6. *Modeling data with Gaussian processes.* In this question, we will explore modeling of geospatial data via Gaussian Processes. There are two files on the course webpage. One is `gptrain.csv`, and the other is `gptest.csv`. Each row in each file corresponds to a temperature reading at a given weather station in Brazil at 2pm on January 1, 2021. The first column gives the `Latitude` of the station, the second the `Longitude`, and the final the `Temperature` reading in degrees Celsius at a given point in time. We will fit a simple Gaussian Process model on the training data, and see how well it informs us of temperature readings at the remaining stations at this fixed time and date.

   We assume a Gaussian Process governs the data. In particular, we assume the covariance between points x and x' is given by

$$K(x, x') = \exp\left(\frac{-\|x - x'\|_2}{2\sigma^2}\right),$$

   and that the mean function is $m(x) = 20$ degrees Celsius for all locations x. Here $\|x - x'\|_2$ denotes the ℓ_2 norm between raw latitude and longitude coordinates.

   Denote the training locations, i.e. weather station coordinates used in the training, by X_{tr}, and the training responses, i.e. temperature readings at these locations, by y_{tr}; define X_{te}, y_{te} analogously.

   (a) The joint distribution of training and test responses can be written as

$$\begin{pmatrix} y_{\text{tr}} \\ y_{\text{te}} \end{pmatrix} \sim N\left(\begin{bmatrix} 20 \\ \vdots \\ 20 \end{bmatrix}, \begin{bmatrix} K_{\text{tr}} & K_{\text{tr,te}} \\ K_{\text{te,tr}} & K_{\text{te}} \end{bmatrix}\right),$$

   where the block matrix

$$\begin{bmatrix} K_{\text{tr}} & K_{\text{tr,te}} \\ K_{\text{te,tr}} & K_{\text{te}} \end{bmatrix}$$

   is the matrix arising from all pairwise evaluations of the kernel K over all training and testing locations, and its diagonal components K_{tr} and K_{te} are the matrices arising from all pairwise evaluations of the kernel within the training and test sets, respectively.

   Explain how the definition of Gaussian Process allows us to come to this conclusion.

   (b) Define

$$m_{\text{te}} := \begin{bmatrix} 20 \\ \vdots \\ 20 \end{bmatrix} \in \mathbb{R}^{11},$$

   since there are 11 locations in the test set. Using the theory of Gaussian distributions, it can be shown that the distribution of test responses given training locations, training responses, and test locations, follows

$$\begin{aligned} &y_{\text{te}} | y_{\text{tr}}, X_{\text{te}}, X_{\text{tr}} \\ &\sim m_{\text{te}} + N\left(K_{\text{te,tr}} \cdot K_{\text{tr}}^{-1}\left(y_{\text{tr}} - m_{\text{te}}\right), \quad K_{\text{te}} - K_{\text{te,tr}} K_{\text{tr}}^{-1} K_{\text{tr,te}}\right), \end{aligned}$$

   In Bayesian language, this is the posterior predictive distribution. Using X_{tr}, X_{te}, and y_{tr}, compute the posterior mean of y_{te} via matrix multiplication, using the value $\sigma = .866$ as the parameter in the kernel matrix. Report the mean squared error between your computed posterior mean and the true values found in y_{te}.

   (c) In practice, one should almost never invert matrices for numerical stability and computational efficiency reasons. How one can use the function `np.linalg.solve` to compute the posterior predictive covariance with one single call to this function?

   (d) The diagonal of the posterior predictive matrix

$$K_{\text{te}} - K_{\text{te,tr}} K_{\text{tr}}^{-1} K_{\text{tr,te}}$$

   gives the conditional variances of the test responses given X_{tr}, X_{te}, and y_{tr}. Create a rich grid ($\approx$ 5000 points) of test latitudes and longitudes within the range found in the training set. Visualize the variances of the predictive distribution at each grid value using the function `matplotlib.pyplot.imshow`, and plot the locations of the stations found in the training set over this visualization.

   What do you notice about the relationship of the variance of the prediction and the distance of that prediction to a training point?