PM2.5 Air Pollution Prediction using Gaussian Processes and Gaussian Mixture Models

1 Goal

In this project, we aim to help a city predict and audit the concentration of fine particulate matter (PM2.5) per cubic meter of air. In an initial phase, the city has collected preliminary measurements using mobile measurement stations. The goal is now to develop a pollution model that can predict the air pollution concentration in locations without measurements. This model will then be used to determine particularly polluted areas where permanent measurement stations should be deployed.

A pervasive class of models for weather and meteorology data are **Gaussian Processes (GPs)**. We are implementing **Gaussian Process regression** in order to model air pollution and try to predict the concentration of PM2.5 at previously unmeasured locations.

2 Problem Set-up and Challenges

2.1 Features

- **Inputs:** Coordinates (X, Y) of the city map
- **Target:** PM2.5 pollution concentration at a given location

2.2 Challenges

- **Model Selection:** Determination of the right kernel and hyperparameters is key for GP performance.
- **Large Scale Learning:** As the number of observations increases, the computational cost of GPs grows exponentially. The posterior complexity is $O(n^3)$.
- **Asymmetric Cost:** Cost-sensitive learning is implemented with a loss function that treats different types of errors differently:

$$\ell_W(f(x),\hat{f}(x)) = (f(x)-\hat{f}(x))^2 \cdot \begin{cases} 25, & \text{if } f(x) \leq \hat{f}(x) \text{ (underprediction)} \\ 1, & \text{if } f(x) \leq 1.2 \cdot f(x) \text{ (slight overprediction)} \\ 10, & \text{if } 1.2 \cdot f(x) \leq \hat{f}(x) \text{ (significant overprediction)} \end{cases}$$

3 Approach and Results

For the model selection challenge, I decided to use a custom kernel made of a combination of Matern kernel, sinusoidal kernel, and RBF kernel.

Regarding the large-scale learning issue, I decided to first fit a Gaussian mixture model (GMM) on the data, using 25 kernels. Then, instead of fitting a single Gaussian Process (GP) on all the points, I decided to fit one Gaussian Process per kernel, hence dividing the training time by 25^2. For the predictions, I tried to use something similar to soft GMM by averaging predictions of each individual Gaussian Process based on cluster membership probability. However, a "hard" GMM worked better, where I would just take the prediction of the GP associated with the closest cluster.

Finally, for the asymmetric cost, to make a prediction for point x, I followed the approach of minimizing the expected squared loss as described below:

Given the conditional distribution:

$$P(y \mid x) = \mathcal{N}(y; \mu_{Y|X=x}, \sigma^2_{Y|X})$$

The cost function used was:

$$C(y,a) = c_1 \cdot \max(y-a,0) + c_2 \cdot \max(a-y,0)$$

where c_1 and c_2 represent the cost of underestimation and overestimation, respectively.

The prediction a^* that minimizes the expected cost is:

$$a^* = \mu_{Y|X=x} + \sigma_{Y|X}\Phi^{-1}\left(\frac{c_1}{c_1+c_2}\right)$$

This provides a bias-adjusted prediction, accounting for the asymmetric costs of under- and over-estimating the PM2.5 concentration at each location.

With this approach, I was able to reach the hard baseline for prediction costs on test data, achieving an average penalty score of 8.61, when the hard baseline expected a prediction penalty score lower than 21.84.