Geodesic GP for Big Datasets

(MATLAB Code Instruction)

Sam D. Tajbakhsh

June 7, 2013

The following is a brief description on how to use the provided MATLAB code for Geodesic Gaussian Process modeling of *big datasets*. This software is suggested when $N > 1500$ or so, for smaller N, the "small N" code can be used instead. This code is supplementary material for the paper titled "Geodesic Gaussian Processes for the Reconstruction of Free-Form Surface" by Enrique del Castillo, Bianca M. Colosimo and Sam D. Tajbakhsh.

To be able to properly run the code, you need the following MATLAB toolboxes:

- Optimization Toolbox
- Global Optimization Toolbox
- Statistics
- NURBS Toolbox

Also you need to copy the following packages to the current directory of your MATLAB software:

- Dimensionality Reduction
- MATALB Mesh
- Sparseinv (by: Tim Davis)

The use of the GGP *big datasets* code, is similar to the original code for small datasets. We have included a driver function, named "driverProgram_BigN" to distinguish it from the original driver function. This function requires a data file, named "fusionData.mat" which is the file used in the paper and contains the 9635 scanned data points coordinates (the file can easily be replaced with any other dataset). The only input to the "driverProgram_BigN" function is a logical value (isomap) which can be either true or false. If true, the routine will use ISOMAP parametrization; otherwise, it uses ARAP parametrization.

As mentioned above, the beginning of this function is similar to original driver function while the parameter estimation functions (MLE_EBLUP3.m) and prediction function (MSEPredictions3.m) are different which are the main focus of this guide. **It is suggested that all the "BIG N" files be placed in the same folder as those programs used for small N, since they require common functions.**

MLE_EBLUP3.m Function

This function estimates the parameters of the model for big datasets using Sang and Huang (2012) paper titled "A full scale approximation of covariance functions for large spatial datasets" and returns model's estimated parameters, inverse of the covariance matrix and the values of the parameters of Sang and Huang 2012 paper mentioned below:

- *gamma* is the tapering parameter. The covariance of any two points farther than the gamma value are set to zero by tapering function.
- *sStar* is the matrix containing the locations of the knots ($n \times 2$ in this case). Here, These locations are over a square grid of size ($\sqrt{m} \times \sqrt{m}$).
- *indcTaper* is an indicator variable ($\in \{1, 2, 3\}$) which determines the type of tapering function to be used. The value of indcTaper equal to 1, 2 or 3 result into Spherical, Wendland-1 and Wendland-2 tapering functions, respectively.

These parameters can be set in the lines 90-100 of the "MLE_EBLUP3.m" function. The inverse of the covariance matrix is stored since calculation of the inverse of the big covariance matrix is challenging and one of the contributions of this code.

MSEPredictions3.m Function

This function predicts the value of the response using the GP-based prediction formula at the new locations $X0$. The function takes the model estimated parameters, Sang and Huang parameters, X, Y and $X0$ as inputs. The predicted response at $X0$ is Yp and the variance of predictions are in stored in $sigma2p$. There is no tuning parameter for this function.