**Paper:** *Consistent Online Gaussian Process Regression Without the Sample Complexity Bottleneck (*<https://arxiv.org/pdf/2004.11094.pdf>*)*

**Author:** Alec Koppel, Hrusikesha Pradhan, Ketan Rajawat

``pog.m’’ is the main MATLAB file which computes the test error performance while training over streaming data. You need to supply the data set on which you want to see the performance of POG algorithm (see above paper).

``dkmppf\_hellinger.m’’ is the compression algorithm, which takes input as the current dictionary, training data, noise prior and compression budget.

``Hellinger\_distance.m’’ calculates the Hellinger distance

``kRBF.m’’ code is the kernel function which takes theta as the parameter. Here we use theta for every dimension of the input data (known as Automatic Relevance Determination (ARD) ).

``runningaverage.m’’ code takes input as the size of the running sample for averaging and returns the running average.

``’neglog\_marginallikelihood.m’’ code calculates the negative of log of marginal likelihood and it is used for the hyperparameter optimisation.

We use ``fmincon’’ a MATLAB inbuilt function for hyperparameter (theta) optimisation. In our model we have hyperparameters as:

1. Bandwidth of gaussian kernel (size same as dimension of data (dim))
2. Amplitude of Gaussian kernel
3. Noise prior

Thus these three above are the hyperparameters, which accounts for the total number of hyperparameters to be “dim+2”. Considering a small subset of training data we have done hyperparameter optimisation and the same set of hyperparameters are used for the entire training process.

Few parameters needs to be set specific to every data set like compression budget (Keps) and size of the subset of training data used for hyperparameter optimisation (hyp\_trainsize).

In the folder we have few data sets:

1. Kin40k
2. Lidar
3. Boston housing data set
4. Abalone data set
5. Pendulum data
6. Pumadyn data
7. Sinc\_3d data (synthetic data)