Gaussian Processes

STA5090Z: Advanced Topics in Regression

Birgit Erni

April 2025

Department of Statistical Sciences, University of Cape Town

What is a Gaussian Process?

stochastic process which generates functions like this

$$f \sim \mathcal{GP}(m, k)$$

mean function m

covariance function k

… sets of correlated data points

Example 1 (Rasmussen)

$$\{ \sim \mathcal{GP}(m, k)$$

$$m(x) = \frac{1}{4}x^2, \qquad k(x, x') = \exp(-\frac{1}{2}(x - x')^2)$$

x, x' two different x-values

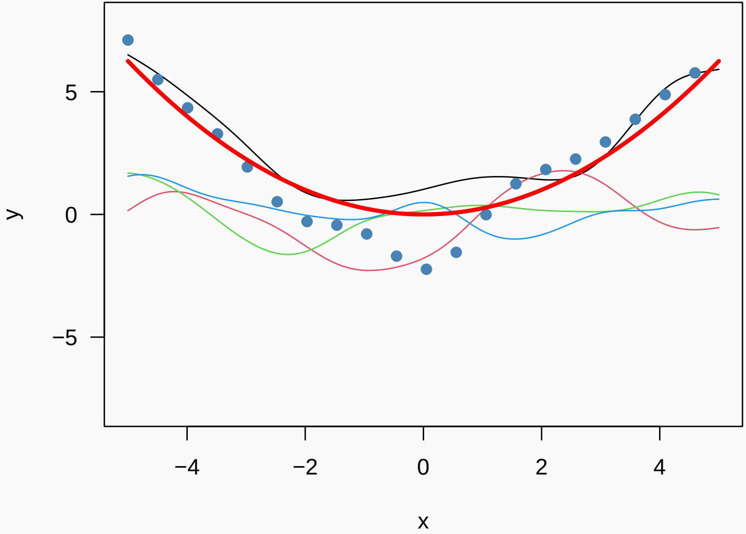
vector $\mathbf{f} \sim N(\mu, \Sigma)$

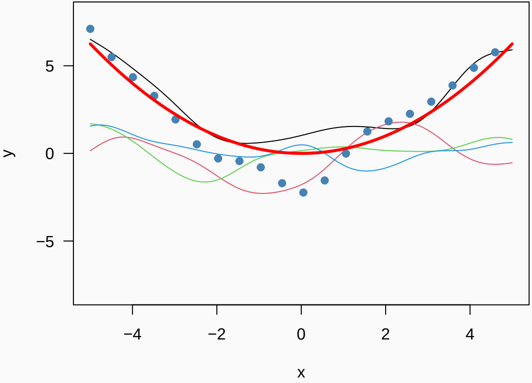
$\mu_i = m(x_i)$, $\Sigma_{ij} = k(x_i, x_j)$

Example 1

```r
library(mvtnorm)
set.seed(1)

x <- seq(-5, 5, length = 100)
d = abs(outer(x, x, "-")) # compute distance matrix, d_{ij} = |x_i - x_j|

mx <- x^2 / 4
Sigma_SE = exp(-d^2 / 2) # squared exponential kernel

y = mvtnorm::rmvnorm(1, mean = mx, sigma = Sigma_SE)

plot(x, y, type = "l", las = 1, ylim = c(-8, 8))

for (i in 1:3) {
  y = mvtnorm::rmvnorm(1, sigma = Sigma_SE)
  lines(x, y, col = i + 1)
}

xsel <- seq(1, 100, by = 5)

y = mvtnorm::rmvnorm(1, mean = mx, sigma = Sigma_SE)
points(x[xsel], y[xsel], col = "steelblue", pch = 19)
lines(x, mx, col = "red", lwd = 3)
```

Example 1





$$f(x) \sim N(m(x), k(x,x))$$

$$y(x) = f(x) + \epsilon, \qquad \epsilon \sim N(0, \sigma_n^2)$$

$$f \sim GP(m, k), \qquad y \sim GP(m, k + \sigma_n^2 \delta_i$$

k(x, x')

- \textcolor{blue{Covariance function} = Kernel
- describes covariance between x and x'

for example: squared exponential

$$k(x, x') = \exp(-\frac{1}{2}(x - x')^2)$$

more general form:

$$k(x, x') = \sigma_y^2 \exp\left(-\frac{(x - x')^2}{2\ell^2}\right) + \sigma_n^2 \delta_{ii'}$$

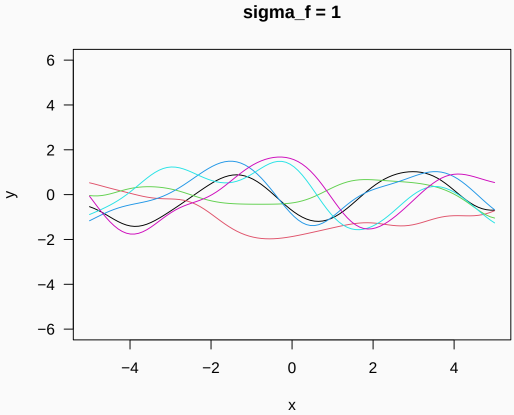
$\sigma_y = \sigma_f$ = marginal variability of f (d = 0)

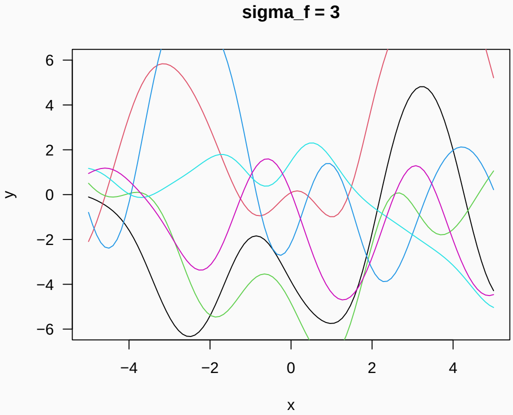
ℓ = length scale (how fast correlation decays) $\rho = \ell^2$

σ_n = observation process noise

$\delta_{ii'}$ = Kronecker delta, = 1 iff $i = i'$, else 0

σ_f





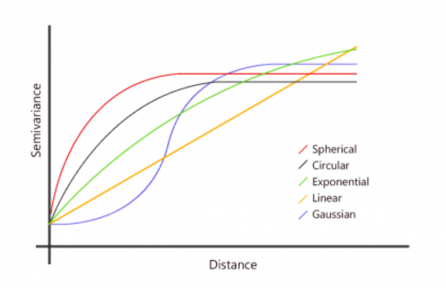
Regression

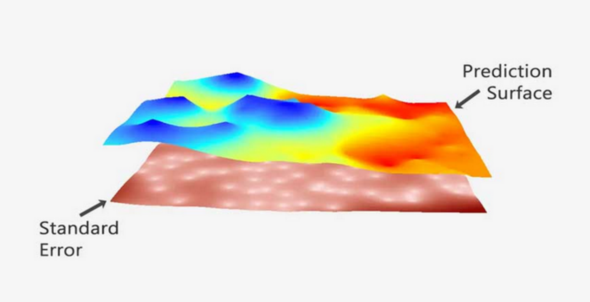
$$y_i = \beta_0 + \beta_1 x_i + e_i, \qquad e_i \sim N(0, \sigma^2)$$

$$\mathbf{y} \sim MVN(\mu, \sigma^2 I)$$

Kriging

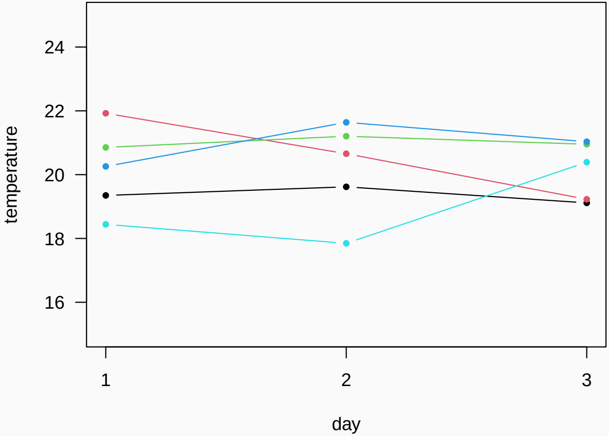
- two-dimensional Gaussian Process
- a random function where any k values on the function follow a MVN distribution

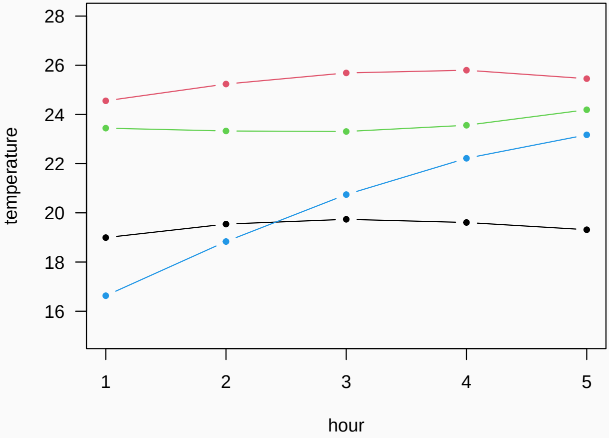




Time Series

- Example: temperature reading correlated in time (temperature on day 1, 2, 3)




temperature
28
26
24
22
20
18
16
1
2
3
4
5
hour

Where would use

- autocorrelated error term in time series regression, longitudinal model
- spatial random effect = kriging
- nonlinear regression
- computer simulations – predict outcome at intermediate points

Limitations

- very expensive (computationally): (O(Nˆ3))
- because of matrix inversion
- Cholesky decomposition of K matrix
- and some other tricks

Why useful?

- uncertainty estimates for predictions
- many dimensions?
- can construct new covariance functions as products of components

How are we going to use them?

- autocorrelated random effects / error terms $y = X\beta + f(t) + e$
- nonlinear regression $y = f(x) + e$, or $y = X\beta + f(z) + e$
- Stan (Bayesian)

https://mc-stan.org/docs/stan-users-guide/gaussian-processes.html

Stan Covariance Functions

https://mc-stan.org/docs/functions-reference/matrix_operations.html#gaussian-process-covariance-functions

squared exponential = exponentiated quadratic kernel

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{2l^2}\right)$$

```
matrix gp_exp_quad_cov(array[] real x, real sigma, real length_scale)
```

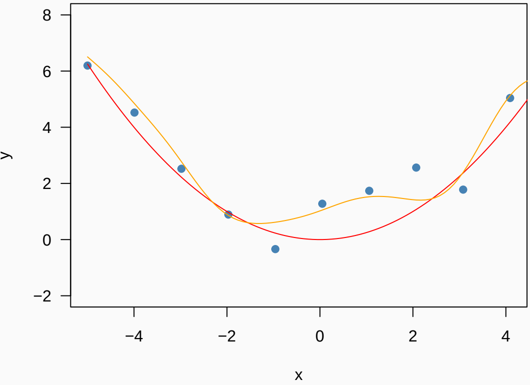
```
matrix[N, N] cov =   cov_exp_quad(x, alpha, rho)
                     + diag_matrix(rep_vector(1e-10, N));
```

add jitter to diagonal elements to stabilize numeral linear algebra

What are the parameters that we need to estimate?

- length scale l
- noise in observation process σ_n
- marginal function variance σ_f
- Bayesian, priors for parameters

Example 1: Rasmussen



$m(x) = \frac{1}{4}x^2$

$$k(x, x') = \exp(-\frac{1}{2}(x - x')^2)$$

.

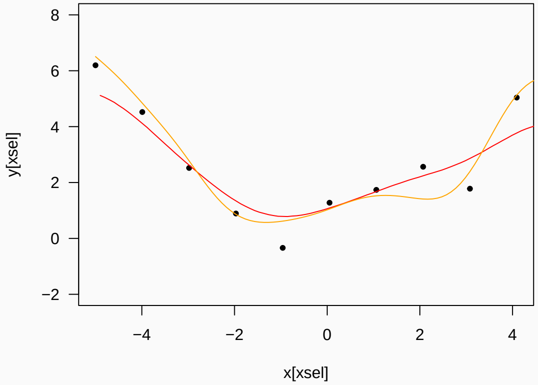
$$\sigma_n = 0.5$$

Fit model (Hamiltonian Monte Carlo)

```
library(cmdstanr)

x2 = x[-xsel]    # test set
data_list <- list(y_obs = y[xsel], x_obs = x[xsel],
                  N_obs = length(xsel),
                  x2 = x2, N2 = length(x2))

m1 <- cmdstan_model(stan_file = "./stan/gp1.stan")

m1.fit <- m1$sample(
  data = data_list,
  seed = 123,
  chains = 4,
  parallel_chains = 2,
```

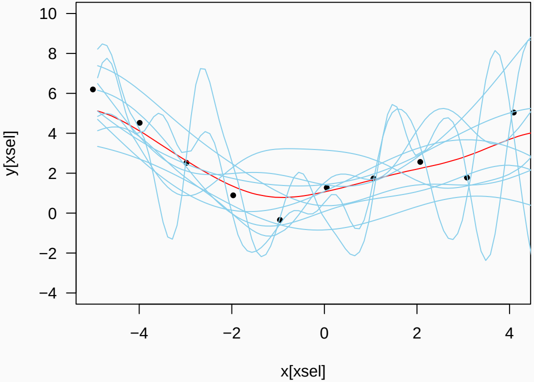
```r
m1.fit$summary()
```

```
# A tibble: 95 x 10
   variable    mean  median    sd   mad      q5    q95  rhat ess_bulk ess_tail
   <chr>      <dbl>   <dbl> <dbl> <dbl>   <dbl>  <dbl> <dbl>    <dbl>    <dbl>
 1 lp__     -16.8   -16.4   1.49  1.19  -19.7   -15.2   1.00    1042.     878.
 2 rho        0.820   0.760 0.364 0.297   0.340   1.48  1.00    1649.     963.
 3 alpha      2.25    2.25  0.583 0.559   1.29    3.20  1.00    1902.    1102.
 4 sigma      1.16    1.08  0.490 0.418   0.522   2.11  1.00    1582.    1084.
 5 f[1]       5.12    5.26  1.81  1.68    1.98    7.77  1.00    2987.    3204.
 6 f[2]       5.04    5.18  1.77  1.65    1.94    7.62  1.00    2904.    2980.
 7 f[3]       4.96    5.11  1.75  1.59    1.93    7.49  1.00    2884.    2957.
 8 f[4]       4.87    5.00  1.70  1.54    1.90    7.39  1.00    3021.    3155.
 9 f[5]       4.75    4.87  1.66  1.51    1.88    7.24  1.00    2824.    3251.
10 f[6]       4.64    4.75  1.63  1.48    1.87    7.08  1.00    2952.    3411.
# i 85 more rows
```



8
6
4
2
0
−2
y[xsel]
−4
−2
0
2
4
x[xsel]

Example 1: Posterior Sample



Theory: Predictive Distribution

$$\begin{bmatrix} f \\ f_* \end{bmatrix} \sim N\left(\begin{bmatrix} \mu \\ \mu_* \end{bmatrix}, \begin{bmatrix} \Sigma & \Sigma_* \\ \Sigma_*^T & \Sigma_{**} \end{bmatrix}\right)$$

Σ_*: $n \times n_*$ matrix

$$f_*|X_*, X, f \sim N\left(\mu_* + \Sigma_*^T \Sigma^{-1}(f - \mu), \Sigma_{**} - \Sigma_*^T \Sigma^{-1} \Sigma_*\right)$$

$$\begin{pmatrix} X_1 \\ X2 \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}\right)$$ then $X_1|X_2 \sim N(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$.

Marginal Likelihood

$$\ell = \log p(y|x,\theta) = -\frac{1}{2}\log|\Sigma| - \frac{1}{2}(y-\mu)^T\Sigma^{-1}(y-\mu) - \frac{n}{2}\log(2\pi)$$

(log) marginal likelihood

$$p(y|X) = \int p(y|X,w)p(w)dw$$

- maximise to estimate parameters w

References

- Rasmussen & Williams. Gaussian Processes for Machine Learning. MIT. https://direct.mit.edu/books/book/2320/Gaussian-Processes-for-Machine-Learning
- Rasmussen 2006. Gaussian Processes in Machine Learning.
- Roberts et al. Gaussian processes for time-series modelling. Philosophical Transactions of the Royal Society
- https://betanalpha.github.io/assets/case_studies/gaussian_processes.html
- Ebden 2008. Gaussian Processes for Regression: A Quick Introduction.
- Neal, R.M. (1997) 'Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification'. arXiv. https://doi.org/10.48550/arXiv.physics/9701026.
- Neal, R.M. (1998) 'Regression and Classification Using Gaussian Process Priors', in

Example 1: Simulated GP

https://michael-franke.github.io/Bayesian-Regression/practice-sheets/10c-Gaussian-processes.html

Conditional Predict

$$f_*|X_*, X, f \sim N(K(X_*, X)K(X, X)^{-1}f, K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X^*))$$

Marginal Likelihood

https://mc-stan.org/cmdstanr/reference/model-method-optimize.html

HMC

Example 2: mcycle