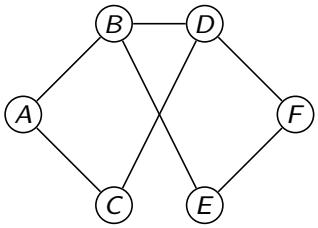
Machine Learning and Deep Learning II

Homework 2

Due date: 11:59 PM, October 19, 2022

1. [10 pts] Let $\mathbf{X} = \{A, C, D\}$, $\mathbf{Y} = \{B, E, F\}$. Consider $P(\mathbf{x}, \mathbf{y})$ modeled using an MRF with an undirected graph H shown below:



$P(\mathbf{x}, \mathbf{y})$ is made only of pairwise potential functions

$$P(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C, \mathbf{y}_C) = \frac{1}{Z} \psi_{A,B}(a, b) \psi_{A,C}(a, c) \cdots \psi_{E,F}(e, f).$$

Derive $P(\mathbf{y} \mid \mathbf{x})$ from the above expression. In your derivation, you must fully express the partition function for $P(\mathbf{y} \mid \mathbf{x})$ and must remove potentials that are irrelevant to $P(\mathbf{y} \mid \mathbf{x})$.

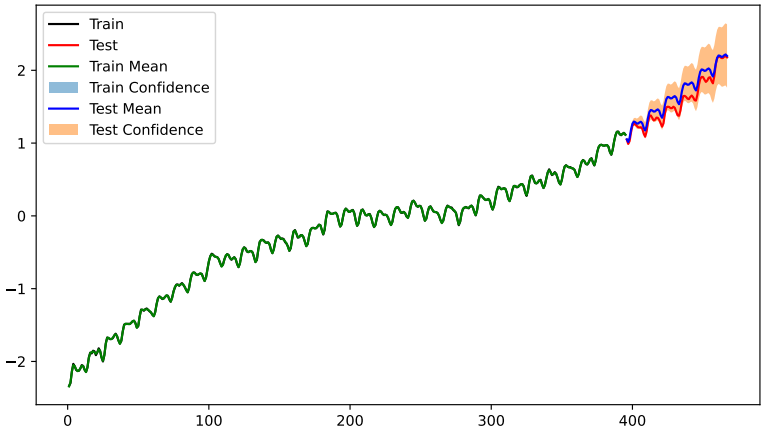
2. [5 pts] (Reading homework) Read XGBoost paper `https://arxiv.org/abs/1603.02754` up to Section 3.1. Please indicate that you've read it.

3. [5 pts] (Reading homework) Read GPML Book `http://gaussianprocess.org/gpml/chapters/RW.pdf` about Mauna Loa Atmospheric Carbon Dioxide (pages 118–122). Please indicate that you've read it.

4. [5 pts] (Reading homework) Read 'How to Use t-SNE Effectively' `https://distill.pub/2016/misread-tsne/`. Please indicate that you've read it.

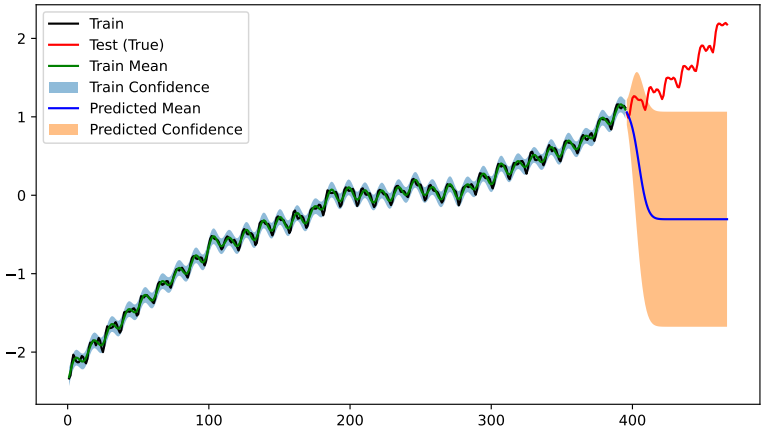
5. [25 pts] Train a Gaussian process for atmospheric methane abundance using *your own custom kernel*. Here, training means learning parameters for mean and covariance function. The data can be found in `https://gml.noaa.gov/ccgg/trends_ch4/`. Use column '`average`' for Y and rows as 'X'. We will use `GPytorch` where you can find a tutorial (`https://docs.gpytorch.ai/en/stable/examples/01_Exact_GPs/Simple_GP_Regression.html`) about how to perform training, predicting, and plotting.

   The main task is to design a custom kernel with learnable parameters using composition like addition and multiplication. A successfully trained GP would output something similar to a plot shown below, which illustrates appropriately captured trends and periodicity (predicting atmospheric methane abundance for the last 6 years)



   A naive combination of `GPytorch` kernels will likely fail to capture the patterns in the training data. (Please check out `gp.py` for a skeletal code.)[1]

```
self.covar_module = ScaleKernel(RBFKernel()) + ScaleKernel(RBFKernel() * PeriodicKernel())
```



   Better training is possible by guiding how kernels should behave based on our domain expertise. For example, you can specify a feasible range of period for `PeriodicKernel` by specifying an interval, e.g.,

```
PeriodicKernel(period_length_constraint=Interval(2, 6))
```

   for period within 2 to 6 months. Similarly, you can specify constraints or priors for the parameters of `RBFKernel`.

   Your tasks are: (i) to write `self.covar_module` appropriately so as for a trained GP to forecast atmospheric methane abundance reasonably well; (i) report training loss and a plot demonstrating mean prediction and confidence region.

   Submit a python file (`gp.py`) together with its resulting plot.

[1] Note that in `GPytorch`, a scale kernel separately exists to scale an existing kernel by wrapping it. Please consult the documentation for `GPytorch`.