Problem Set 5

Issued: Monday, April 16th
Due Date (Part 1): Friday, April 25th, 11:59 PM ET
Due Date (Part 2): Friday, May 2nd, 11:59 PM ET

Submission Instructions

1. Add the names of any people you worked with for each the problem, or "Collaborators: none" if you solved the problem completely by yourself.
2. Solutions to all parts of the problem set should be submitted online to Gradescope in PDF. During the submission dialog, make sure to assign all pages spanning your solution for each sub-problem to the marker for that sub-problem. This makes grading much easier.
3. Formatting your problem set in LaTeX will make it easier for us to read; however, any method of generating the PDF is acceptable (including scanning handwritten documents) as long as it is clearly legible.
4. You are not required to submit the analytical code in any case, only the final results. We will not grade you code. In addition, be as clear and precise as possible in your write-up of the homework solutions.
5. Part 1 and Part 2 should be submitted separately and have separate submission deadlines as noted above.

Part 1

The Philippine Archipelago is a fascinating multiscale ocean region. Its geometry is very complex, with multiple straits, islands, steep shelf-breaks, and coastal features, leading to partially interconnected seas and basins. In this part, we will be studying, understanding and navigating through the ocean current flows.

The data set may be found in `OceanFlow.zip`. It consists of the ocean flow vectors for time T from 1 to 100. The flow in the data set is an averaged flow from the surface to either near the bottom or 400m of depth, whichever is shallower. It is thus a 2D vector field. The files `*u.csv` contain the horizontal components of the vectors, while the files `*v.csv` contain the vertical component. The numbers in the file names indicate the time. For instance, files `24u.csv` and `24v.csv` contain the information of the flow at time 24. The file `mask.csv`, if needed, contains a a 0-1 matrix identifying land and water.

Additional info and units: The data has been collected in January 2009. Multiply the flow values by 25/0.9 to get a unit of cm/second (cmps). The time interval between the data snapshots is 3hrs. The grid spacing used is 3km. The matrix index $(0, 0)$ will correspond in this problem to the coordinate (0km,0km), or the ***bottom, left*** of the plot. For simplicity, we will not be using longitudes and latitudes in this problem.

The data has been provided by the MSEAS research group at MIT (`http://mseas.mit.edu/`). The flow field is from a data-assimilative multiresolution simulation obtained using their MSEAS

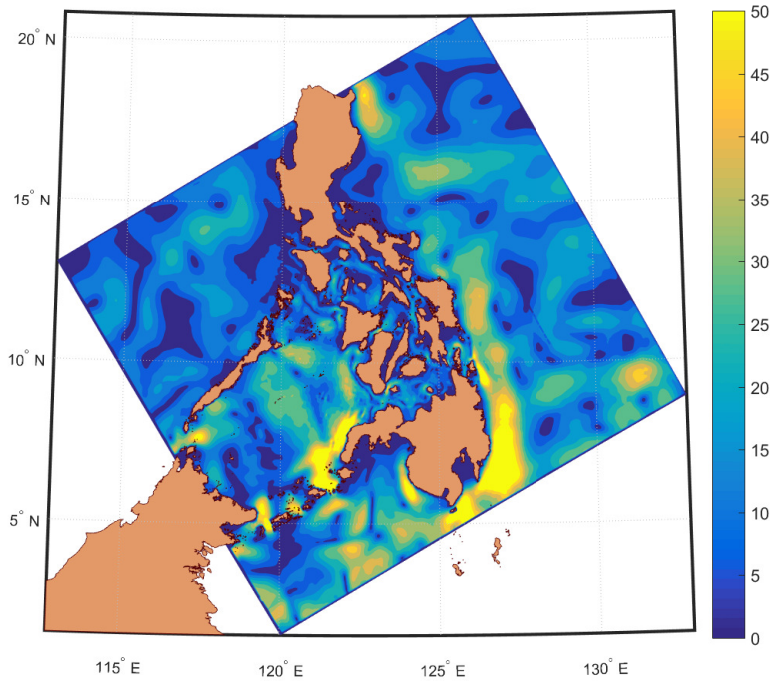

Figure 1: Snapshot of the ocean flow speed in the Philippine Archipelago.

primitive-equation ocean modeling system. It simulates tidal flows to larger-scale dynamics in the region, assimilating a varied set of gappy observations.

Problem 4.1: [30pts] Flows and correlation.

We first study spatial correlations in the ocean flow.

(a) Visualize the average flow (averaged over all times, and not location) as a 2-D vector field. What are the strongest flow currents that run in the archipelago? Again, remember, the matrix index $(0, 0)$ will correspond in this problem to the coordinate (0km,0km), or the ***bottom, left*** of the plot.

(b) Visualize the average speed of the flow (i.e. for each grid point, take the average of speed) as a 2-D graph. Do you notice any areas with high average speed but low average flow, or vice versa? Why might that be?

(c) Consider the spatial correlation of this dataset. Take two points: $(140, 115)$ and $(400, 400)$. For each point, plot the average correlation of ocean flow (choose either speed or both horizontal and vertical flow) with other points as a function of grid distance/L1-distance (consider only points within a reasonable distance, e.g. distance < 100 or 200). How would you describe the spatial correlations in ocean flow? Explain why Gaussian Processes would or would not be a good model for the correlation in ocean flow.

Part 2

Problem 4.2: [20pts] Predicting trajectories.

The goal of this problem is to simulate the trajectory of a particle moving in the flow.

(a) We assume that a particle in the ocean, with certain coordinates, will inherit the velocity correponding to the flow at those coordinates. Implement a procedure to track its position and movement caused by the time-varying flow. Explain the procedure, and show that it works by providing examples and plots.

(**Hint / suggested approach:** The data provides a discretization of the ocean flow. The particles will however be moving on a continuous surface. For simplicity, let us assume that the surface is the plane $\mathbb{R}^2$. The data can be seen to provide flow information at integer points, namely at (m, n) for m and n integers. Divide the continuous surface into squares, in such a way that each square contains a unique data point. One way to achieve this is to assign to every point in the surface the closest data point. For instance, given $(x, y) \in \mathbb{R}^2$, this consist of rounding both x and y to the closest integer. You may then suppose that each square has the same flow information as the data point it contains.

Now take a particle at (x, y) in a certain square. The flow in the square will displace it at the registered velocity. Once the particle moves out of this square, it is then governed by the new squares' flow information.)

(b) A (toy) plane has crashed in the Sulu Sea at $T = 0$. The exact location is unknown, but data suggests that the location of the crash follows a Gaussian distribution with mean $(100, 350)$ (namely $(300km, 1050km)$) with covariance matrix $\sigma^2 I$. The debris from the plane have been carried away by the ocean flow. You are about to lead a search expedition for the debris. Where would you expect the parts to be at 48hrs, 72hrs, 120hrs? Study the problem varying the variance of the Gaussian distribution. Either pick a few variance samples or sweep through the variances if desired. (**Hint:** Sample particles and track their evolution.)

Recommended Python Packages:

Some packages that might be useful for implementing the coding parts of this assignment (ask classmates on Piazza for non-python suggestions) :

```
matplotlib.pyplot.quiver
matplotlib.pyplot.imshow
numpy.random.multivariate_normal
```

Problem 4.3: [optional for all] Gaussian processes.

(a) Different kernels or covariance functions can have different properties for Gaussian process regression. Consider the (noise-free) squared exponential/RBF covariance function:

$$\kappa(x_i, x_j) = \sigma^2 \exp\left(\frac{-(x_i - x_j)^2}{2\ell^2}\right)$$

What is the effect of changing the signal variance (σ^2) and lengthscale (ℓ)?

(b) Prediction with Gaussian processes can become computationally very expensive when we have lots of observations. Why?

(c) How could we make this computation more efficient? Suggest a solution and an algorithm (including pseudocode).

Hint: there are many possibilities, e.g., shrinking the data or using the Sherman-Morrison-Woodbury formula. If you follow any of these or your own idea, say how exactly you would do it and why, and how you would ensure to not lose too much prediction quality. Your predictions will be approximations to predictions using the full, expensive Gaussian process. You may use the form of prediction with noisy observations.

As a further hint, if you want to use the Sherman-Morrison-Woodbury matrix inversion lemma, here is a form that is useful for the problem:

$$(Z + UWV^\top)^{-1} = Z^{-1} - Z^{-1}U(W^{-1} + V^\top Z^{-1}U)V^\top Z^{-1},$$

where $Z \in \mathbb{R}^{n\times n}$, and $U, V \in \mathbb{R}^{n\times m}$.