Sampling Paths from a GP

XC

03/07/2021

```
#tinytex::install_tinytex()
```

General set-up

- a zero-centered GP(m(.), K(.,.))
- f is a function drawn from this GP
- a vector (x1, … xn), the function values (f(x1), … f(xn)) or (Y)
- must have a multivariate GP with
  - mean: (m(x1),…, m(xn))
  - covariance matrix Sigma with Sigma_{ij} = K(xi, xj)
- so we could make use of this property (with 2 moments determin the whole distribution), draw this function from GP:
  1. select a fine grid of x-coords
  2. use mvrnorm() from MASS to draw function values at these points
  3. then connect them with straight lines

Generate Covariance matrix

Generate Covariance matrix from a known kernel function at points x

```
cov_matrix <- function(x, kernel_fn, ...) {
  outer(x, x, function(a, b) kernel_fn(a, b, ...))
}
```

Sample

Given x coords, take N draws from the GP with K evaluated using kernel_fn at x

```
draw_samples <- function(x, N, kernel_fn, ...) {
  set.seed(03-07-2021)

  Y <- matrix(NA, nrow = length(x), ncol = N)
  for(n in 1:N) {
    K <- cov_matrix(x, kernel_fn, ...)
    Y[, n]<- mvrnorm(1, mu = rep(0, length(x)), Sigma = K)
  }
  Y
}
```

Parameters

Use the following parameters for the rest code

```
x <- seq(0, 2, length.out = 201)          # x-coords
N <- 3                                    # no. of draws
col_list <- list("red", "blue", "black")  # col for lines of different draws
```

Squared exponential(SE) kernel

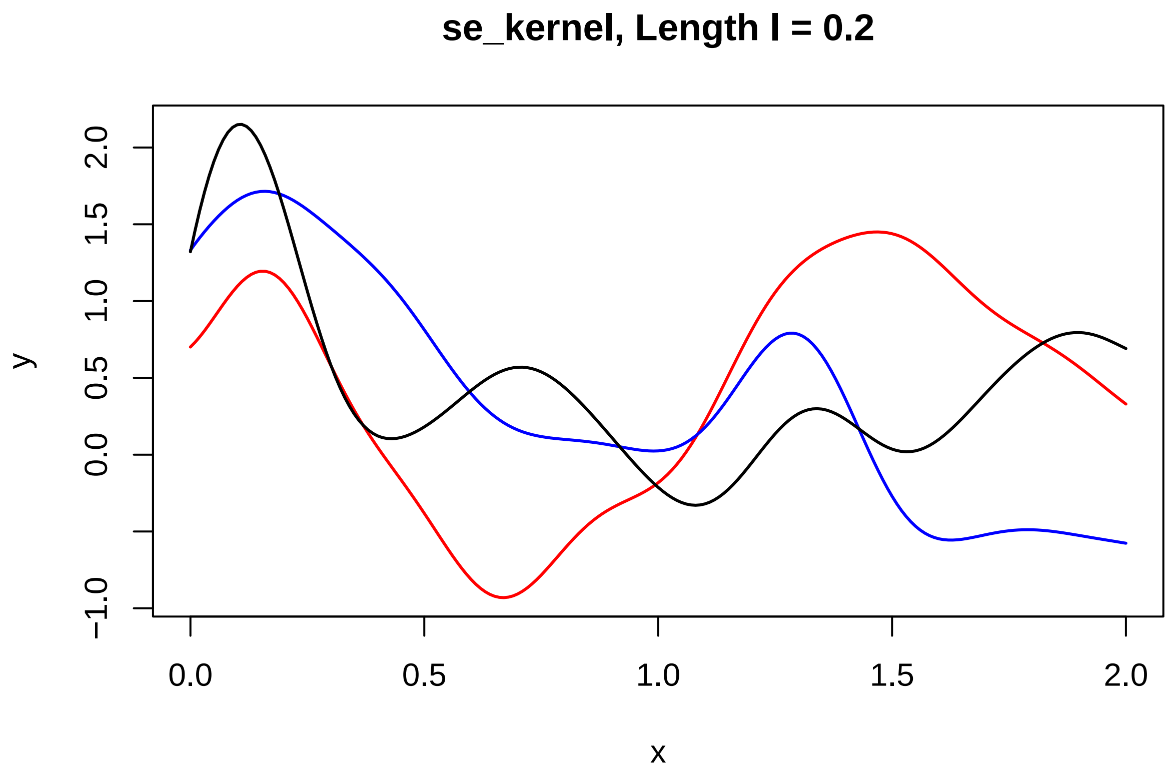
SE also known as *radial basis furntion (RBF) kernel* or the *Gaussian kernel* has the form: $K(x, x') = \sigma^2 exp(-\frac{||x-x'||^2}{2l^2})$,

where $\sigma^2 > 0$ and $l > 0$ are hyperparameters. $\sigma^2 > 0$ tells how variable the function is overall, and set to 1 for simplicity.

It's the most commonly used kernel as its computational tractability.

Now generates 3 draws from the SE kernel with $l = 0.2$

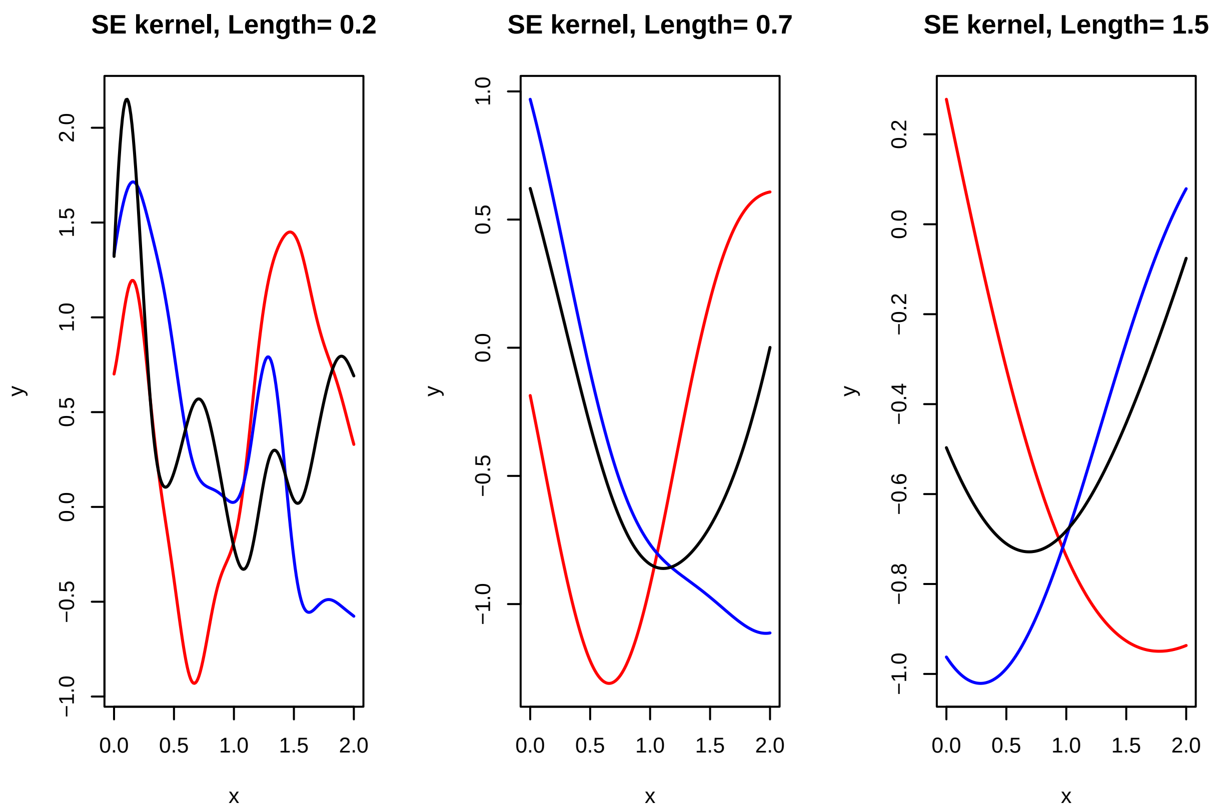
```
se_kernel <- function(x, y, Sigma = 1, Length = 1) {
  Sigma^2 * exp(- (x - y)^2 / (2 * Length^2))
}


Y <- draw_samples(x, N, kernel_fn = se_kernel, Length = 0.2)

plot(range(x), range(Y), xlab = "x", ylab = "y", type = 'n',
     main = "se_kernel, Length l = 0.2")


for(n in 1:N) {
  lines(x, Y[, n], col = col_list[[n]], lwd = 1.5)
}
```



Show how changing "length-scale" parmaeter l affects the function drawn.

The smaller the l, the more wiggly the function drawn

```
par(mfrow = c(1, 3))

for(l in c(0.2, 0.7, 1.5)) {
  Y <- draw_samples(x, N, kernel_fn = se_kernel, Length = l)   # draw samples

  plot(range(x), range(Y), xlab = "x", ylab = "y", type = "n",
       main = paste("SE kernel, Length=", l))                  # plot framework

  for(n in 1:N) {
    lines(x, Y[, n], col = col_list[[n]], lwd = 1.5)           # lines up samples
  }

}
```




```
for(l in c(0.2, 0.7, 1.5)) {
  Y <- draw_samples(x, N, kernel_fn = se_kernel, Length = l)  # draw samples
  str(Y)
}
```

```
##  num [1:201, 1:3] 0.701 0.732 0.768 0.806 0.848 ...
##  num [1:201, 1:3] -0.186 -0.21 -0.233 -0.256 -0.28 ...
##  num [1:201, 1:3] 0.278 0.265 0.252 0.239 0.227 ...
```

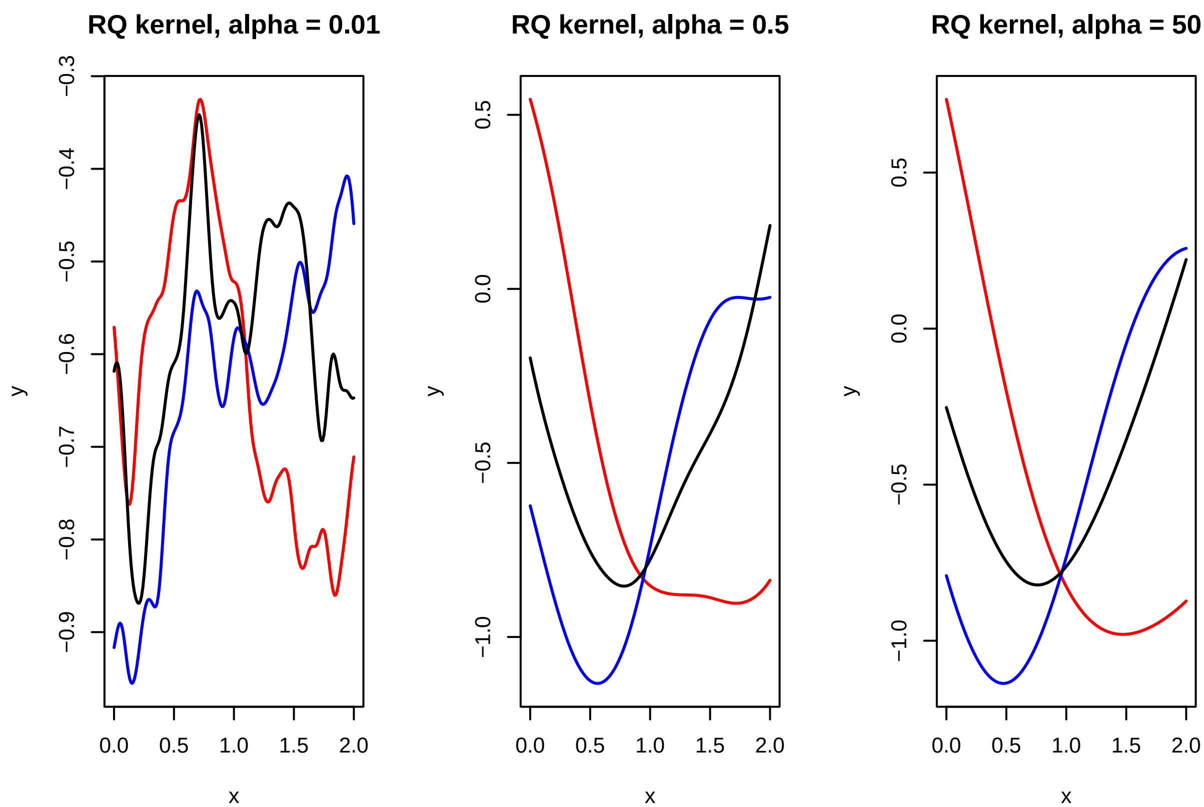
Rational quadratic (RQ) kernel

The rational quadratic (RQ) kernel has the form $K(x, x') = \sigma^2(1 + \frac{||x-x'||^2}{2\alpha l^2})^{-\alpha}$, where $\sigma > 0$ and $\alpha > 0$ are hyperparameters.

Below we create the RQ kernel function and see how length l affects the function drawn:

```
rq_kernel <- function(x, y, Sigma = 1, alpha = 1, Length = 1) {
  Sigma^2 * (1 + (x - y)^2 / (2 * alpha * Length^2))^(-alpha)
}

par(mfrow = c(1, 3))

for (a in c(0.01, 0.5, 50)) {
  Y <- draw_samples(x, N, kernel_fn = rq_kernel, alpha = a)
```

```
  plot(range(x), range(Y), xlab = "x", ylab = "y", type = "n",
       main = paste("RQ kernel, alpha =", a))

  for(n in 1:N) {
    lines(x, Y[, n], col = col_list[[n]], lwd = 1.5)
  }

}
```



Matérn covariance function

The Matérn covariance function has the form

$K(x,x') = \sigma^2 \frac{1}{2^{v-1}\Gamma(\nu)}(\frac{\sqrt(2\nu)||x-x'||}{l})^{\nu} K_{\nu}(\frac{\sqrt(2\nu)||x-x'||}{l})$,

where

- $\frac{\sqrt(2\nu)}{l} > 0$: spatial scale paramter, while its inverse is sometimes referred to as a correlation length
- $\nu > 0$: smooth parameter defines the Hausdorff dimension and the differentiability of the sample paths
  - if $\nu = n + 1/2$, the Matérn function reduces to the product of an exponential function and a polynomial $M(\mathbf{h}|n+1/2,a) = exp(-a||\mathbf{h}||)\Sigma_{k=0}^{n}\frac{(n+k)!}{(2n)!}\binom{n}{k}(2a||\mathbf{h}||)^{n-k}$, n = 0, 1, ...
  - the larger the ν, the smoother the process
  - in practice, $\nu = 1/2$, $\nu = 3/2$ and $\nu = 5/2$ are used more often