Gaussian Process Modeling of Large-Scale Terrain

Problem

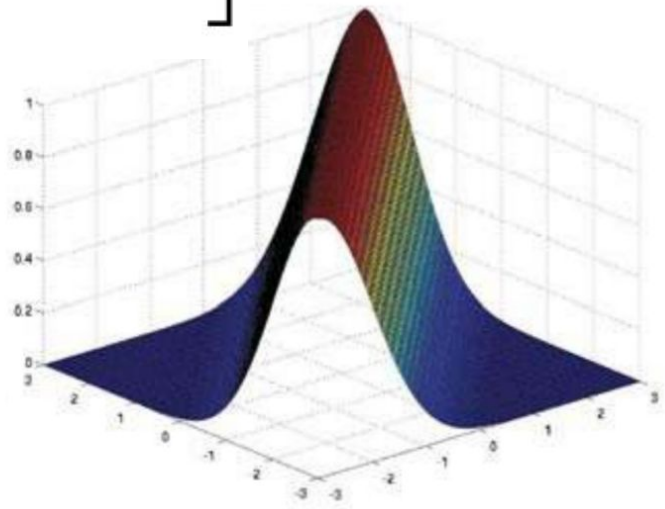
Given set of points (x, y, z), establish a model to predict $z = f(x, y)$.

Gaussian Process

- Determined by mean and covariance (kernel) functions
- Mean function can be assumed to be zero (in this case)
- Kernel function models correlation between input variables

Squared exponential kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma (\mathbf{x} - \mathbf{x}')\right]$$

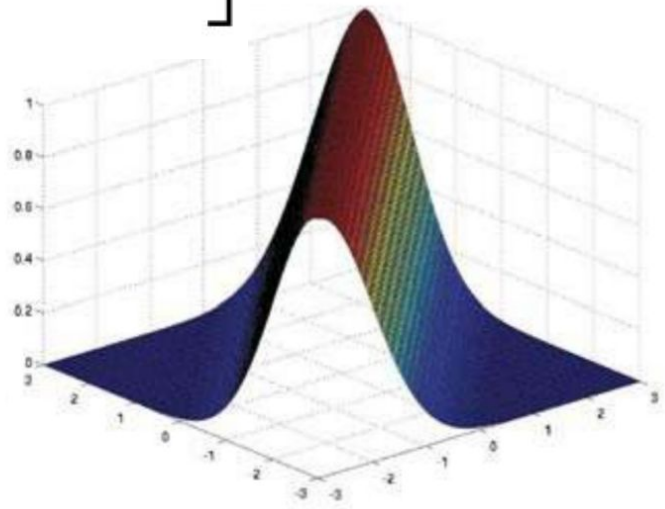


(a) SQEXP kernel behavior

Squared exponential kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma (\mathbf{x} - \mathbf{x}')\right]$$

stationary

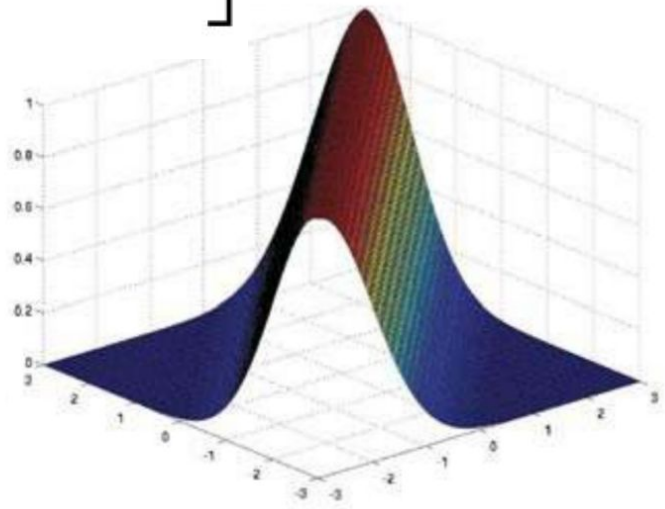


(a) SQEXP kernel behavior

Squared exponential kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma (\mathbf{x} - \mathbf{x}')\right]$$

exponential decay as distance increases

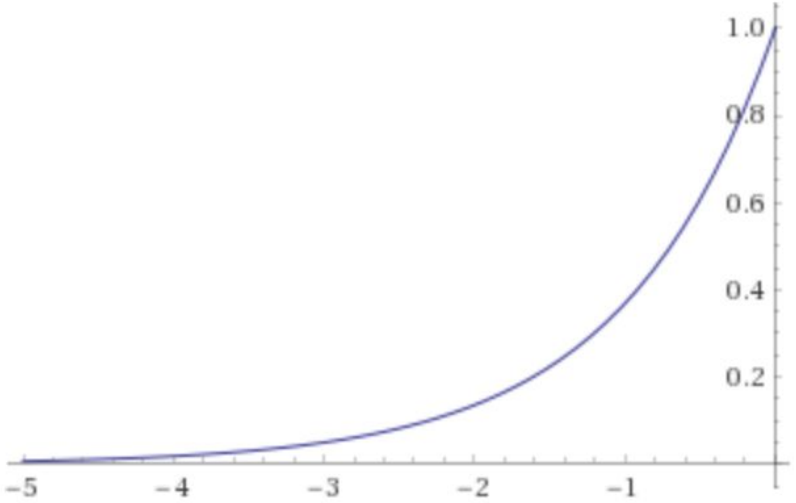


(a) SQEXP kernel behavior

Squared exponential kernel

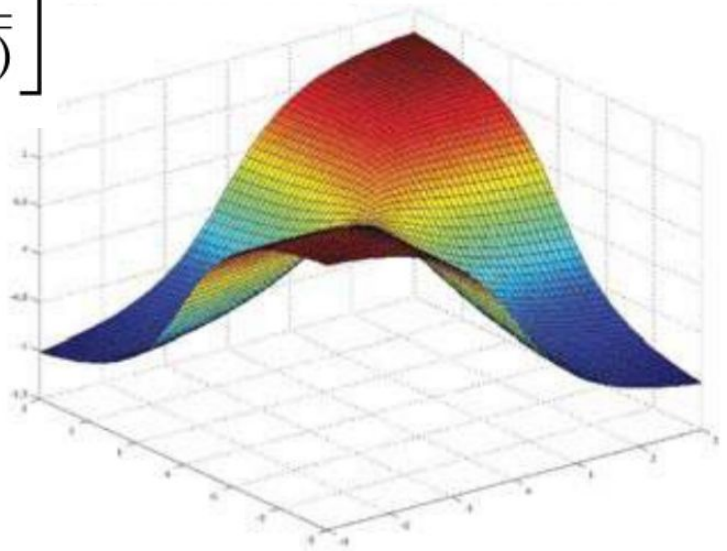
$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \Sigma (\mathbf{x} - \mathbf{x}')\right]$$

exponential decay as
distance increases



Neural Network Kernel

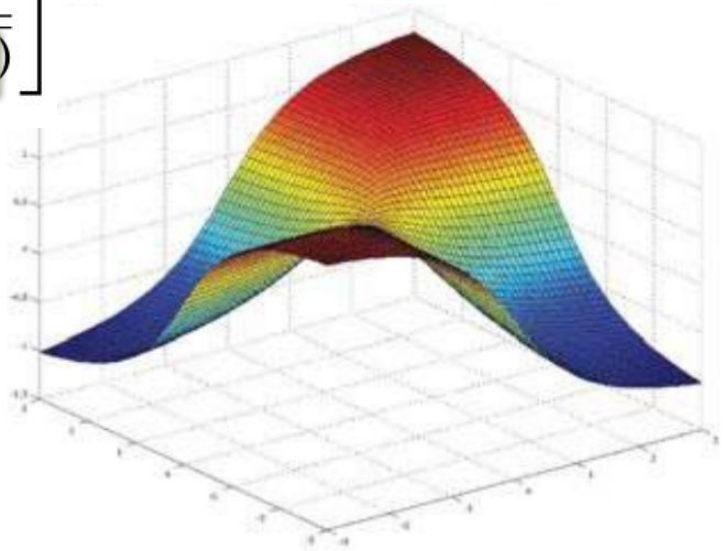
$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \arcsin\left[\frac{\beta + 2\mathbf{x}^T \Sigma \mathbf{x}'}{\sqrt{(1 + \beta + 2\mathbf{x}^T \Sigma \mathbf{x})(1 + \beta + 2\mathbf{x}'^T \Sigma \mathbf{x}')}}\right]$$



Neural Network Kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \arcsin\left[\frac{\beta + 2\mathbf{x}^T \Sigma \mathbf{x}'}{\sqrt{(1 + \beta + 2\mathbf{x}^T \Sigma \mathbf{x})(1 + \beta + 2\mathbf{x}'^T \Sigma \mathbf{x}')}}\right]$$

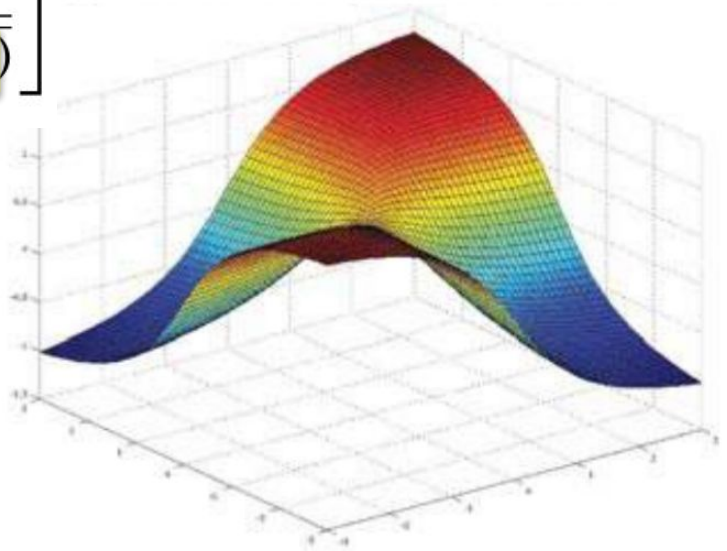
non-stationary



Neural Network Kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \arcsin\left[\frac{\beta + 2\mathbf{x}^T \Sigma \mathbf{x}'}{\sqrt{(1 + \beta + 2\mathbf{x}^T \Sigma \mathbf{x})(1 + \beta + 2\mathbf{x}'^T \Sigma \mathbf{x}')}}\right]$$

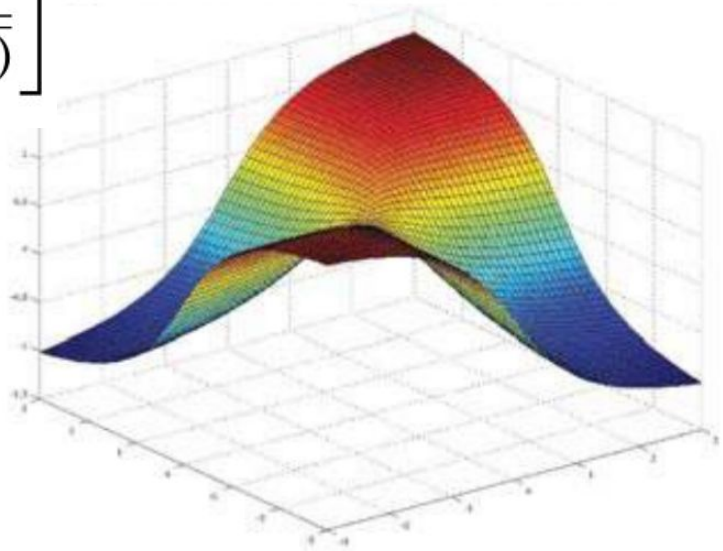
Correlation increases with distance (until saturation)



Neural Network Kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \arcsin\left[\frac{\beta + 2\mathbf{x}^T \Sigma \mathbf{x}'}{\sqrt{(1 + \beta + 2\mathbf{x}^T \Sigma \mathbf{x})(1 + \beta + 2\mathbf{x}'^T \Sigma \mathbf{x}')}}\right]$$

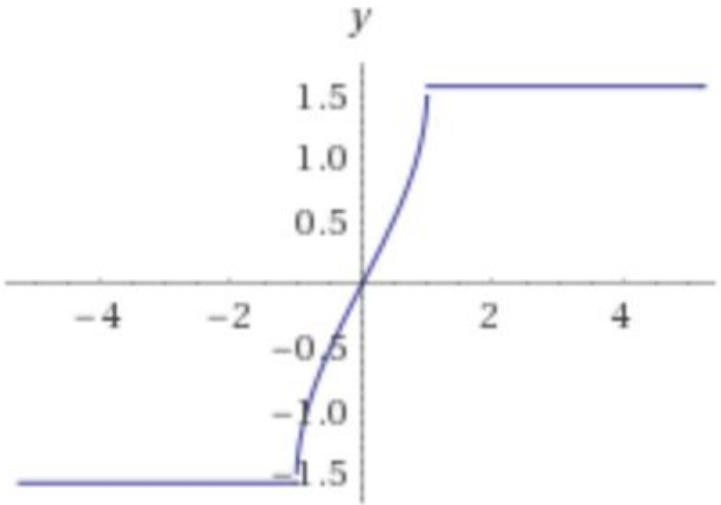
Bounded at high and low inputs



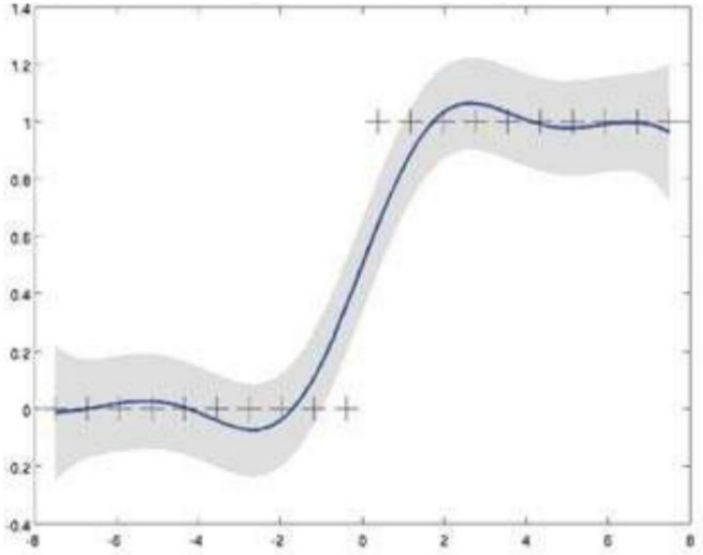
Neural Network Kernel

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \arcsin\left[\frac{\beta + 2\mathbf{x}^T \Sigma \mathbf{x}'}{\sqrt{(1 + \beta + 2\mathbf{x}^T \Sigma \mathbf{x})(1 + \beta + 2\mathbf{x}'^T \Sigma \mathbf{x}')}}\right]$$

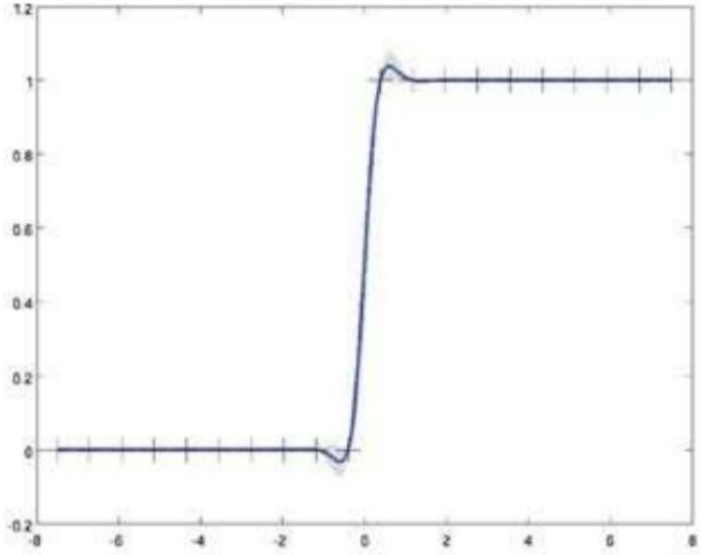
Bounded at high and low inputs



Comparison



(a) GP modeling of data using SQEXP kernel



(b) GP modeling of data using NN kernel

Non-stationary kernel?

Data will become skewed at large distance from origin

Non-stationary kernel?

Moving window

(implemented via k-d tree nearest neighbors)

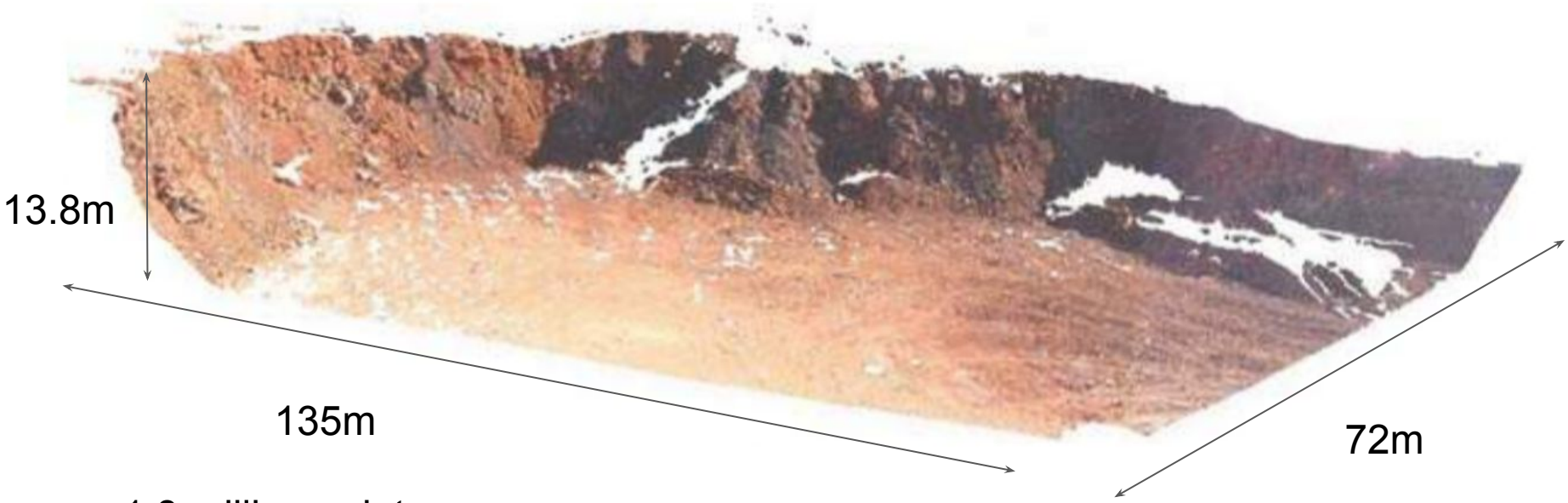
Moving window

1. Necessary for kernel effectiveness

Moving window

1. Necessary for kernel effectiveness
2. Reduces computational overhead

Tom Price Dataset



~1.8 million points

Tom Price Dataset

Table II. Kernel performance: Tom Price data set (1,806,944 data points over 135×72 m^2, 3,000 training data, 10,000 test points for MSE).

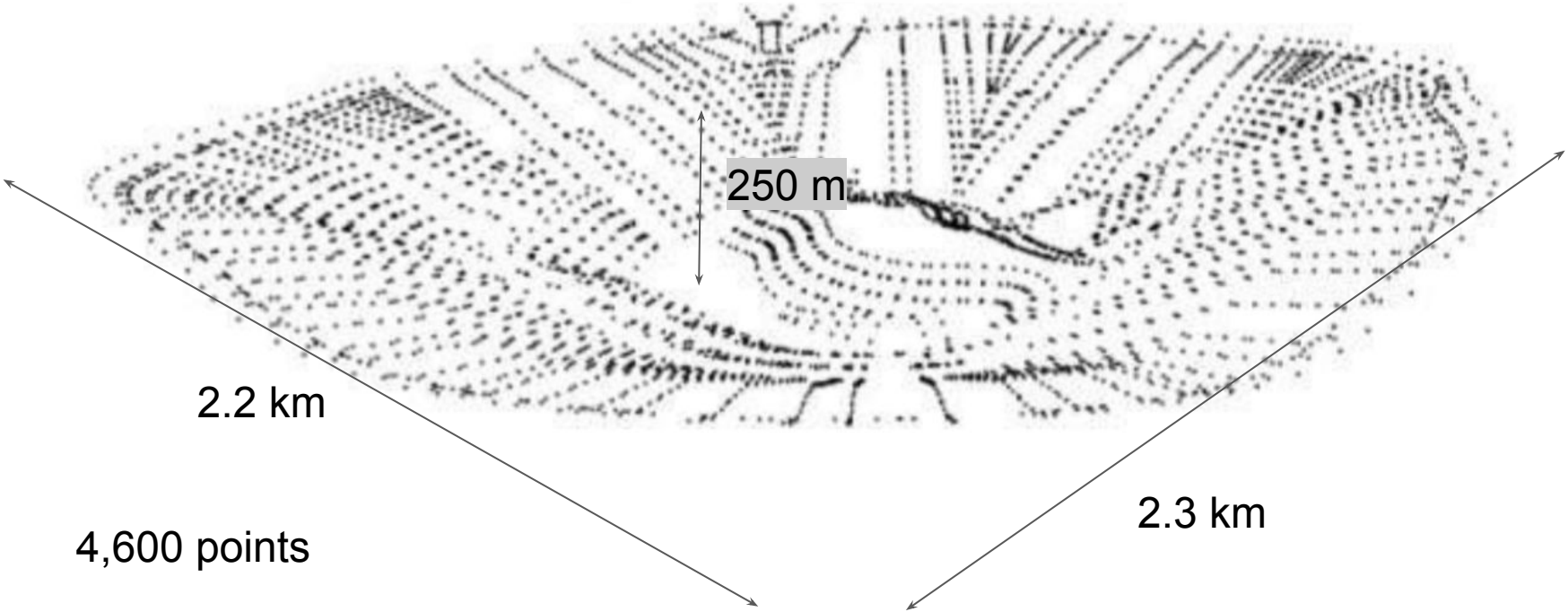
Kernel	MSE (m^2)
SQEXP	0.0136
NN	0.0137

Tom Price Dataset

Table V. Tom Price data set 10-fold cross validation with uniform sampling. Comparison of interpolation techniques.

	1,000 test data per fold		10,000 test data per fold	
Interpolation method	Mean MSE (m^2)	Std. dev. MSE (m^2)	Mean MSE (m^2)	Std. dev. MSE (m^2)
GP neural network	0.0107	0.0012	0.0114	0.0004
GP squared exponential	0.0107	0.0013	0.0113	0.0004
Nonparametric linear	0.0123	0.0047	0.0107	0.0013
Nonparametric cubic	0.0137	0.0053	0.0120	0.0017
Nonparametric biharmonic	0.0157	0.0065	0.0143	0.0019
Nonparametric mean of neighborhood	0.0143	0.0010	0.0146	0.0007
Nonparametric nearest neighbor	0.0167	0.0066	0.0149	0.0017
Parametric linear	0.0107	0.0013	0.0114	0.0005
Parametric quadratic	0.0110	0.0018	0.0104	0.0005
Parametric cubic	0.0103	0.0018	0.0103	0.0005
Triangulation linear	0.0123	0.0046	0.0107	0.0013
Triangulation cubic	0.0138	0.0053	0.0120	0.0017

Kimberlite Mine Dataset



Kimberlite Mine Dataset

Table III. Kernel performance: Kimberlite Mine data set (4,612 points spread over 2.17×2.28 km^2).

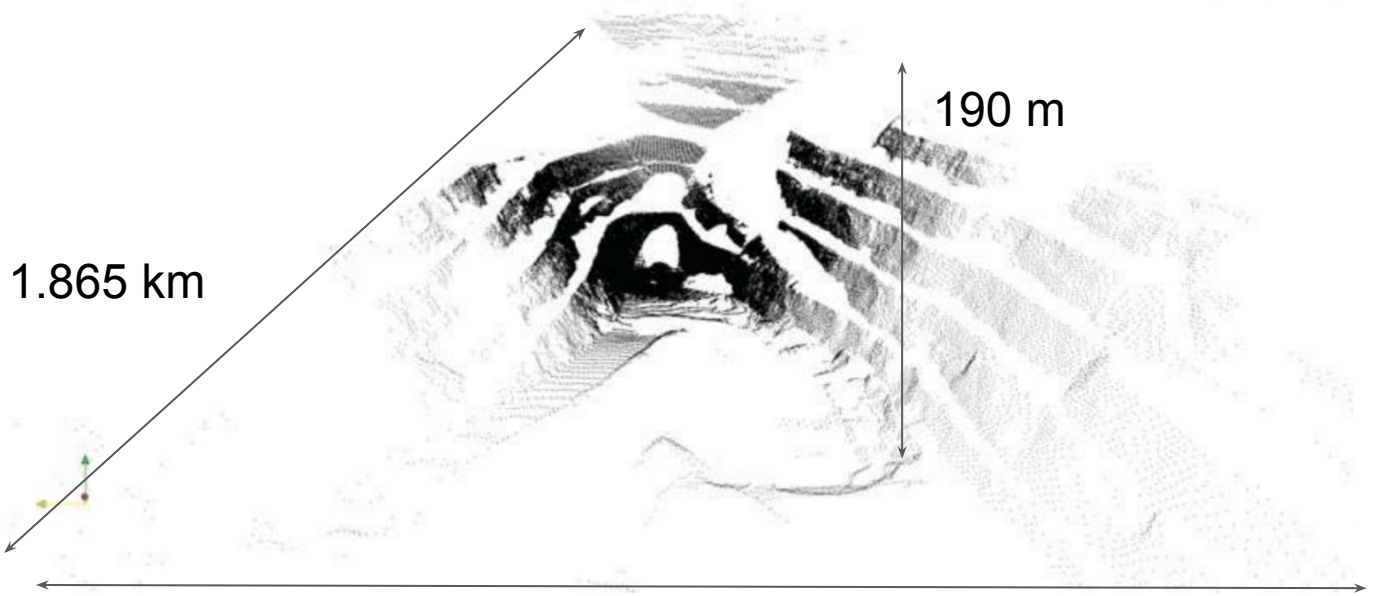
Kernel	Number of training data	MSE (m^2)
SQEXP	1,000	13.014 (over 3,612 points)
NN	1,000	8.870 (over 3,612 points)
SQEXP	4,512	4.238 (over 100 points)
NN	4,512	3.810 (over 100 points)

Kimberlite Mine Dataset

Table VI. Kimberlite Mine data set 10-fold cross validation with uniform sampling. Comparison of interpolation techniques.

Interpolation method	Mean MSE (m^2)	Std. dev. MSE (m^2)
GP neural network	3.9290	0.3764
GP squared exponential	5.3278	0.3129
Nonparametric linear	5.0788	0.6422
Nonparametric cubic	5.1125	0.6464
Nonparametric biharmonic	5.5265	0.5801
Nonparametric mean of neighborhood	132.5097	2.9112
Nonparametric nearest neighbor	20.4962	2.5858
Parametric linear	43.1529	2.2123
Parametric quadratic	13.6047	0.9047
Parametric cubic	10.2484	0.7282
Triangulation linear	5.0540	0.6370
Triangulation cubic	5.1091	0.6374

West Angelas Dataset



West Angelas Dataset

Table IV. Kernel performance: West Angelas data set (534,460 data points over 1.865×0.511 km^2, 3,000 training data, 106,292 test points for MSE).

Kernel	MSE (m^2)
SQEXP	0.590
NN	0.019

West Angelas Dataset

Table VII. West Angelas data set 10-fold cross validation with uniform sampling. Comparison of interpolation techniques.

	1,000 test data per fold		10,000 test data per fold	
Interpolation method	Mean MSE (m^2)	Std. dev. MSE (m^2)	Mean MSE (m^2)	Std. dev. MSE (m^2)
GP neural network	0.0166	0.0071	0.0219	0.0064
GP squared exponential	0.4438	1.0289	0.7485	0.7980
Nonparametric linear	0.0159	0.0075	0.0155	0.0021
Nonparametric cubic	0.0182	0.0079	0.0161	0.0021
Nonparametric biharmonic	0.0584	0.0328	0.1085	0.1933
Nonparametric mean of neighborhood	0.9897	0.4411	0.9158	0.0766
Nonparametric nearest neighbor	0.1576	0.0271	0.1233	0.0048
Parametric linear	0.1019	0.0951	0.0927	0.0173
Parametric quadratic	0.0458	0.0130	0.0390	0.0059
Parametric cubic	0.0341	0.0109	0.0288	0.0038
Triangulation linear	0.0162	0.0074	0.0157	0.0022
Triangulation cubic	0.0185	0.0078	0.0166	0.0023

My questions:

“A data set of N (for example, 1 million) points may be split into, for instance, n_train (say, 3,000) training points and n_test (say, 100,000) test points, and the remaining n_eval(897,000) are evaluation points. The training points are those over which the GP is learned, the test points are those over which the MSE is evaluated, and the evaluation points together with the training points are used to make predictions at the test points.”

My questions:

“As expected, it was found that the combined strategy performed best. The gradient-based optimization requires a good starting point or the optimizer may get stuck in local minima. The stochastic optimization (simulated annealing) provides a better chance to recover from local minima as it allows for “jumps” in the parameter space during optimization. Thus, the combination first localizes a good parameter neighborhood and then zeros-down on the best parameters for the given scenario.”