Abstract

1   Introduction

2   Fixed Rank Kriging

2.1   SRE model

2.2   Estimation

2.3   Prediction

2.4   Applicaiton of sparse matrix inversion

3   Distributed computation with Hadoop

3.1   Hadoop and Rhipe for parallelising data and computations

3.2   FRK Operations to parallelise

4   Package layout

4.1   Classes and methods

4.2   Overall program structure

4.3   Simple examples

```
> library(sp)
> library(ggplot2)
> library(dplyr)
> library(FRK)
> #      devtools::load_all("~/Wollongong/pkgs/FRK",
> #                               export_all = FALSE)
> opts_FRK$set("progress",FALSE)
> opts_FRK$set("parallel",0L)

> ## Load data
> set.seed(1)
> sim_process <- data.frame(x = seq(0.005,0.995,by=0.01)) %>%
+              mutate(y=0,proc = sin(x*10) + 0.3*rnorm(length(x)))
> sim_data <- sample_n(sim_process,50) %>%
+              mutate(z = proc + 0.1*rnorm(length(x)), std = 0.1)
> coordinates(sim_data) = ~x + y# change into an sp object
```

```
> ## Prediction (BAU) grid
> grid_BAUs <- auto_BAUs(manifold=real_line(),data=sim_data,cellsize = c(0.01),type="grid")
> grid_BAUs$fs = 1

> ## Set up SRE model
> G <- auto_basis(m = real_line(),data=sim_data,
+                 nres = 2,
+                 regular=6,
+                 type = "bisquare",
+                 subsamp = 20000)

[1] "Number of basis at resolution 1 = 6"
[1] "Number of basis at resolution 2 = 12"

> f <- z ~ 1
> S <- SRE(f,list(sim_data),G,
+          grid_BAUs,
+          est_error = FALSE)

[1] "Binned data in 0.0619999999999998 seconds"

> S <- SRE.fit(S,n_EM = 50,tol = 1e-5,print_lik=TRUE)

[1] "Maximum EM iterations reached"
[1] "Warning: Ignoring constants in log-likelihood computation"

> grid_BAUs <- SRE.predict(S,pred_locs = grid_BAUs,use_centroid = TRUE)

> X <- grid_BAUs@data %>%
+      filter(x >= 0 & x <= 1)
> g1 <- LinePlotTheme() +
+            geom_line(data=X,aes(x,y=mu)) +
+            geom_errorbar(data=X,aes(x=x,ymax = mu + 2*sqrt(var), ymin= mu - 2*sqrt(var)))
+            geom_point(data = data.frame(sim_data),aes(x=x,y=z),size=3) +
+            geom_line(data=sim_process,aes(x=x,y=proc),col="red")
> print(g1)
```

Need an end comment
1-d and 2-d