In this exercise, you will use a Metropolis algorithm to sample from the posterior distribution of a simple Bayesian regression model with Cauchy noise.

The data is a set of \( n \) pairs of real numbers, \((x_1, y_1), \ldots, (x_n, y_n)\). We aim to predict the \( y_i \) from the \( x_i \), using the following model:

\[
y_i \mid x_i, \alpha, \beta, \omega \sim \text{Cauchy}(\alpha + \beta x_i, \omega)
\]

\[
\alpha \sim N(0, 1)
\]

\[
\beta \sim N(0, 1)
\]

\[
\omega \sim \text{Exp}(1)
\]

We observe the \( x_i \) and \( y_i \), and wish to find the posterior distribution of \( \alpha \), \( \beta \), and \( \omega \).

The Cauchy(\( \theta, \omega \)) distribution for a real-valued variable \( y \) has the density function \([\pi \omega (1 + (y - \theta)^2/\omega^2)]^{-1}\).

The standard normal, \( N(0, 1) \), distribution for a real-valued variable \( x \) has density \((2\pi)^{-1/2} \exp(-x^2/2)\).

The \( \text{Exp}(1) \) distribution (exponential with mean one) for a positive real variable \( z \) has density \( \exp(-z) \).

You should sample from the posterior distribution given the following eight data points:

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.1</td>
<td>-1.6</td>
</tr>
<tr>
<td>2.3</td>
<td>1.7</td>
</tr>
<tr>
<td>3.2</td>
<td>2.7</td>
</tr>
<tr>
<td>4.3</td>
<td>3.6</td>
</tr>
<tr>
<td>5.1</td>
<td>4.7</td>
</tr>
<tr>
<td>6.3</td>
<td>5.6</td>
</tr>
<tr>
<td>25.3</td>
<td>27.0</td>
</tr>
<tr>
<td>25.9</td>
<td>23.9</td>
</tr>
</tbody>
</table>

To do this, you should use multivariate Metropolis updates with a proposal distribution in which \( \alpha \), \( \beta \), and \( \omega \) are independent, each drawn from a distribution that is uniform between their value in the current state minus \( u \) and their value in the current state plus \( u \). Here, \( u \) is a tuning parameter of the sampling method. Note that since \( \omega \) must be positive, you should reject any proposal in which \( \omega \) is negative; otherwise you use the standard Metropolis acceptance criterion.

You should try both \( u = 0.1 \) and \( u = 0.5 \). For each, you should simulate 200,000 Metropolis updates from an initial state where \( \alpha = 0 \), \( \beta = 0 \), and \( \omega = 1 \). Try each with four random number seeds to see how results vary randomly. (You should set the seeds explicitly, so that you can reproduce your results.)

For each of \( u = 0.1 \) and \( u = 0.5 \), you should hand in the sample means of \( \alpha \), \( \beta \), and \( \omega \) using iterations from 2000 on, for runs started with each of the four random number seeds.

For the each of \( u = 0.1 \) and \( u = 0.5 \), you should hand in the following plots and other output for the first random seed value only:

- Three plots of the values of \( \alpha \), \( \beta \), and \( \omega \) versus iteration number for the first 5000 iterations.
- Three plots of \( \alpha \) vs. \( \beta \), \( \alpha \) vs. \( \omega \), and \( \beta \) vs. \( \omega \) for every 200th iteration starting at iteration 2000.
- The fraction of Metropolis proposals rejected.

You should also hand in your program code (written in any language you like).

Finally, you should comment briefly on how the choices of \( u = 0.1 \) and \( u = 0.5 \) differ, and on the properties of the posterior distribution.