STAT 7650. Homework 2 (Due: Tuesday, February 10, 2026)

1. (25 points) The following data are iid sample from a Cauchy(θ, 1) distribution.

$$1.77, -0.23, 2.76, 3.80, 3.47, 56.75, -1.34, 4.24, -2.44, 3.29,$$
$$3.71, -2.40, 4.53, -0.07, -1.05, -13.87, -2.53, -1.75, 0.27, 43.21$$

Answer the following questions.

(a) Graph the log likelihood function. Find the MLE for θ using the Newton-Raphson method. Try all of the following starting points: -11, -1, 0, 1.5, 4, 4.7, 7, 8, and 38. Discuss your results. Is the mean of the data a good starting point?

(b) Apply the bisection method with starting points -1 and 1. Use additional runs to illustrate manners in which the bisection method may fail to find the global maximum.

(c) Apply fixed-point iterations as $G(x) = \alpha g'(x) + x$, starting from -1, with scaling choices of $\alpha = 1$, 0.64, and 0.25. Investigate other choices of starting values and scaling factors.

(d) From starting values of $(\theta^{(0)}, \theta^{(1)}) = (-2, -1)$, apply the secant method to estimate θ. What happens when $(\theta^{(0)}, \theta^{(1)}) = (-3, 3)$, and for other starting choices?

(e) Use this example to compare the speed and stability of the Newton–Raphson method, bisection, fixed-point iteration, and the secant method.

2. (20 points) The sample is drawn independently from χ^2_d with unknown d.

$$6.9, 5.4, 2.3, 5.0, 6.1, 6.8, 3.5, 1.6, 3.4, 2.9$$

Find the MLE of d using different methods as follows.

(a) Let $\tilde{d}$ denote the method-of-moments estimator of d, and let $\hat{d}$ denote the MLE of d. Derive the variance of $\tilde{d}$ and $\hat{d}$, respectively. Compute the asymptotic relative efficiency (ARE) of $\hat{d}$ relative to $\tilde{d}$, defined as ARE $= var(\tilde{d})/var(\hat{d})$. Finally, plot the ARE as a function of d.

(b) Implement Newton's method to compute the MLE of d and its standard error. The input `x` is the vector of data values. You may include other input arguments such as initial point, tolerance, maximal number of iterations, etc.

```
chisq_mle_newton = function(x) { ...;  list(d = , se = ); }
```

(c) Implement the secant method to find the MLE of d and its standard error.

```
chisq_mle_secant = function(x) { ...;  list(d = , se = ); }
```

(d) Implement the fixed-point method to find the MLE of d and its standard error.

```
chisq_mle_fixedpoint = function(x) { ...;  list(d = , se = ); }
```

3. (10 points) Each female horseshoe crab has a male crab residing in her nest. Satellite males are additional male crabs that reside nearby. Let the response variable be $Y = 1$ if the female crab has at one satellite male and $Y = 0$ otherwise. The predictor variable is the width of the female crab. In the data file, `satell` denotes the number of satellite males and `width` denote the female crab's width.

`http://www.auburn.edu/~zengpen/teaching/STAT-7030/datasets/crab.txt`

Denote the observed data as $\{(y_i, x_i), y_i \in \{0, 1\}, x_i \in \mathbb{R}^p, i = 1, \ldots, n\}$. Consider a logit model,

$$\text{logit}(\pi_i) = \log \frac{\pi_i}{1 - \pi_i} = \beta^T x_i, \quad i = 1, \ldots, n,$$

where $x = (1, \text{width})^T$ in this problem.

(a) Implement the iterative reweighted least squares (IRLS) algorithm to compute the MLE of β and its the standard errors. You may include other input arguments such as initial point, tolerance, maximal number of iterations, etc. You may set the initial value as $\hat{\beta} = 0$.

```
logit_IRLS = function(y, x) { ...; list(b = , se = ); }
```

(b) Apply your implemented code to the data and summarize the results. Compare your estimates and standard errors with those produced by the `glm()` function.

4. (20 points). Using the data set from the previous question, fit a probit model instead.

$$\pi_i = P(Y_i = 1 \mid X = x_i) = \Phi(\beta^T x_i), \quad i = 1, \ldots, n,$$

where $\Phi(\cdot)$ denotes the CDF of $N(0, 1)$. You may set the initial value as $\hat{\beta} = 0$ when computing MLE in the following questions. You may include other input arguments such as initial point, tolerance, maximal number of iterations, etc. It may be necessary to implement backtracking if the first implementation does not work.

(a) Implement Newton's method to compute the MLE of β.

```
probit_newton = function(y, x) { ... }
```

(b) Implement the steepest ascent method to compute the MLE of β.

```
probit_steepest = function(y, x) { ... }
```

(c) Implement the BFGS method to compute the MLE of β.

```
probit_bfgs = function(y, x) { ... }
```

(d) Write a function that consolidates all the previous functions. The argument `method` can take values `newton`, `steepest`, or `bfgs`, and the corresponding method is then used to compute the MLE of β.

```
probit = function(y, x, method) { ... }
```

5. (25 points) Let $y_1, \ldots, y_n$ be an iid sample from a multivariate normal distribution,

$$y_i \sim^{iid} N(\mu, \sigma^2 C), \quad i = 1, \ldots, n,$$

where $y_i \in \mathbb{R}^d$, $\mu \in \mathbb{R}^d$, $\sigma^2 > 0$, and $C = ((1-\rho)I_d + \rho 1_d 1_d^T)$, for $\rho \in (-1, 1)$. Derive an algorithm to compute the MLEs of $\theta = (\mu, \rho, \sigma^2)$. Note that the determinant of C and the inverse of C are given by

$$|C| = (1-\rho)^{d-1}(1+(d-1)\rho),$$
$$C^{-1} = \frac{1}{(1-\rho)} I_d - \frac{\rho}{(1-\rho)[1+(d-1)\rho]} 1_d 1_d^T.$$

(a) Compute the profile log-likelihood function $\tilde{\ell}(\rho)$. Specifically, for fixed ρ, find the MLE of μ and σ^2, and substitute them into the log-likelihood function to obtain $\tilde{\ell}(\rho)$.

(b) Compute the derivative of the profile log-likelihood function $\tilde{\ell}'(\rho)$.

(c) Develop an appropriate algorithm to compute the MLE of $\theta = (\mu, \sigma^2, \rho)$. Your description should include the choice of initial values, updating equations, convergence criterion, and stopping rule.

(d) Implement the proposed algorithm as an R function. The input `y` is a $n \times d$ matrix, with each row representing one observation. The output includes the MLEs of θ.

```
mvnorm_mle = function(y) { ...; list(mu = , sigma2 = , rho = ) }
```

(e) Test your code using the following simulation study. Set $d = 4$, $\mu_* = -1$, $\rho_* = 0.5$, and $\sigma_* = 1.5$.

- Simulate datasets with sample sizes $n = 25, 50, 100, 200, 400$.
- Apply your algorithm to estimate μ, ρ, σ^2.
- Evaluate the performance of the estimators using

$$D(\hat{\theta}, \theta) = \sqrt{(\hat{\mu} - \mu_*)^2 + (\hat{\rho} - \rho_*)^2 + (\hat{\sigma} - \sigma_*)^2}$$

- Repeat the above procedure 100 times for each sample size.
- Plot the mean performance measure $D(\hat{\theta}, \theta)$ as a function of n, with error bars representing the 95% confidence intervals for mean performance.