}))' = 0$$
$$\frac{-nr}{1-\hat{p}} + \Sigma_{i=1}^n \frac{x_i}{\hat{p}} = 0$$
$$-\frac{nr\hat{p}}{\hat{p}(1-\hat{p})} + \frac{n\bar{X}(1-\hat{p})}{\hat{p}(1-\hat{p})} = 0$$
$$\frac{n\bar{X}(1-\hat{p}) - nr\hat{p}}{\hat{p}(1-\hat{p})} = 0$$
$$n\bar{X}(1-\hat{p}) - nr\hat{p} = 0$$
$$\bar{X}(1-\hat{p}) - r\hat{p} = 0$$
$$\bar{X} - \hat{p}\bar{X} - \hat{p}r = 0$$
$$\bar{X} = \hat{p}(\bar{X}+r)$$
$$\hat{p} = \frac{\bar{X}}{\bar{X}+r}$$

Thus the MLE of p is given by $\hat{p} = \frac{\bar{X}}{\bar{X}+r}$. We now find the sampling distribution of the MLE of p, under large n. We begin by computing the Fisher Information $I(p)$:

$$I(p) = \text{Var}_p[\frac{\partial}{\partial p}\text{log } f(x|p)] = \text{Var}_p[\frac{\partial}{\partial p}[\text{log}(\binom{x+r-1}{x}) + r\text{log}(1-p) + x\text{log}(p)]]$$
$$= \text{Var}_p[\frac{-r}{1-p} + \frac{x}{p}] = \frac{1}{p^2}\text{Var}_p[x] = \frac{r}{p(1-p)^2}$$

So this means that for the MLE $\hat{p}$, $\sqrt{n}(\hat{p}-p) \to \mathcal{N}(0, \frac{1}{I(p)})$ or $\sqrt{n}(\hat{p}-p) \to \mathcal{N}(0, \frac{p(1-p)^2}{r})$ so $\hat{p} \to \mathcal{N}(p, \frac{p(1-p)^2}{nr})$.

3.

(a) The PDF of Pareto(α, x_m) is given by $f(x|\alpha, x_m) = \frac{\alpha x_m^\alpha}{x^{\alpha+1}}$. Thus, the PDF of Pareto$(\theta, 1)$ is given by:

$$f(x|\theta, 1) = \frac{\theta \cdot 1^\theta}{x^{\theta+1}} = \frac{\theta}{x^{\theta+1}}$$

Expressed differently, we can set functions $T(x) = \log(\frac{1}{x}), A(\theta) = \log(\frac{1}{\theta}), h(x) = \frac{1}{x}$ so the PDF

$$f(x|\theta) = e^{\theta T(x) - A(\theta)}h(x) = \frac{e^{\theta T(x)}}{xe^{A(\theta)}} = \frac{(e^{\log(\frac{1}{x})})^\theta}{xe^{\log\frac{1}{\theta}}} = \frac{1}{x^\theta \cdot x \cdot \frac{1}{\theta}} = \frac{\theta}{x^{\theta+1}}$$

is identical to the aforementioned derived Pareto$(\theta, 1)$ PDF.

(b) We differentiate both sides of the identify $1 = \int_\mathcal{X} f(x|\theta)dx$ to compute $\mathbb{E}_\theta[T(X)]$:

$$\int_\mathcal{X} f(x|\theta)dx = 1$$
$$\frac{d}{d\theta}\int_\mathcal{X} f(x|\theta)dx = \int_\mathcal{X} \frac{d}{d\theta}f(x|\theta)dx = \frac{d}{d\theta}(1) = 0$$
$$\int_\mathcal{X} \frac{d}{d\theta}[e^{\theta T(x) - A(\theta)}h(x)]dx = 0$$
$$\int_\mathcal{X} [(e^{\theta T(x) - A(\theta)})'h(x) + h'(x)e^{\theta T(x) - A(\theta)}]dx = \int_\mathcal{X} [(e^{\theta T(x) - A(\theta)})(T(x) - A'(\theta))h(x) + 0]dx = 0$$
$$\int_X [e^{\theta T(x) - A(\theta)}h(x)](T(x) - A'(\theta))dx = 0$$
$$\int_\mathcal{X} f(x|\theta)(T(x) - A'(\theta))dx = \mathbb{E}_\theta[T(x) - A'(\theta)] = 0$$
$$\mathbb{E}_\theta[T(x)] = \mathbb{E}_\theta[A'(\theta)] = A'(\theta)$$

So we have $\mathbb{E}_\theta[T(x)] = A'(\theta)$. We now verify this is the case for our Pareto$(\theta, 1)$ model with its defined functions $T(x) = \log(\frac{1}{x}), A(\theta) = \log(\frac{1}{\theta}), h(x) = \frac{1}{x}$ in part (a). We first compute $\mathbb{E}_\theta[T(x)]$:

$$\mathbb{E}_\theta[T(x)] = \mathbb{E}_\theta[\log(\frac{1}{x})] = \int_{-\infty}^{\infty} f(x|\theta)\log(\frac{1}{x})dx = \int_{1}^{\infty} \frac{\theta}{x^{\theta+1}}\log(\frac{1}{x})dx$$

We can use a u-substitution of $u = \frac{1}{x}$ so $-u^{\theta-1}du = -\frac{x}{x^\theta} \cdot -\frac{1}{x^2}dx = \frac{1}{x^{\theta+1}}dx$. Thus, this integral can be re-written as:

$$\int_1^0 \log(u) \cdot \theta \cdot (-u^{\theta-1})du = \theta \int_0^1 \log(u) \cdot u^{\theta-1}du$$

Using integration by parts, we can set $a = \log(u), da = \frac{1}{u}du, db = u^{\theta-1}du, b = \frac{1}{\theta}u^\theta$. As such, we get:

$$\theta \int_0^1 \log(u) \cdot u^{\theta-1}du = \theta[ab \Big|_{u=0}^{u=1} - \int_0^1 bda]$$

$$= \theta[\frac{\log(u) \cdot u^\theta}{\theta}] \Big|_0^1 - \theta \int_0^1 \frac{1}{\theta}u^{\theta-1}du = 0 - \int_0^1 u^{\theta-1}du = -\frac{1}{\theta}u^\theta \Big|_0^1 = -\frac{1}{\theta} + 0 = -\frac{1}{\theta}$$

So $\mathbb{E}_\theta[T(x)] = -\frac{1}{\theta}$. We now compute $A'(\theta)$:

$$A'(\theta) = \frac{1}{\frac{1}{\theta}} \cdot \frac{-1}{\theta^2} = \frac{-\theta}{\theta^2} = -\frac{1}{\theta}$$

So because $\mathbb{E}_\theta[T(x)] = -\frac{1}{\theta} = A'(\theta)$, we have verified for our Pareto model in part (a) that $\mathbb{E}_\theta[T(x)] = A'(\theta)$.

(c) We first find the log-likelihood function $\ell_n(\theta)$:

$$\ell_n(\theta) = \log[\text{lik}(\theta)] = \Sigma_{i=1}^n \log f(x_i|\theta) = \Sigma_{i=1}^n \log[e^{\theta T(x_i) - A(\theta)}h(x_i)] =$$
$$\Sigma_{i=1}^n \log[e^{\theta T(x_i)}] - \log[e^{A(\theta)}] + \log[h(x_i)] = -nA(\theta) + \Sigma_{i=1}^n \theta T(x_i) + \log[h(x_i)]$$

To get the MLE of θ we solve for the solution to $0 = \ell_n'(\theta)$:

$$\ell_n'(\theta) = 0$$
$$(-nA(\theta) + \Sigma_{i=1}^n \theta T(x_i) + \log[h(x_i)])' = 0$$
$$-nA'(\theta) + \Sigma_{i=1}^n T(x_i) = 0$$
$$nA'(\theta) = \Sigma_{i=1}^n T(x_i)$$
$$A'(\theta) = \frac{1}{n}\Sigma_{i=1}^n T(x_i)$$
$$\mathbb{E}_\theta[T(x)] = A'(\theta) = \frac{1}{n}\Sigma_{i=1}^n T(x_i)$$

So therefore the MLE of θ is the unique solution to $\mathbb{E}_\theta[T(x)] = \frac{1}{n}\Sigma_{i=1}^n T(x_i)$. Because the generalized MoM estimator of θ is also the solution to $\mathbb{E}_\theta[T(x)] = \frac{1}{n}\Sigma_{i=1}^n T(x_i)$, and we are guaranteed this solution is unique, the generalized MoM estimator of θ is equivalent to the MLE of θ.

An example of this occuring would be the generalized MoM estimator based on $T(x) = \log(x)$ for the Pareto$(\theta, 1)$ model coinciding with the MLE.

4.

(a) By the CLT, $\sqrt{n}(\hat{p}-p) \to \mathcal{N}(0, p(1-p))$. Informally, this gives us that $\text{Var}[\sqrt{n}(\hat{p}-p)] \approx \hat{p}(1-\hat{p}) \implies n\text{Var}(\hat{p}-p) \approx \hat{p}(1-\hat{p}) \implies \text{Var}(p) \approx \frac{\hat{p}(1-\hat{p})}{n}$. Using this approximation for the variance of p, we can compute a 95% confidence interval for p as $\hat{p} \pm z^{\frac{0.05}{2}}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$ where $z^{\frac{0.05}{2}}$ is the upper-$\frac{0.05}{2}$ point of the standard normal distribution.

(b) We solve $\sqrt{n}(\hat{p}-p) = \pm\sqrt{p(1-p)}z^{(\alpha/2)}$ below to create a confidence interval for p:

$$
\begin{aligned}
\sqrt{n}(\hat{p}-p) &= \pm\sqrt{p(1-p)}z^{(\alpha/2)} \\
n(\hat{p}-p)^2 &= p(1-p)(z^{(\alpha/2)})^2 \\
n\hat{p}^2 - 2np\hat{p} + np^2 &= (z^{(\alpha/2)})^2 p - (z^{(\alpha/2)})^2 p^2 \\
(n + (z^{(\alpha/2)})^2)p^2 - (2n\hat{p} + (z^{(\alpha/2)})^2)p + n\hat{p}^2 &= 0
\end{aligned}
$$

Using the quadratic formula, we construct the following confidence interval for p:

$$
p = \frac{(2n\hat{p} + (z^{(\alpha/2)})^2) \pm \sqrt{(2n\hat{p} + (z^{(\alpha/2)})^2)^2 - 4n\hat{p}^2(n + (z^{(\alpha/2)})^2)}}{2(n + (z^{(\alpha/2)})^2)}
$$

(c) For our confidence interval in part (a), we report the simulated coverage probabilities below:

	$n = 10$	$n = 40$	$n = 100$
$p = 0.1$	0.65061	0.91435	0.93267
$p = 0.3$	0.83926	0.92955	0.94984
$p = 0.5$	0.8922	0.91919	0.94205

We similarly report the simulated coverage probabilities for our confidence interval in part (b):

	$n = 10$	$n = 40$	$n = 100$
$p = 0.1$	0.93053	0.94406	0.9375
$p = 0.3$	0.92463	0.94432	0.93659
$p = 0.5$	0.97861	0.961	0.94316

We can clearly see from comparing these two tables that our interval construction in part (b) leads to generally much higher simulated coverage probabilities.

```
        # %%
import numpy as np
from scipy.stats import norm

```

```python
z_alpha = np.abs(norm.ppf(0.05/2))
z_alpha_squared = z_alpha ** 2

NUM_SIMULATIONS = 100 * 1000
SAMPLE_SIZES = [10, 40, 100]
TRUE_PARAMS = [0.1, 0.3, 0.5]

def generate_data(n,p):
    return np.random.binomial(n=n, p=p) # number of successes

def part_a_interval(num_successes, n):
    p_hat = num_successes / n
    delta =  z_alpha * np.sqrt(p_hat * (1 - p_hat) / n)
    return p_hat - delta, p_hat + delta

def part_b_interval(num_successes, n):
    p_hat = num_successes / n
    denominator = 2 * (n + z_alpha_squared)
    center = (2 * n * p_hat + z_alpha_squared) / denominator
    delta = np.sqrt((2 * n * p_hat + z_alpha_squared) ** 2 - 4 * n *
    ↪  (p_hat ** 2) * (n + z_alpha_squared))
    delta /= denominator
    return center - delta, center + delta

# %%


for n in SAMPLE_SIZES:
    for p in TRUE_PARAMS:
        NUM_COVERAGE_A = 0
        for _ in range(NUM_SIMULATIONS):
            n_successes = generate_data(n, p)
            lower, upper = part_a_interval(n_successes, n)

            if p >= lower and p <= upper: NUM_COVERAGE_A += 1

        print(f"[PART (A)] For sample size n={n} and p={p}, simulated
        ↪  coverage probability: {NUM_COVERAGE_A / NUM_SIMULATIONS}")



for n in SAMPLE_SIZES:
    for p in TRUE_PARAMS:
        NUM_COVERAGE_B = 0
        for _ in range(NUM_SIMULATIONS):
            n_successes = generate_data(n, p)
            lower, upper = part_b_interval(n_successes, n)
```

```python

                if p >= lower and p <= upper: NUM_COVERAGE_B += 1

            print(f"[PART (B)] For sample size n={n} and p={p}, simulated
            ↪  coverage probability: {NUM_COVERAGE_B / NUM_SIMULATIONS}")


# %%
```