ECE368: Probabilistic Reasoning
Lab 2: Bayesian Linear Regression

In this lab, we use Bayesian regression to fit a linear model. Consider a linear model of the form

$$z = a_1 x + a_0 + w, \tag{1}$$

where x is the scale input variable, and $\mathbf{a} = (a_0, a_1)^T$ is the vector-valued parameter with unknown entries a_0, a_1, and w is the additive Gaussian noise:

$$w \sim \mathcal{N}(0, \sigma^2), \tag{2}$$

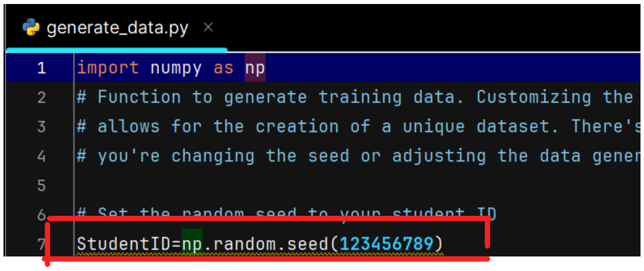
where σ^2 is a known parameter.

Suppose that we have access to a training data set containing N samples $\{x_1, z_1\}, \{x_2, z_2\}, \ldots, \{x_N, z_N\}$. We aim to estimate the parameter $\mathbf{a}$ by finding its posterior distribution. When the training finishes, we make predictions based on new inputs. We consider a Bayesian approach, which models the parameter $\mathbf{a}$ as a zero mean isotropic Gaussian random vector whose probability distribution is expressed as

$$p(\mathbf{a}) = \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \beta & 0 \\ 0 & \beta \end{bmatrix}\right), \tag{3}$$

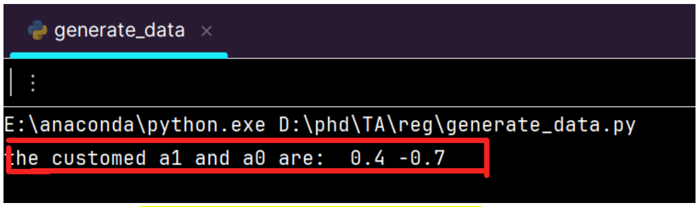
where β is a known hyperparameter. Download reg.zip on Quercus and unzip the file. File training.txt contains the training data: the first column is the inputs;the second column is the targets.
Please answer the questions below and complete regression.py. File util.py contains a few useful functions.

Before proceeding, please run generate_data.py to create your personalized dataset (training.txt). Remember to update the script with your student ID to ensure the dataset is uniquely generated for you. The training dataset is generate by z=a_1x+a_0+w, where a1 and a0 are specific values output by generate_data.py.



```
import numpy as np
# Function to generate training data. Customizing the
# allows for the creation of a unique dataset. There's
# you're changing the seed or adjusting the data gener

# Set the random seed to your student ID
StudentID=np.random.seed(123456789)
```



```
E:\anaconda\python.exe D:\phd\TA\reg\generate_data.py
the customed a1 and a0 are:  0.4 -0.7
```

output of generate_data.py

Questions

1. Express the posterior distribution $p(\mathbf{a}|x_1, z_1, \ldots, x_N, z_N)$ using $\sigma^2, \beta, x_1, z_1, x_2, z_2, \ldots, x_N, z_N$.

2. Let $\sigma^2 = 0.1$ and $\beta = 1$. Based on the posterior distribution obtained in the last question, draw four contour plots corresponding to $p(\mathbf{a})$, $p(\mathbf{a}|x_1, z_1)$, $p(\mathbf{a}|x_1, z_1, \ldots, x_5, z_5)$, and $p(\mathbf{a}|x_1, z_1, \ldots, x_{100}, z_{100})$. In all contour plots, the x-axis represents a_0, and the y-axis represents a_1. The range is set as $[-1, 1] \times [-1, 1]$. In each figure, also draw the true value of $\mathbf{a}$.

3. Suppose that there is a new input x, for which we want to predict the target value z. Write down the distribution of the prediction z, i.e., $p(z|x, x_1, z_1, \ldots, x_N, z_N)$.

4. Let $\sigma^2 = 0.1$ and $\beta = 1$. Suppose that the set of the new inputs is $\{-4, -3.8, -3.6, \ldots, 0, \ldots, 3.6, 3.8, 4\}$. Plot three figures corresponding to the following three cases:

   (a) The predictions are based on one training sample, i.e., based on $p(z|x, x_1, z_1)$.

   (b) The predictions are based on 5 training samples, i.e., based on $p(z|x, x_1, z_1, \ldots, x_5, z_5)$.

   (c) The predictions are based on 100 training samples, i.e., based on $p(z|x, x_1, z_1, \ldots, x_{100}, z_{100})$.

   In all figures, the x-axis is the input, the y-axis is the target, and the range is set as $[-4, 4] \times [-4, 4]$. Each figure should contain three components: 1) the new inputs and the predicted targets; 2) a vertical interval at each predicted target, indicating the range within one standard deviation; 3) the training sample(s) that are used for the prediction. Use plt.errorbar for 1) and 2); use plt.scatter for 3).

References: C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer New York, 2006, pp. 152–159. & K. Murphy, *Machine Learning: A Probabilistic Approach*, MIT Press, 2012, pp. 231–234.