Gaussian process regression

Model:  \( Y(s) = X(s)\beta + w(s) + \epsilon(s). \)

1. \( Y(s) \) response variable at 'location' \( s \).
2. \( \beta \) regression effects. \( X(s) \) covariates at \( s \).
3. \( w(s) \) structured (space-time correlated) Gaussian process with 0 mean.
4. \( \epsilon(s) \) unstructured (independent) Gaussian measurement noise.
Gaussian model

Model: \( Y(s) = X(s)\beta + w(s) + \epsilon(s) \).
Data at \( n \) 'locations': \( Y = (Y(s_1), \ldots, Y(s_n))' \).
Main goals are:
- Parameter estimation
- Prediction
Likelihood for parameter estimation:

\[
I(\mathbf{Y}; \beta, \theta) = -\frac{1}{2} \log |\mathbf{C}| - \frac{1}{2} (\mathbf{Y} - \mathbf{X} \beta)' \mathbf{C}^{-1} (\mathbf{Y} - \mathbf{X} \beta)
\]

\[
\mathbf{C}(\theta) = \mathbf{C} = \Sigma + \tau^2 \mathbf{I}_n
\]

\(
\text{Var}(\mathbf{w}) = \Sigma, \ \text{Var}(\epsilon(s_i)) = \tau^2 \text{ for all } i.
\)

\(\theta\) include parameters of the covariance model.
Maximum likelihood

\[ \hat{\theta}, \hat{\beta} = \arg\max_{\theta, \beta} \{ l(Y; \beta, \theta) \}. \]
Analytical derivatives

Formulas for matrix derivatives.

\[
\begin{align*}
Q(\theta) &= C^{-1} \\
\hat{\beta} &= [X'QX]^{-1}X'QY, \\
Z &= Y - X\hat{\beta} \\
d\log |C| = d\theta_r &= \text{trace}(Q \frac{dC}{d\theta_r}) \\
dZ'QZ = d\theta_r &= -Z'Q \frac{dC}{d\theta_r} QZ.
\end{align*}
\]
Score and Hessian for $\theta$

\[
\frac{dl}{d\theta_r} = -\frac{1}{2} \text{trace}(Q \frac{dC}{d\theta_r}) + \frac{1}{2} Z' Q \frac{dC}{d\theta_r} Q Z,
\]

\[
E \left( \frac{d^2l}{d\theta_r d\theta_s} \right) = -\frac{1}{2} \text{trace}(Q \frac{dC}{d\theta_s} Q \frac{dC}{d\theta_r}).
\]
Updates for each iteration

\[ Q = Q(\theta_p) \]
\[ \hat{\beta}_p = [X'QX]^{-1}X'QY, \]
\[ \hat{\theta}_{p+1} = \hat{\theta}_p - E \left( \frac{d^2l(Y; \hat{\beta}_p, \hat{\theta}_p)}{d\theta^2} \right)^{-1} \frac{dl(Y; \hat{\beta}_p, \hat{\theta}_p)}{d\theta}, \]

Iterative scheme usually starts from preliminary guess, obtained via summary statistics.
Illustration maximization

Exponential covariance with nugget effect. $\theta = (\theta_1, \theta_2, \theta_3)'$: log precision, logistic range, log nugget precision.
Asymptotic properties

\[ \hat{\theta} \approx N(\theta, G^{-1}). \]

Information matrix:

\[ G = G(\hat{\theta}) = -E \left( \frac{d^2 l}{d\theta^2} \right). \]
Prediction from joint Gaussian formulation

Prediction

\[ \hat{Y}_0 = E(Y_0|Y) = X_0\hat{\beta} + C_{0,.}C^{-1}(Y - X\hat{\beta}). \]

\( C_{0,.} \) is size 1 \( \times \) \( n \) vector of cross-covariances between prediction site \( s_0 \) and data sites.

Prediction variance

\[ \text{Var}(Y_0|Y) = C_0 - C_{0,.}C^{-1}C'_{0,.}. \]
Synthetic data

Consider unit square. Create grid of \(25^2 = 625\) locations. Use 49 data randomly assigned, or along center line (two designs).

Covariance \(C(h) = \tau^2 I(h = 0) + \sigma^2 (1 + \phi h) \exp(-\phi h),\ h = |s_i - s_j|\). \(\theta\) include transformations of: \(\sigma, \tau\) and \(\phi\).
Predictions

Random: Prediction

<table>
<thead>
<tr>
<th>Northing</th>
<th>Easting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Random: Prediction error

<table>
<thead>
<tr>
<th>Northing</th>
<th>Easting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Center: Prediction

<table>
<thead>
<tr>
<th>Northing</th>
<th>Easting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Center: Prediction error

<table>
<thead>
<tr>
<th>Northing</th>
<th>Easting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Likelihood optimization

True parameters $\beta = (-2, 3, 1), \theta = (0.25, 9, 0.0025)$.
Random design:
$\beta = [-2(0.486), 3.43(0.552), 0.812(0.538)]$
$\theta = [0.298(0.118), 7.89(1.98), 0.00563(0.00679)]$
Center design:
$\hat{\beta} = [-2.06(0.576), 3.4(0.733), 0.353(0.733)]$
$\hat{\theta} = [0.255(0.141), 7.19(1.97), 0.00283(0.00128)]$
Computational challenge for large $n$

1. Build and store $\Sigma(\theta) = \Sigma = C + \tau^2 I_n$
2. Compute $\log |\Sigma|$
3. Compute $\Sigma^{-1}$ or $(Y - X\beta)' \Sigma^{-1} (Y - X\beta)$

In general, the computational cost is $O(n^3)$. 
Possible solutions for large Gaussian models

- Approximate likelihood, Composite likelihoods.
- Basis representation.
- Markov representation.
- Predictive process models, sparse GPs.
- Tapered likelihood.
- Numerical linear algebra.
Composite likelihood

- Use pairs of joints, not full joint.

\[ l_{cl}(Y; \beta, \theta) = \sum_i \sum_{j>i} \log f(Y(s_i), Y(s_j); \beta, \theta) \]

- Fast calculations & Quantify loss in efficiency & Allows parallel computing.

\[ M \] blocks.

\[ l_{CL}(Y; \beta, \theta) = \sum_{k=1}^{M-1} \sum_{l>k} \log f(Y_k, Y_l; \beta, \theta) \]

\[ = \sum_{k=1}^{M-1} \sum_{l>k} \left\{ -\frac{1}{2} \log |\Sigma_{kl}| - \frac{1}{2} Z_k' Q_{kl} Z_l \right\} , \]

\[ Z_{kl} = (Y_k, Y_l)' - (X_k, X_l)' \beta \]

\[ \Sigma_{kl} = \Sigma_{kl}(\theta) \] block-pair covariance. Size \((n_k + n_l) \times (n_k + n_l)\)

\[ Q_{kl} = \Sigma_{kl}^{-1} \]

\[ n = \sum_{k=1}^{M} n_k \]
Asymptotic properties: Godambe sandwich

\[ \hat{\theta} \approx N(\theta, G^{-1}) \]

\[ G = G(\hat{\theta}) = H(\hat{\theta}) J^{-1}(\hat{\theta}) H(\hat{\theta}), \]

\[ H(\hat{\theta}) = -E \left( \frac{d^2 l_{CL}}{d\theta^2} \right), \quad J(\hat{\theta}) = \text{Var} \left( \frac{dl_{CL}}{d\theta} \right). \]
Markov property

In the time domain, the Markov property holds if for any \( t > s > u \),

\[
p(y(t)|y(s), y(u)) = p(y(t)|y(s)).
\]

The exponential correlation function gives a Markov process. (Proof by trivariate distribution, and conditioning.)
Precision matrix $Q$ : inverse covariance matrix

\[ \Sigma^{-1} = Q = \begin{bmatrix} Q_A & Q_{A,B} \\ Q_{B,A} & Q_B \end{bmatrix}. \]

$Q$ holds the conditional variance structure.
Interpretation of precision

\[ Q_A^{-1} = \text{Var}(Y_A|Y_B), \]

\[ \mathbb{E}(Y_A|Y_B) = \mu_A - Q_A^{-1}Q_{A,B}(Y_B - \mu_B), \]

(Proof by \( Q\Sigma = I \).
Or by writing out quadratic form and \( p(Y_A|Y_B) \propto p(Y_A, Y_B). \))
Algebraically equivalent forms

\[ E(Y_A|Y_B) = \mu_A + \Sigma_{A,B} \Sigma_B^{-1} (Y_B - \mu_B), \]
\[ \text{Var}(Y_A|Y_B) = \Sigma_A - \Sigma_{A,B} \Sigma_B^{-1} \Sigma_{B,A}. \]

\[ E(Y_A|Y_B) = \mu_A - Q_A^{-1} Q_{A,B} (Y_B - \mu_B), \]
\[ \text{Var}(Y_A|Y_B) = Q_A^{-1}. \]
Sparse precision matrix $Q$

For graphs the precision matrix is sparse.

$Q_{ij} = 0$ if nodes $i$ and $j$ are not neighbors. Conditionally independent.

$Q_{i,i+2} = 0$ for exponential covariance function on a regular grid in time.
All other variables than $y_i$ are denoted $y_{-i}$. Neighborhood of node $i$ is denoted $\mathcal{N}_i$.

$$p(y_i | y_{-i}) = p(y_i | y_j; j \in \mathcal{N}_i)$$

The neighborhood structure is given by the non-zero entries in $Q$. 

Conditional independence via $Q$
Sparse precision matrix \( Q \)

\[
p(Y_7 | Y_1, Y_2, Y_3, Y_4, Y_5, Y_6) = p(Y_7 | Y_5)
\]

\[
p(Y_i | Y_1, \ldots, Y_{i-1}) = p(Y_i | Y_{i-1})
\]

This sparseness means that several techniques from numerical analysis can be used. Solve \( Qb = a \) quickly for \( b \).
Cholesky factorization of $Q$

Common method for sampling and evaluation:

$$Q = \begin{bmatrix} Q_{1,1} & \cdots & Q_{1,n} \\ \vdots & \ddots & \vdots \\ Q_{n,1} & \cdots & Q_{n,n} \end{bmatrix} = L_Q L_Q' ,$$

Lower triangular matrix

$$L_Q = \begin{bmatrix} L_{Q,1,1} & 0 & \cdots & 0 \\ L_{Q,2,1} & L_{Q,2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{Q,n,1} & L_{Q,n,2} & \cdots & L_{Q,n,n} \end{bmatrix} ,$$

The Cholesky factor is often sparse, but not as sparse as $Q$, because it holds the partial (ordered) conditional structure, according to an ordering. This gives 'fill in'. The ordering matters in how the fill-in takes place.
Sparse $L_Q$

$L_Q$ is related to a recursion:

$$p(y) = p(y_n)p(y_{n-1}|y_n) \cdots p(y_1|y_2, \ldots, y_n)$$

Which can be removed in the conditioning? If $L_{Q,i,j} = 0$, it can be removed.

Sparsity is maintained for exponential covariance function in time dimension (Markov).
Sampling and evaluation using $L_Q$

$$Q = \begin{bmatrix} Q_{1,1} & \cdots & Q_{1,n} \\ \cdots & \cdots & \cdots \\ Q_{n,1} & \cdots & Q_{n,n} \end{bmatrix} = L_Q L'_Q,$$

$$L_Q Y = Z.$$

(Previously, for covariance we had $Y = LZ$.)

$$\log |Q| = 2 \log |L_Q| = 2 \sum_i L_{Q,ii}$$
GMRF for spatial applications.

A Markovian model can be constructed for a spatial Gaussian processes (Lindgren et al., 2011). The spatial process is viewed as a stochastic partial differential equation (SPDE), and the solution is embedded in a triangularized graph over a spatial domain. More later (23 Jan).