Hierarchical Bayesian spatial modelling with Multivariate, Nonstationary and Nonseparable processes for air pollution in Euskadi

Sébastien Coube-Sisqueille

Basque Center for Applied Mathematics
Applied Statistics team

June 23, 2023

The spatial model

My way to deal with large geostatistical data

My way to deal with dense goestatistical data

Air contamination in Euskadi

What is at stake?

- Public health problem.
  - Surges in emergency admissions in hospitals after contamination peaks.
  - Long-term exposure even when there is no peak.
- Subsequent social justice problem due to environmental inequalities.
- Public policies assessment (low-emission zones, low-speed zones, wood-fired power plants).

Who are our culprits?

- Nitrogen Oxides (NO, NO_2, NO_X).
- Ozone.
- Fine particles of size 10 and $2.5\mu m$.

What is the format of our data?

- Fixed measurement stations.
- One observation per day.
- Some NAs.
- Some misalignment.

Modelling choices 1/2: the big decisions

We pursue a Bayesian **space-time** hierarchical model aiming to **interpolate** the data.

- No modelling of the diffusion and transformation processes.
- We can nonetheless assess the effect of some predictors using linear effects.

We want a **multivariate** model.

- "Borrow strength" between variables to have better predictions.
- Explore the correlations between those variables.

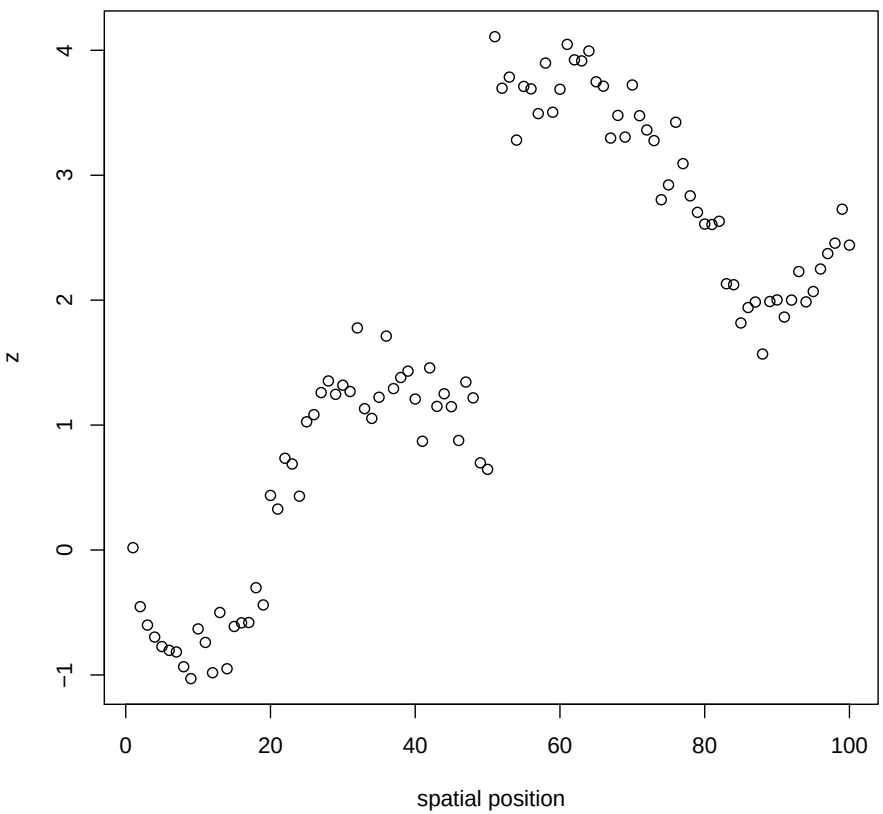
We want to separate long-term and short-term components.

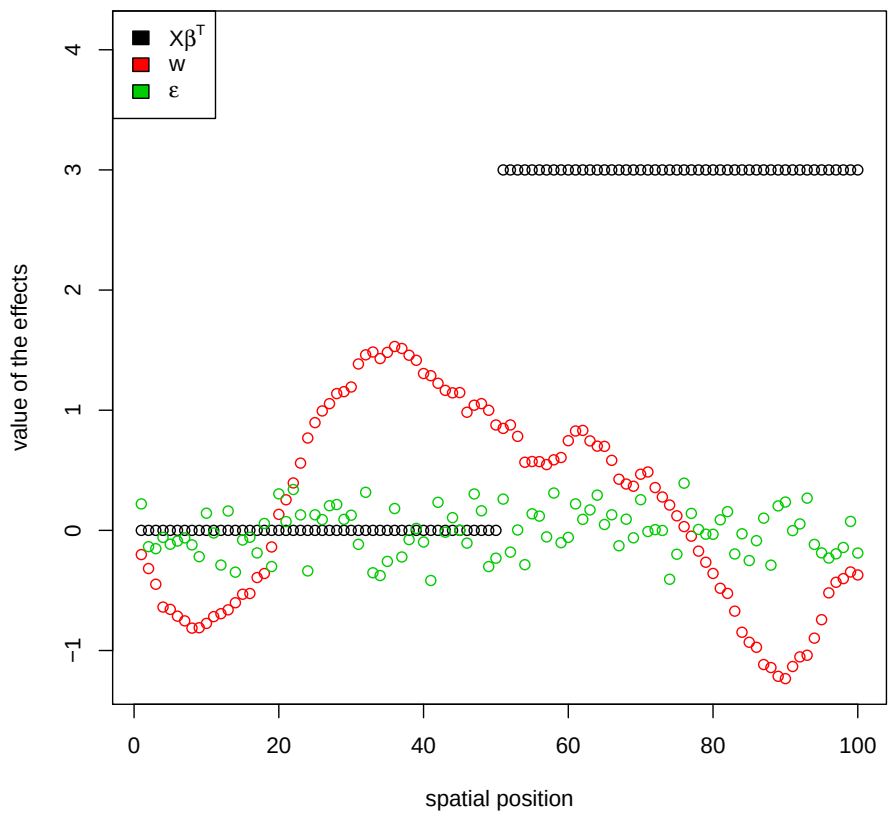
- Long-term exposure.
- Peaks.

Gaussian data model

The data is modelled using:

- Linear effects (like in a good old linear model).
- Noise (like in a good old linear model).
- A **spatially coherent** component not explained by the linear effects.





Example

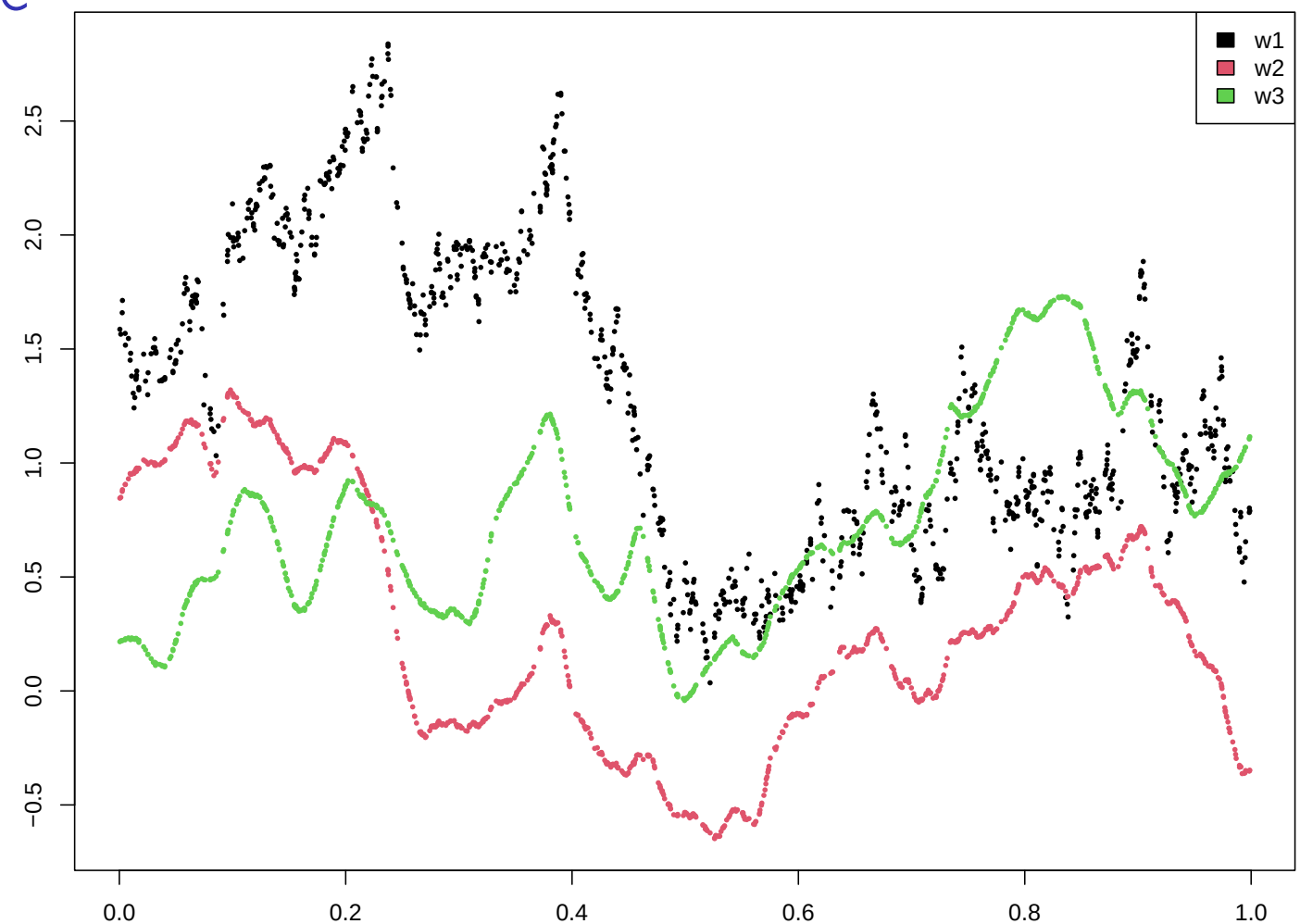
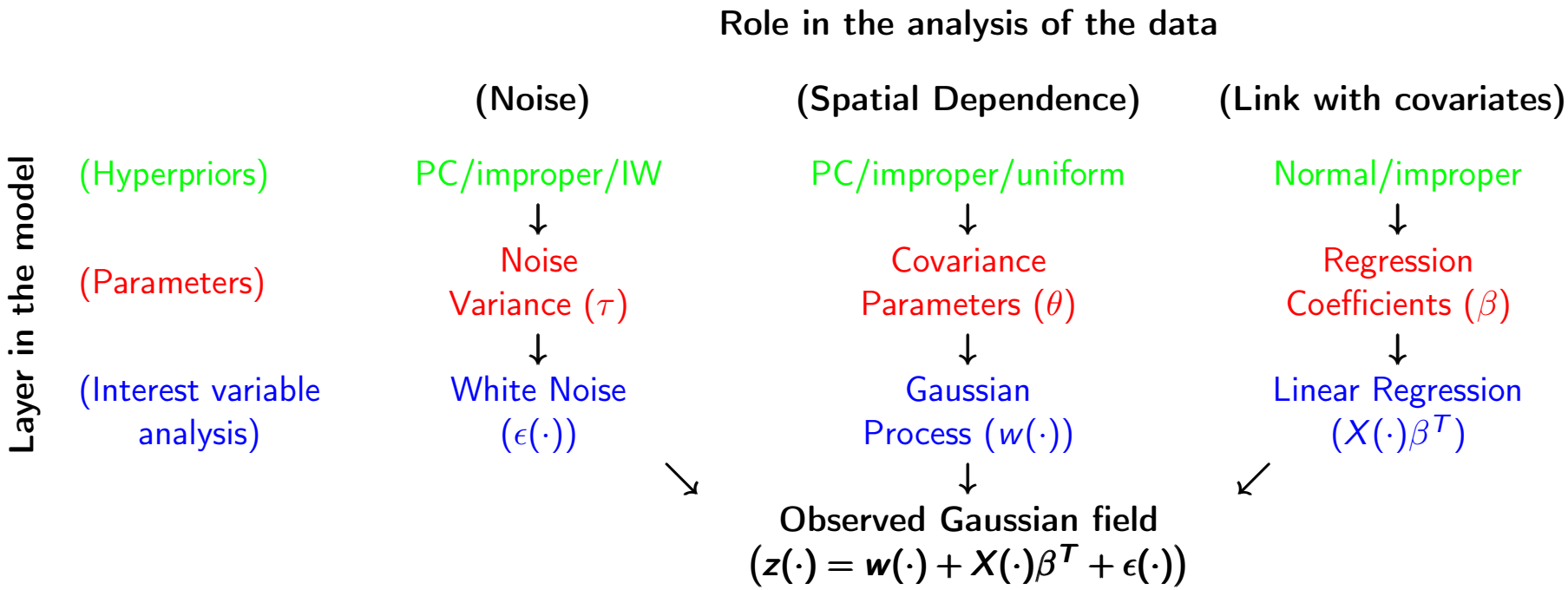


Figure 2: A multivariate Gaussian field

Hierarchical Model Architecture

The decomposition of the data is embedded in a **hierarchical model**:

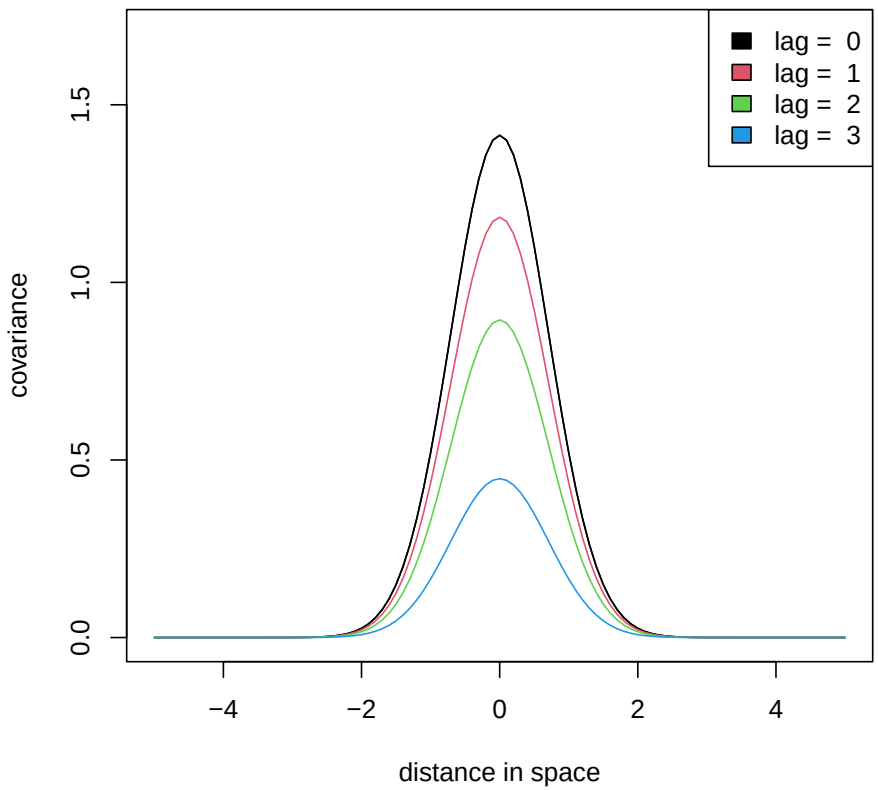
$$p(parameters, process|data) \propto$$
$$p(data|parameters, process)p(process|parameters)p(parameters).$$

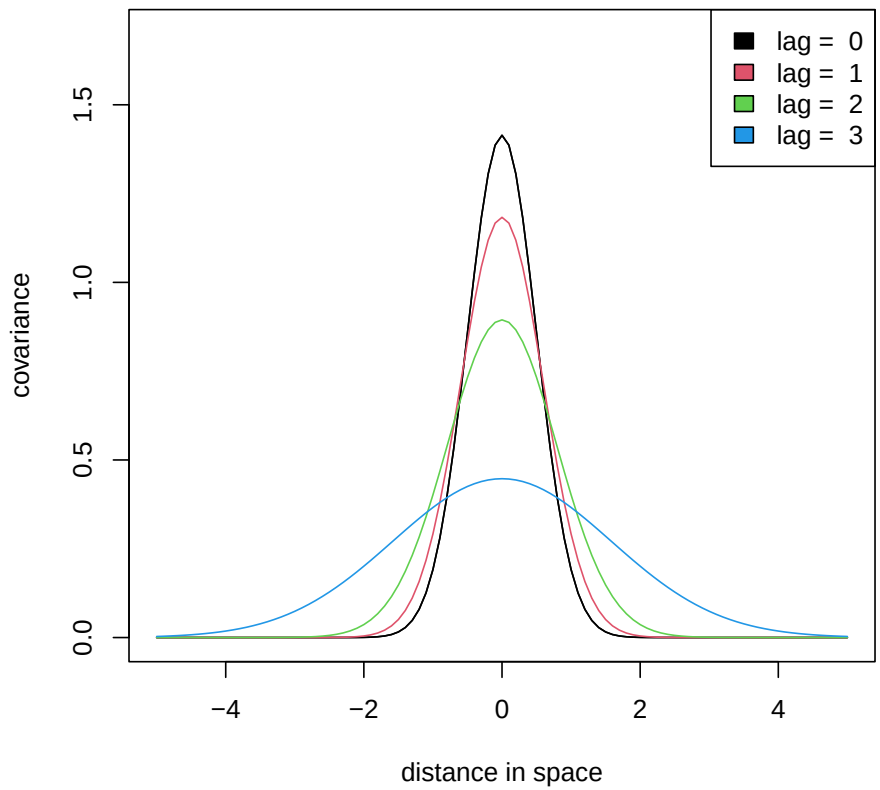


Nonseparability

I want to add **Nonseparability**, using the work of Allard et al. (2022).

- Better performance in prediction Allard et al. (2022).
- Better interpretation of the diffusion of a pollutant.





The spatial model

My way to deal with large geostatistical data

My way to deal with dense goestatistical data

Un problème de taille [1]

The **marginal dimensions** of the data set are **misleading**. In a data set with...

- $n_{var} = 4$ variables (that's Iris).
- $n_{space} = 10000$ spatial sites (with Inla you need 10 minutes).
- $n_{time} = 3500$ measurements in time (10 years of daily measurements - just perfect to teach time series to your students).

... we get **140 M** observations. And that is not all!
We need to set up a **denser** model, with:

- Space links because we have a spatial component.
- Time links because we have a temporal component.
- Space-time links because of nonseparability.
- Links between several variables to evaluate multivariate correlations.

[1] In French: "a matter of size" / "a big problem"

Objectives for the implementation

I would like to have an **efficient** and **scalable** implementation.

- Something that can be run from a laptop for a "moderate" ($10^5 - 10^6$ observations) data set.
- Something that can exploit the capacities of a cluster.

In practice, what does it take?

- Control CPU use.
- Control RAM use.
- Allow for **parallelization**.
  - Introduce parallelizability...
  - and avoid **bottlenecks**!
- Use GPUs?

Nearest Neighbor Gaussian Process approximation

Problem: allow the covariance parameters to dictate the behavior of the latent field. The naive case (kriging) is

- $O(n^3)$ FLOPs.
- $O(n^2)$ RAM.

We need to use an approximation that is

- cheap in RAM.
- cheap in FLOPs.
- parallelizable.

A Nearest Neighbor Gaussian Process approximation looks like an **Auto-Regressive model** (for time series) but with a **spatial component**.

- Linear FLOP and RAM cost in the number of observations.
- Embarrassingly parallelizable.
- Induces **sparsity**.

But is that enough when you have millions of observations?

Recursive Nearest Neighbor Gaussian Processes

Idea: compute the approximation **for only 1 time period** and recycle it!
Restrictions:

- The measurements must be **regularly spaced in time**.
- The measurement sites must be **fixed in space**.
- The process must be **stationary in correlation** (nonstationary marginal variance is still allowed).

Strengths:

- No need for full observations (NAs are perfectly OK).
- No need to have all the variables observed at a measure site (I can have only NOX and Ozone in one site, PMs in another site, etc).
- **The cost does not depend on the number of time periods!**

Parallel programming in R

How to do parallel computing in R?

- With parallel or its loop-friendly version foreach.
- With omp through Rcpp.
- With future.

Problems:

- in Windows, parallel needs to **copy** the objects it uses.
- parallel does not allow to modify objects from each thread, you need to **collect** and **re-process** the results.
- omp **crashes when you put some R** in the loop.
- with parallel, there is an **overhead** depending empirically on:
  - a **fixed cost**.
  - the **RAM size** of the **outputs**.
  - the **RAM size** of the **inputs**.

My approach:

- use a **stepwise approach**.
- things that should work well do not always work well.
- the reverse is much less frequent.

Cutting corners (of the matrix)

The upper corner of the covariance patrix between a point and its parents is **block-Toeplitz** with **symmetric blocks**.

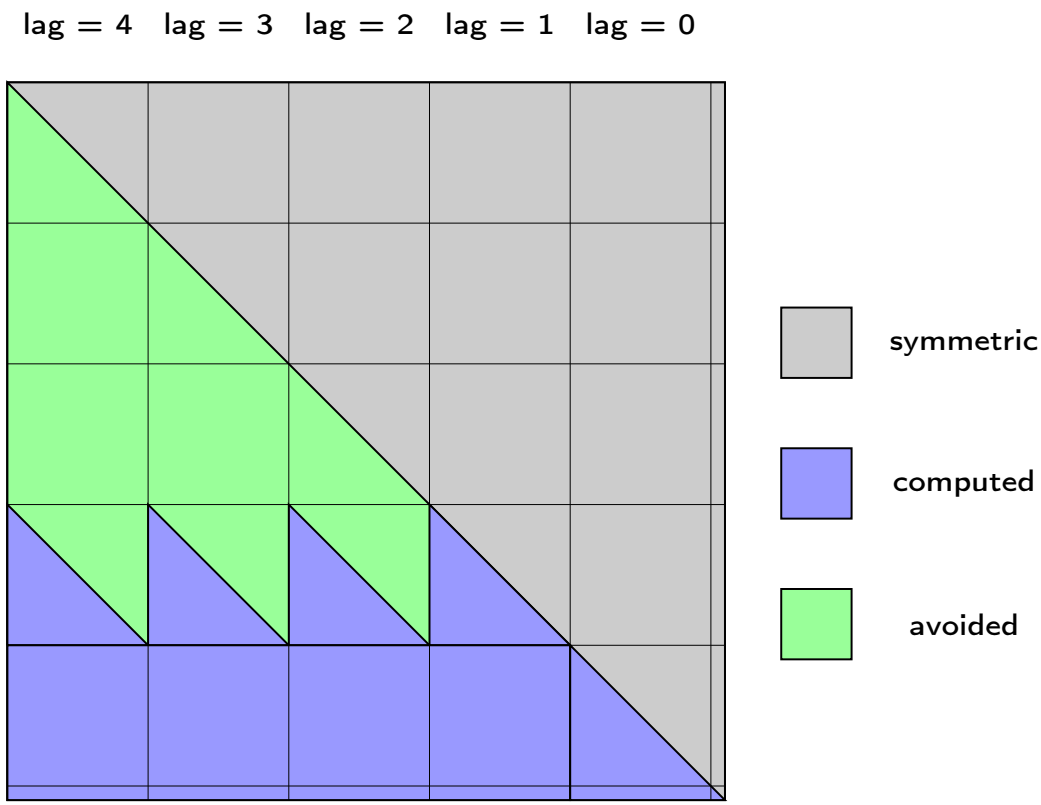


Figure 4: Avoided computations in the covariance matrix

The spatial model

My way to deal with large geostatistical data

My way to deal with dense goestatistical data

Updating the latent field

The latent field has two problems:

- space-time autocorrelation.
- coupling with the high-level parameters.

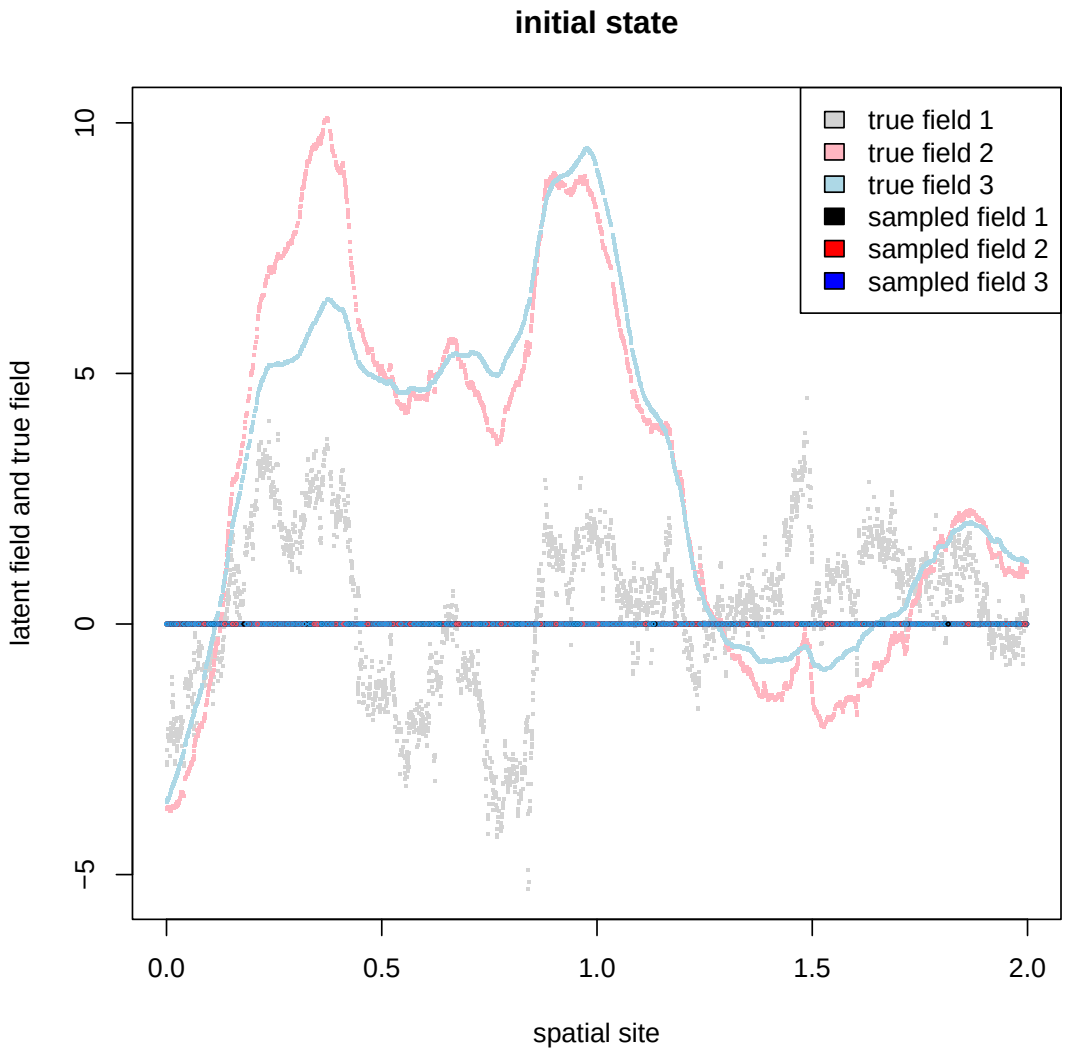
Those two problems need **sampling from the latent field** to be solved.

- The update groups must be **small** to limit the cost of an update.
- The clusters must be **large** to deal with spatial auto-correlation.

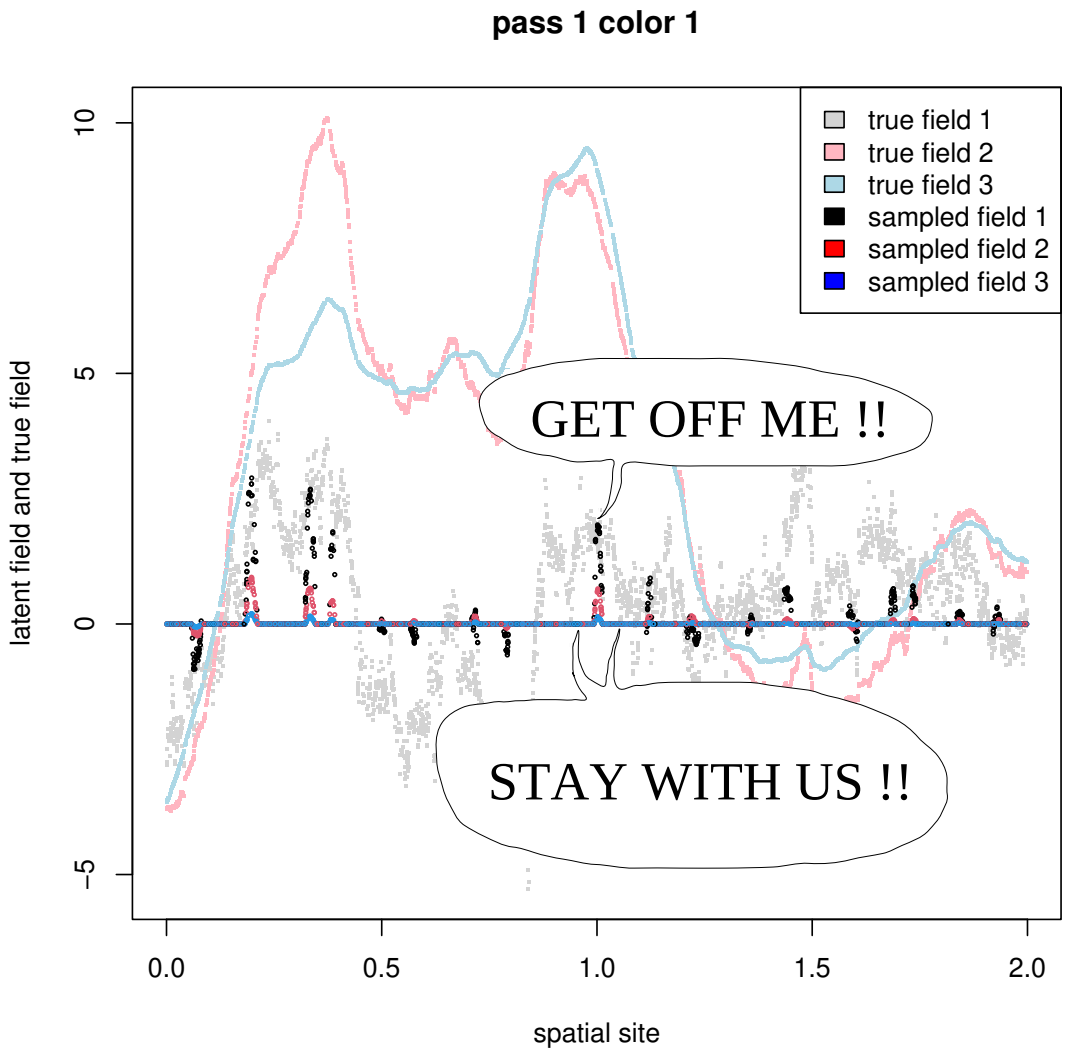
Original solution: using a **blocks-and-bases** sampling with **spatially coherent perturbations**.

- Parallelizable thanks to chromatic schemes like in Coube-Sisqueille and Liquet (2021).
- Deals with auto-correlation thanks to various resolutions.
- Cheap thanks to sparsity of the basis functions (tapering).

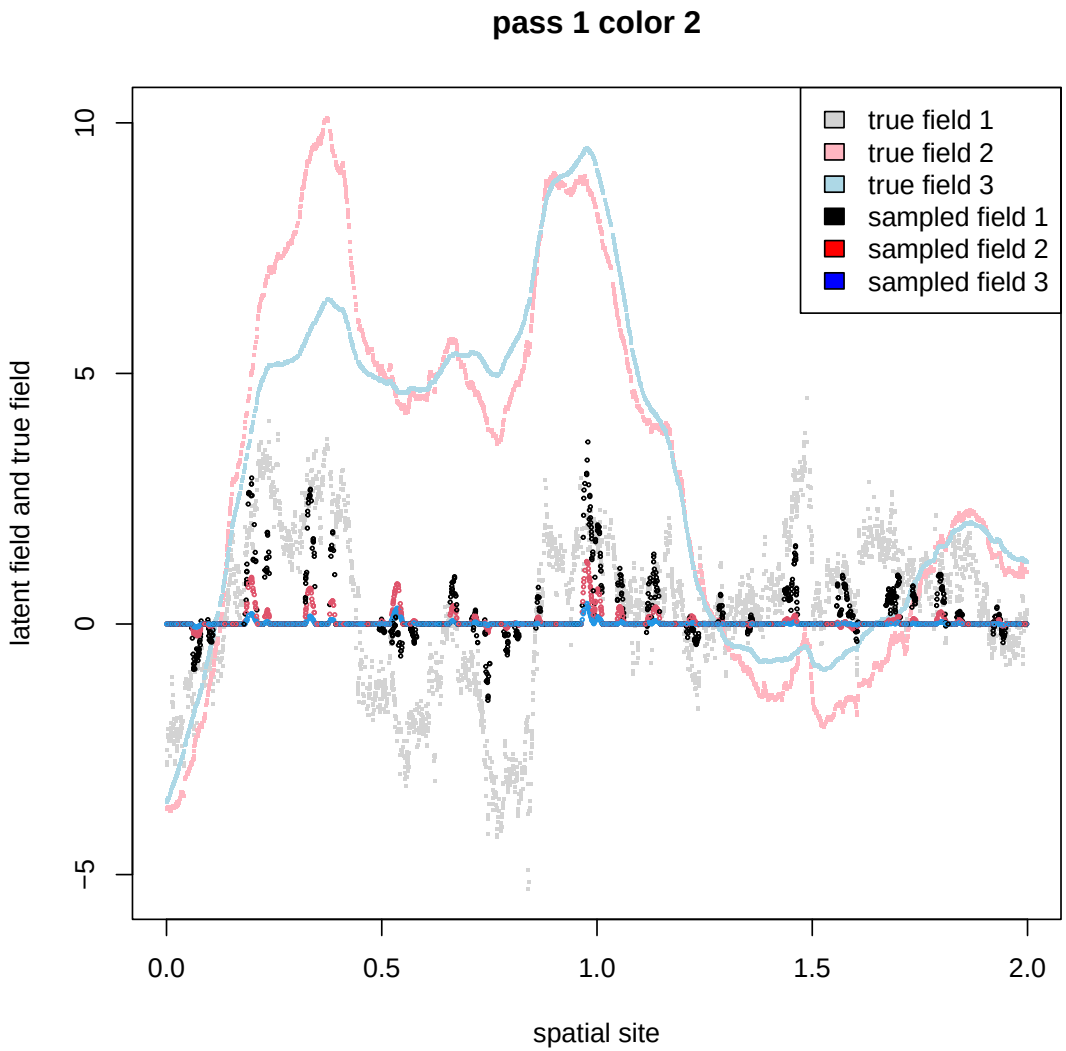
Only blocks (usual algorithm)



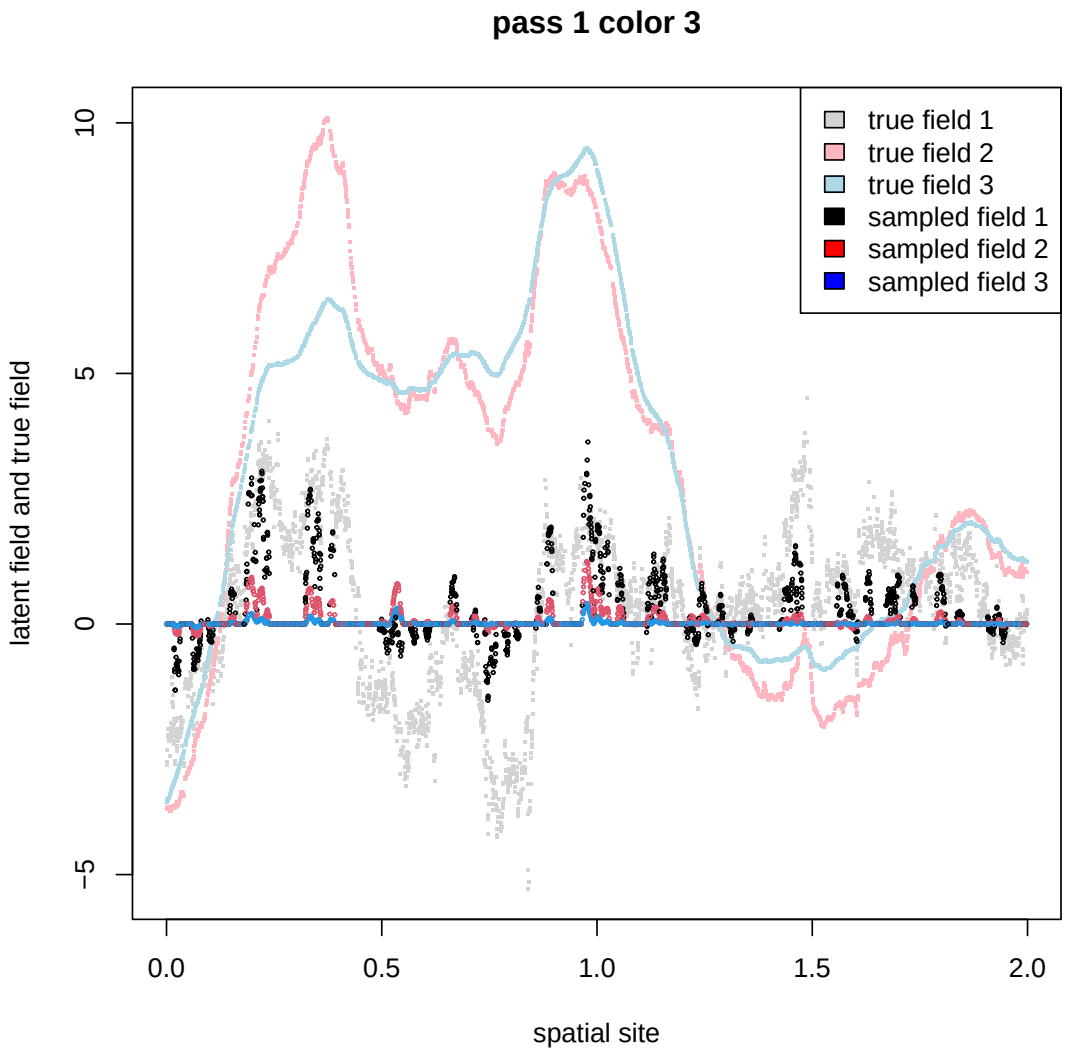
Only blocks (usual algorithm)



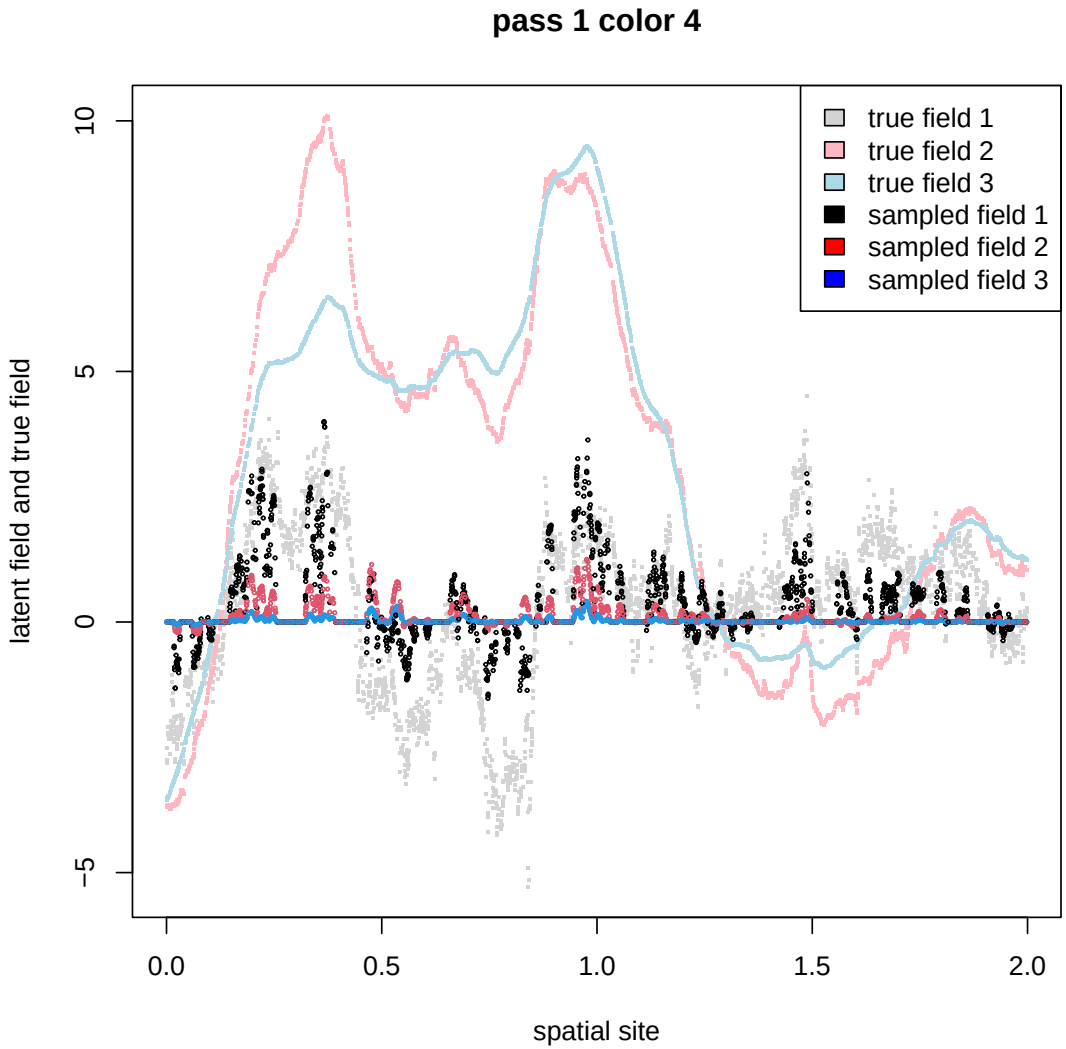
Only blocks (usual algorithm)



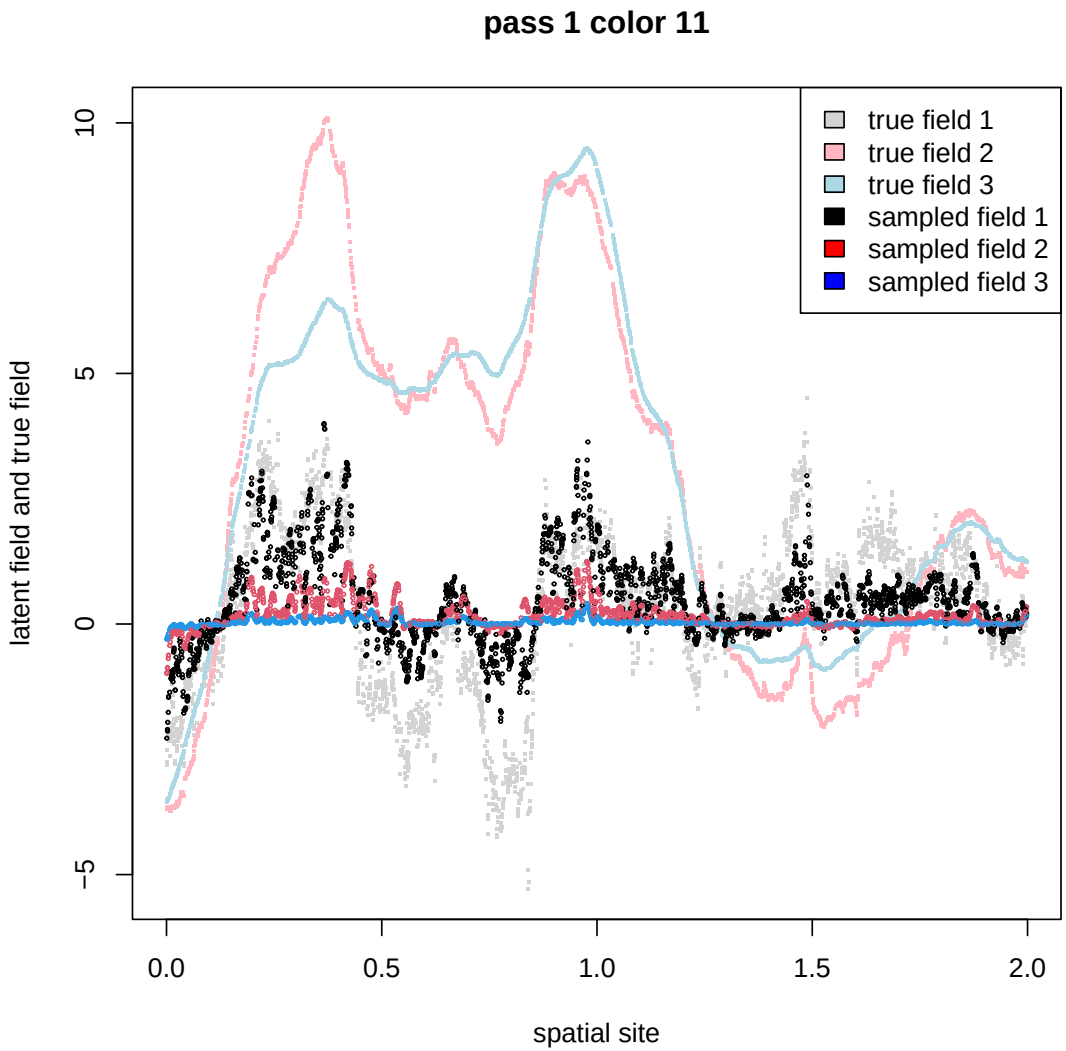
Only blocks (usual algorithm)



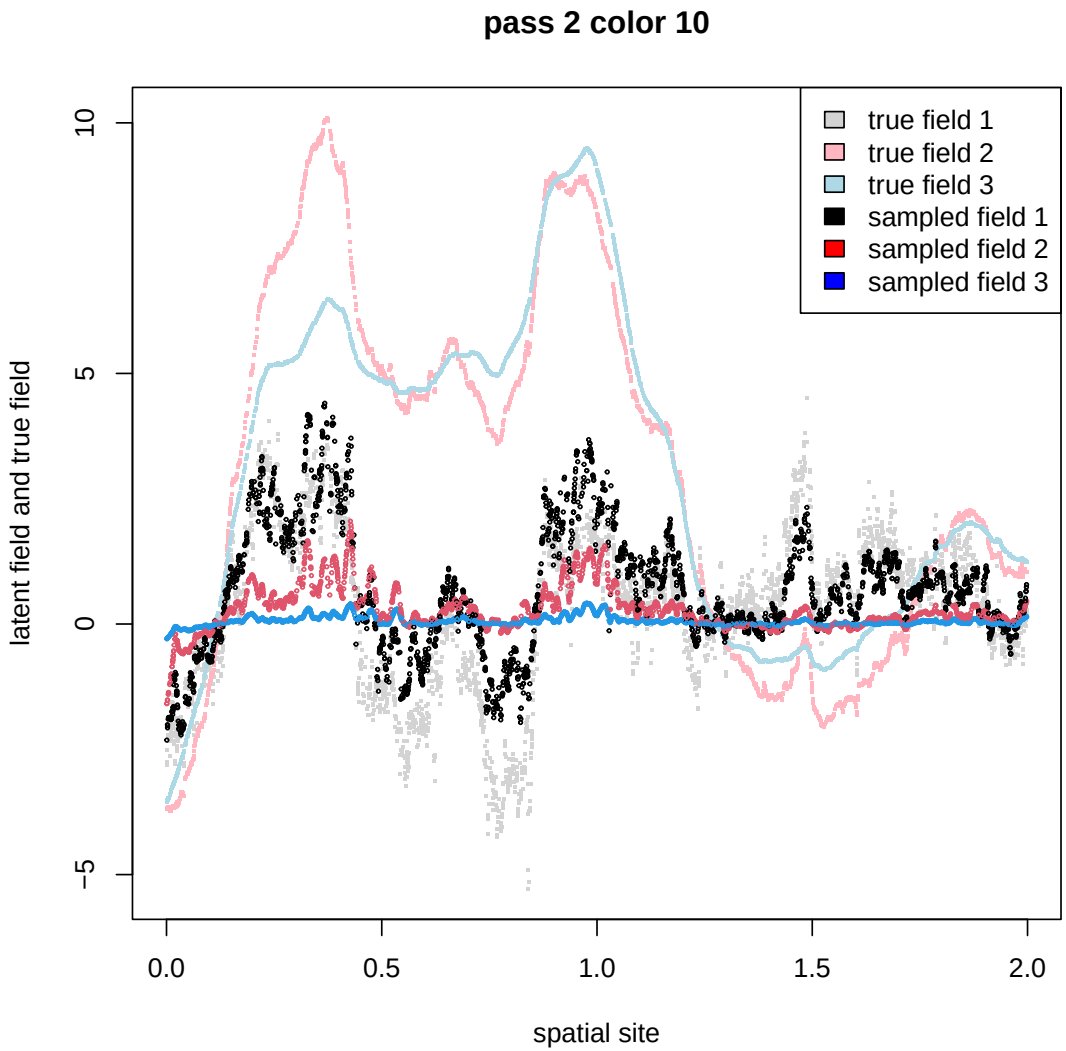
Only blocks (usual algorithm)



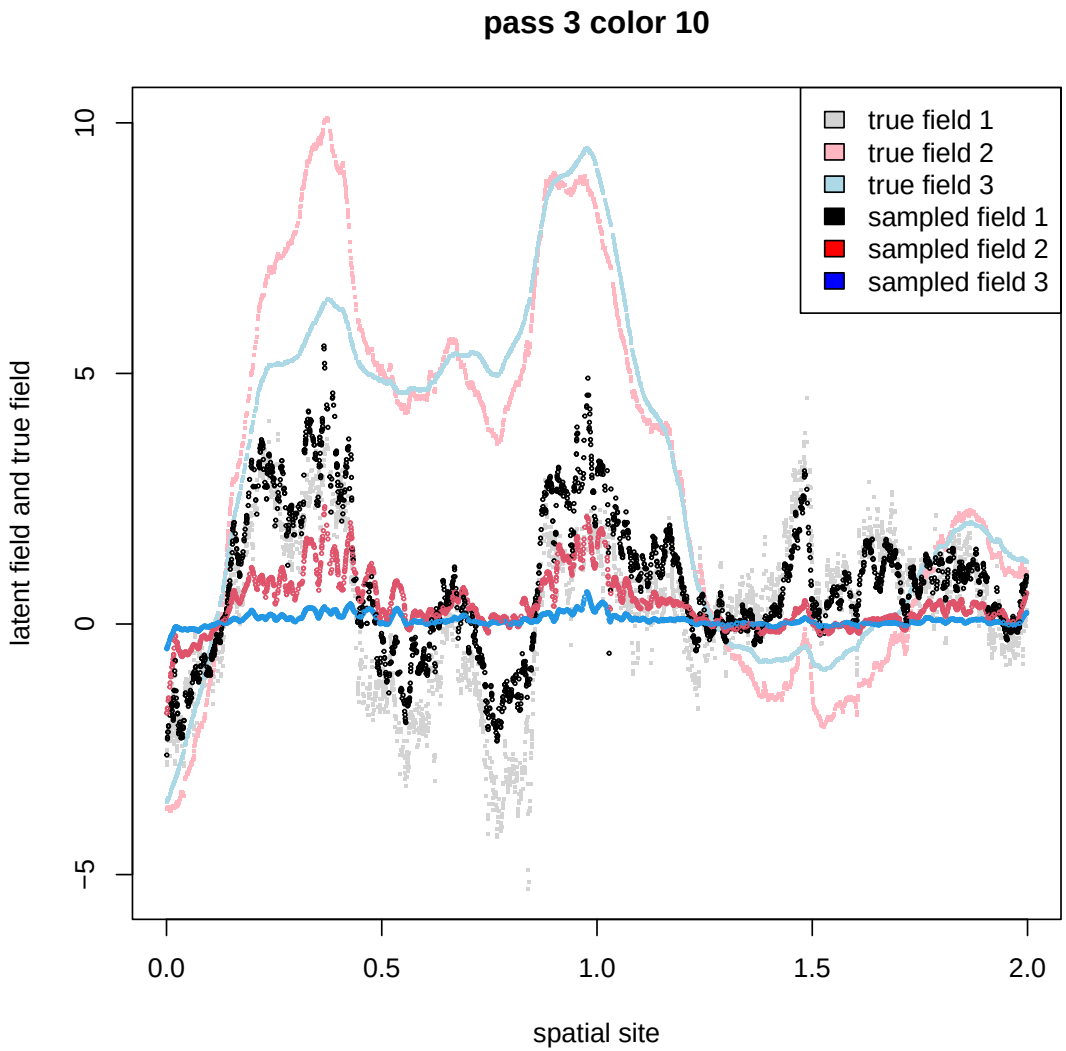
Only blocks (usual algorithm)



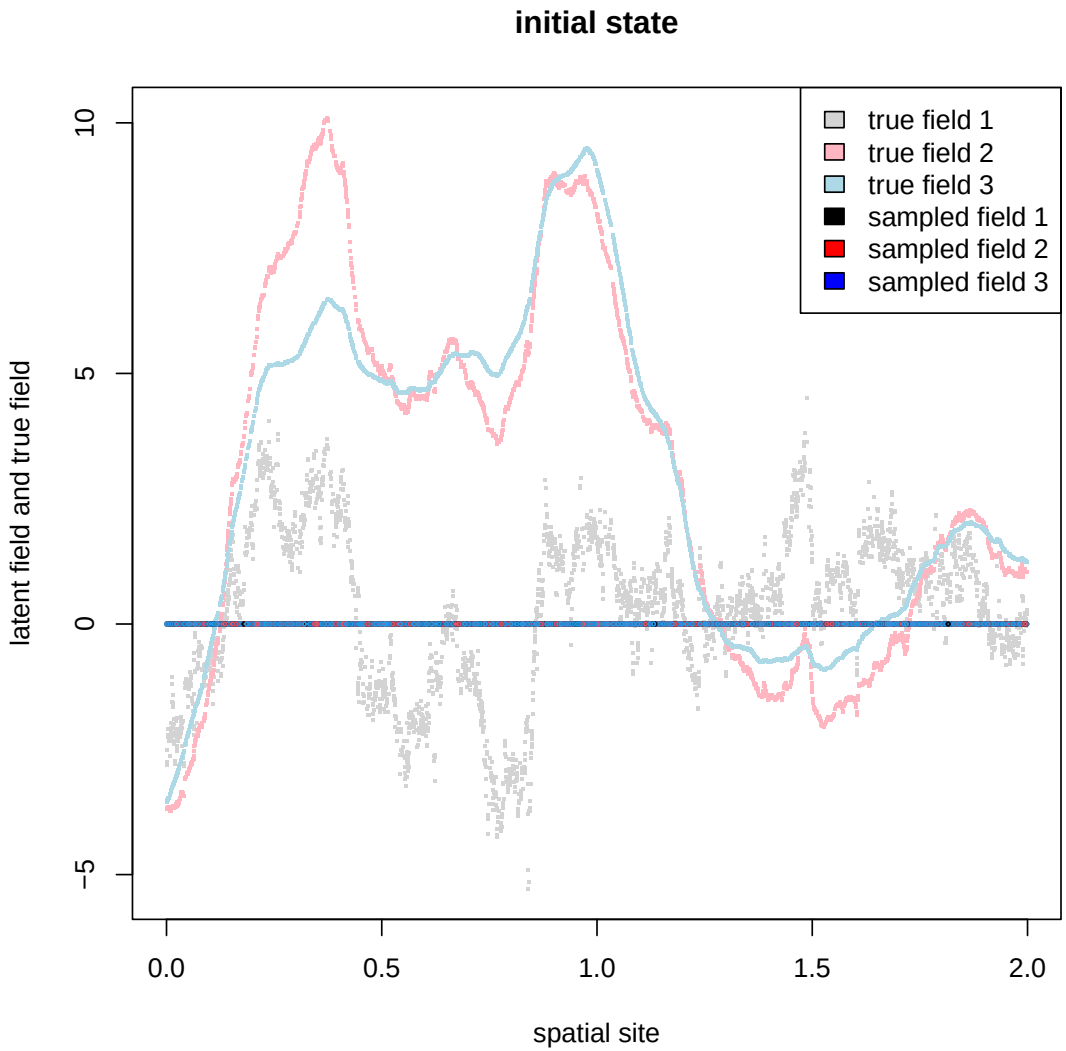
Only blocks (usual algorithm)



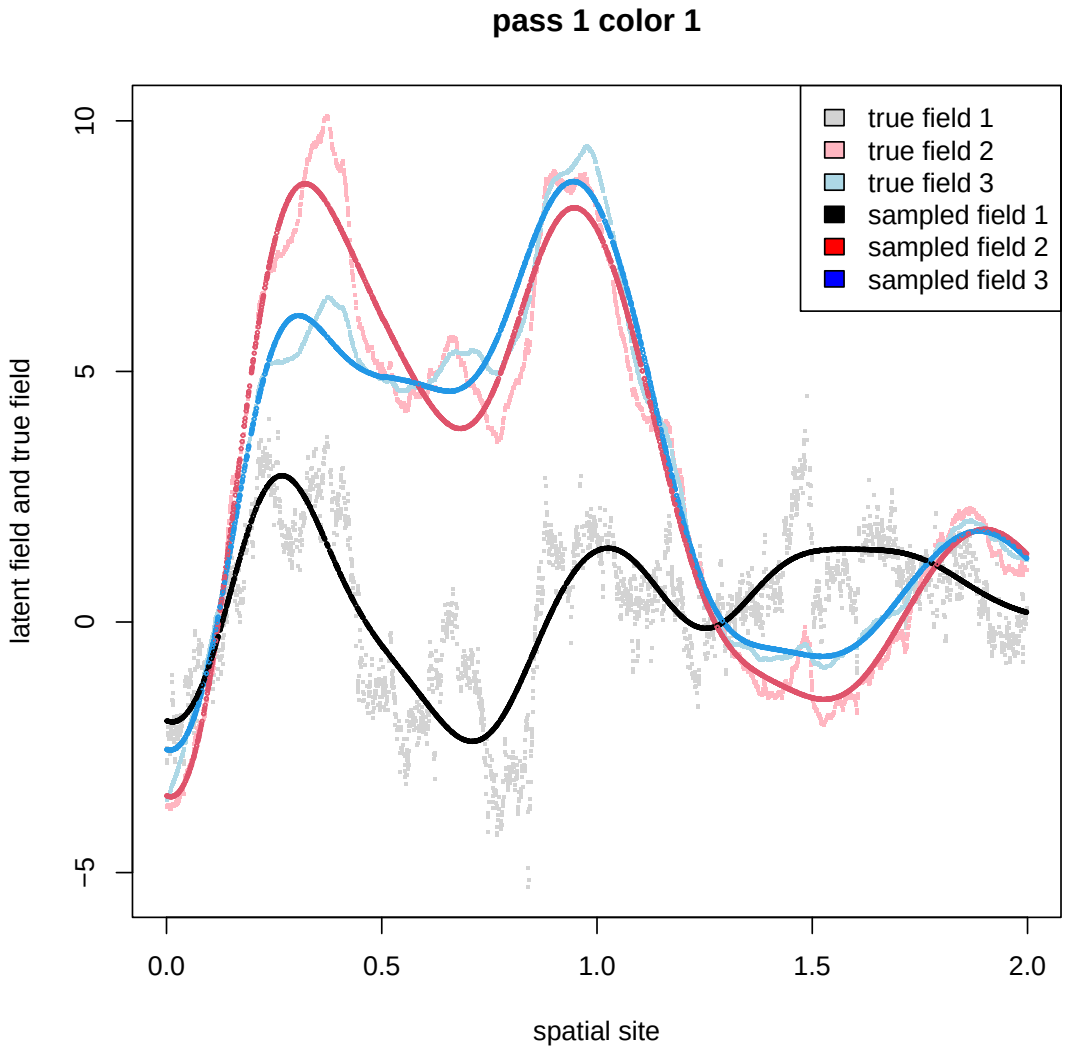
Only blocks (usual algorithm)



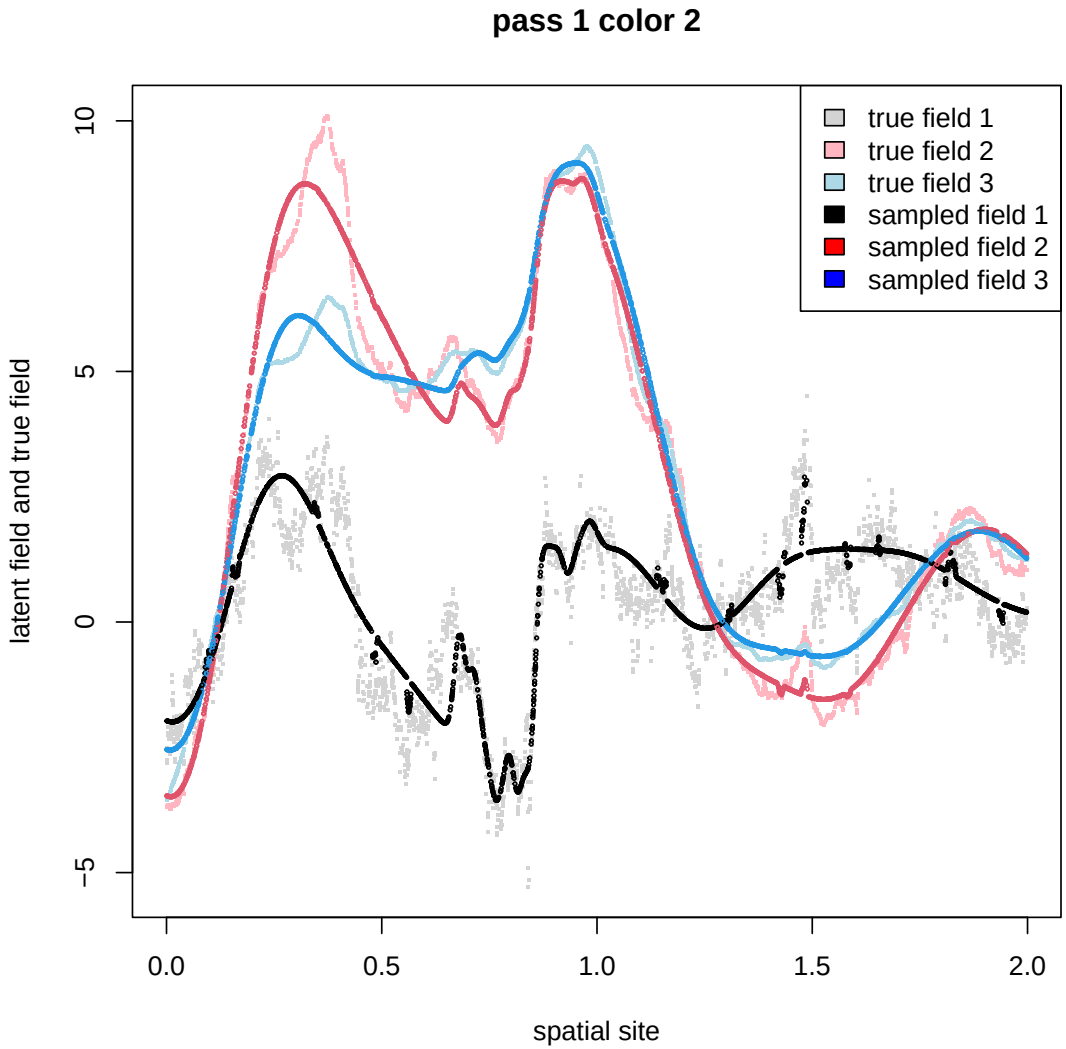
Blocks 'n' Bases



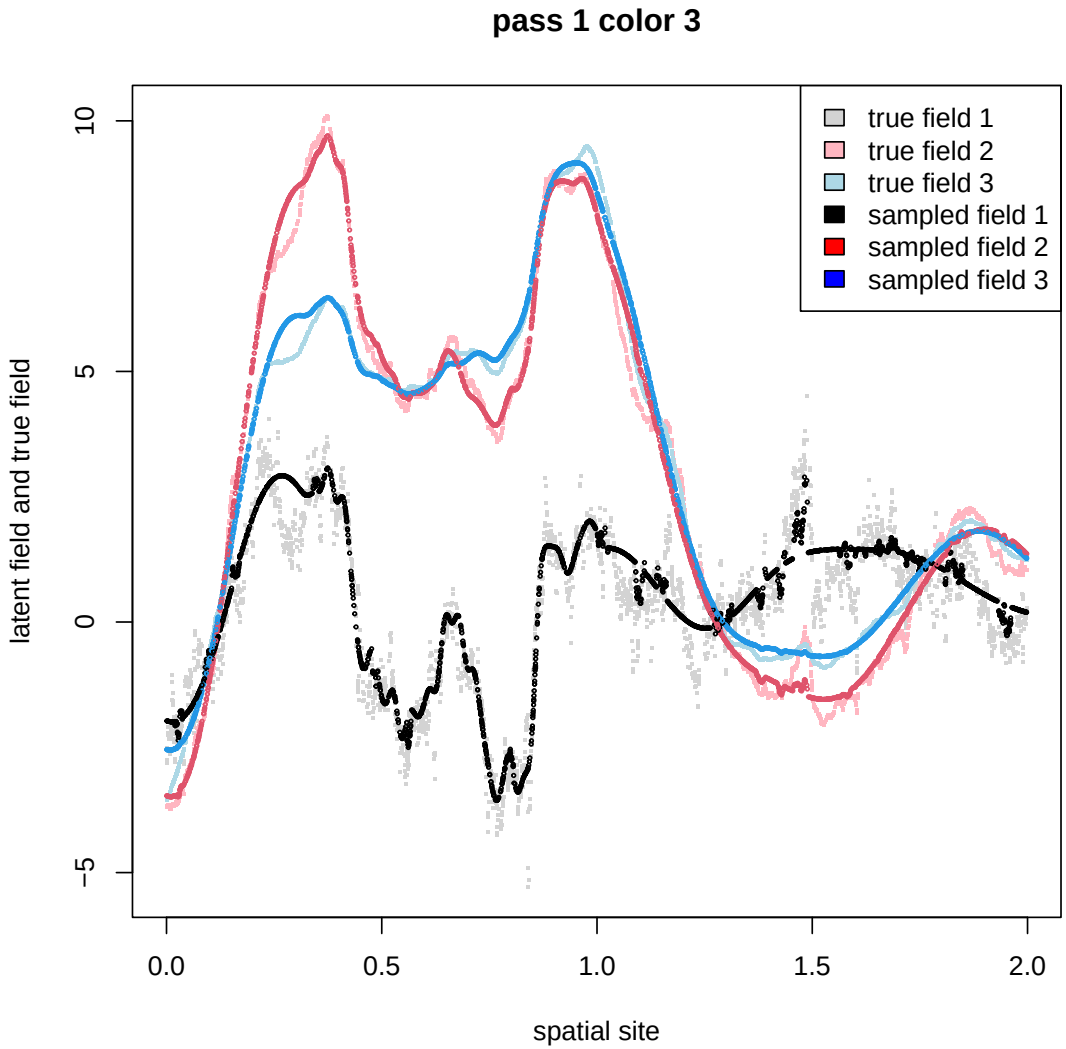
Blocks 'n' Bases



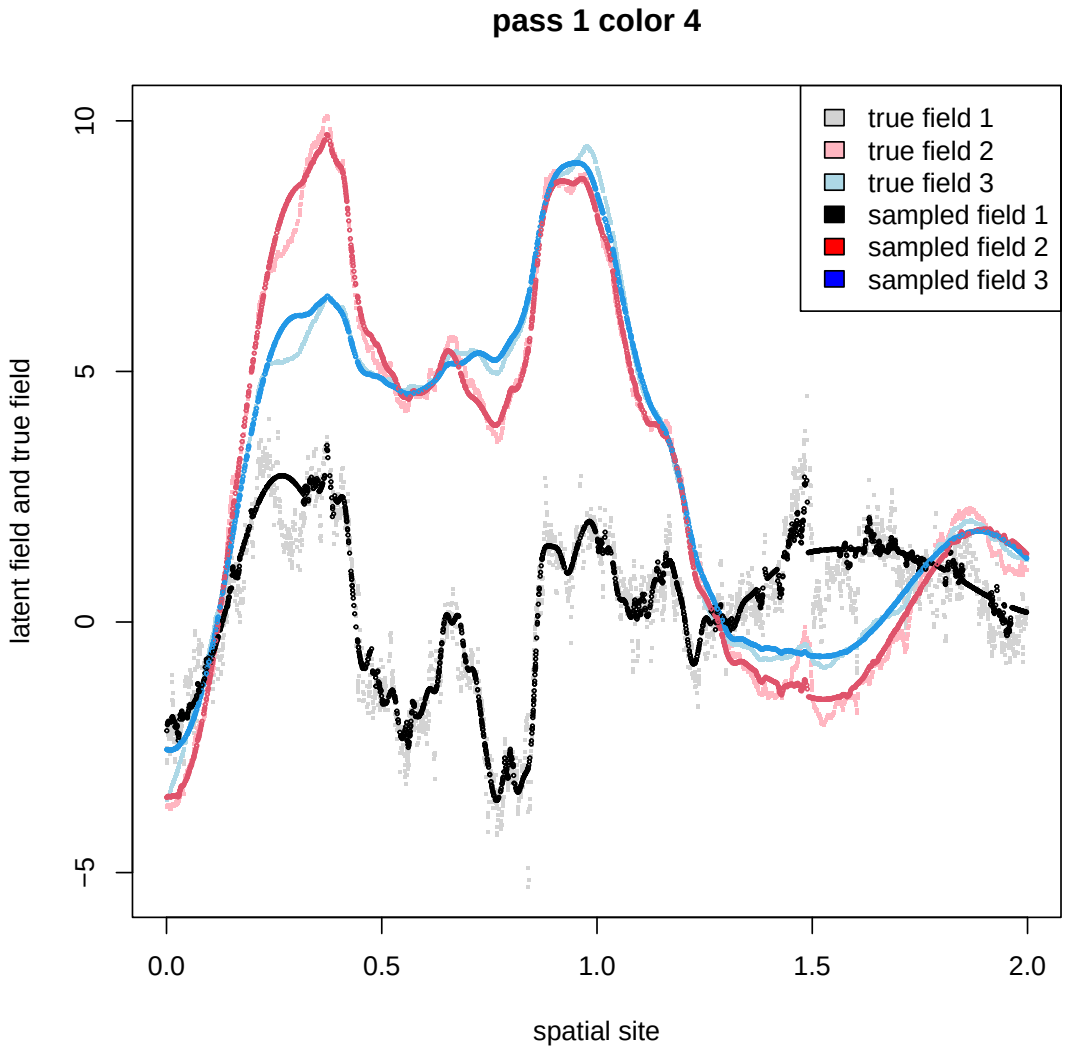
Blocks 'n' Bases



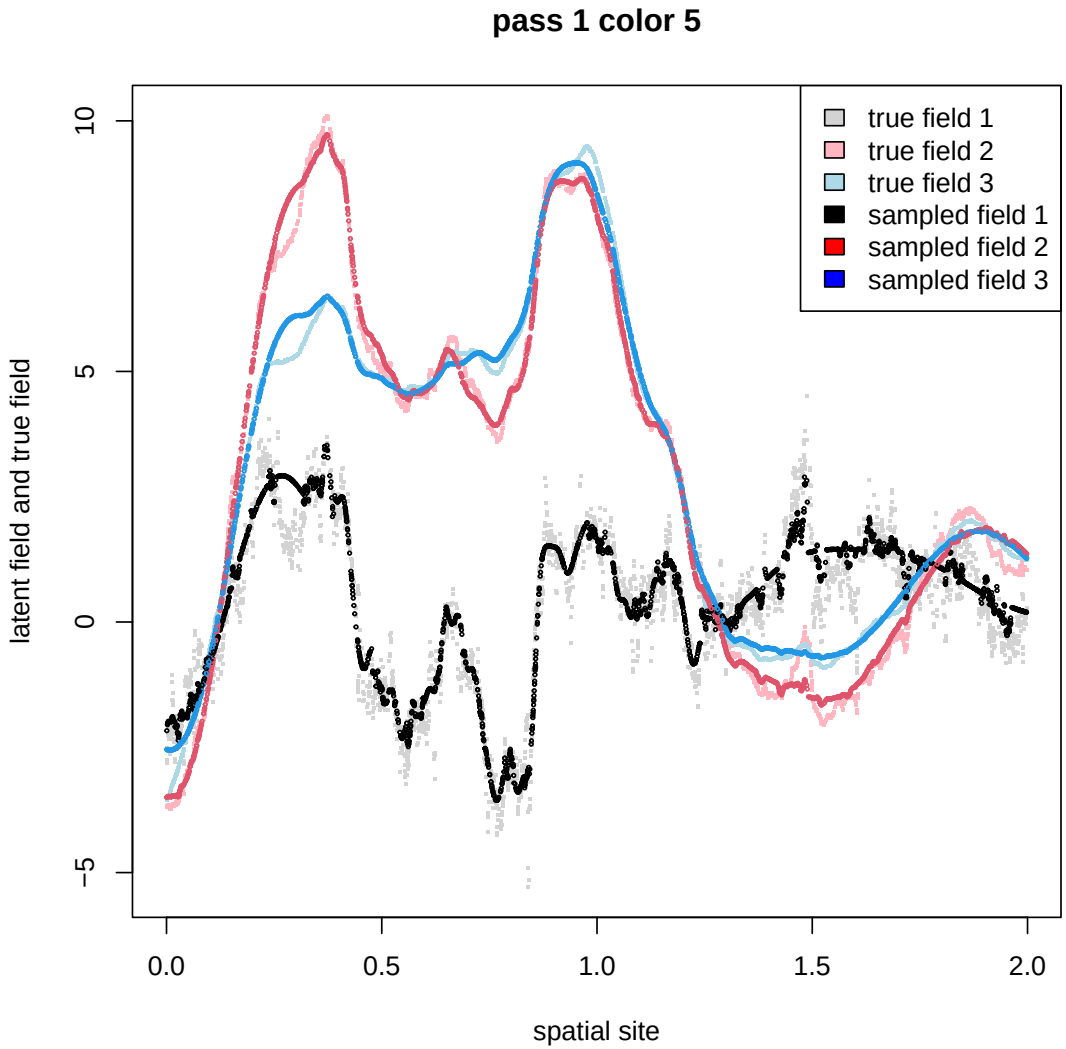
Blocks 'n' Bases



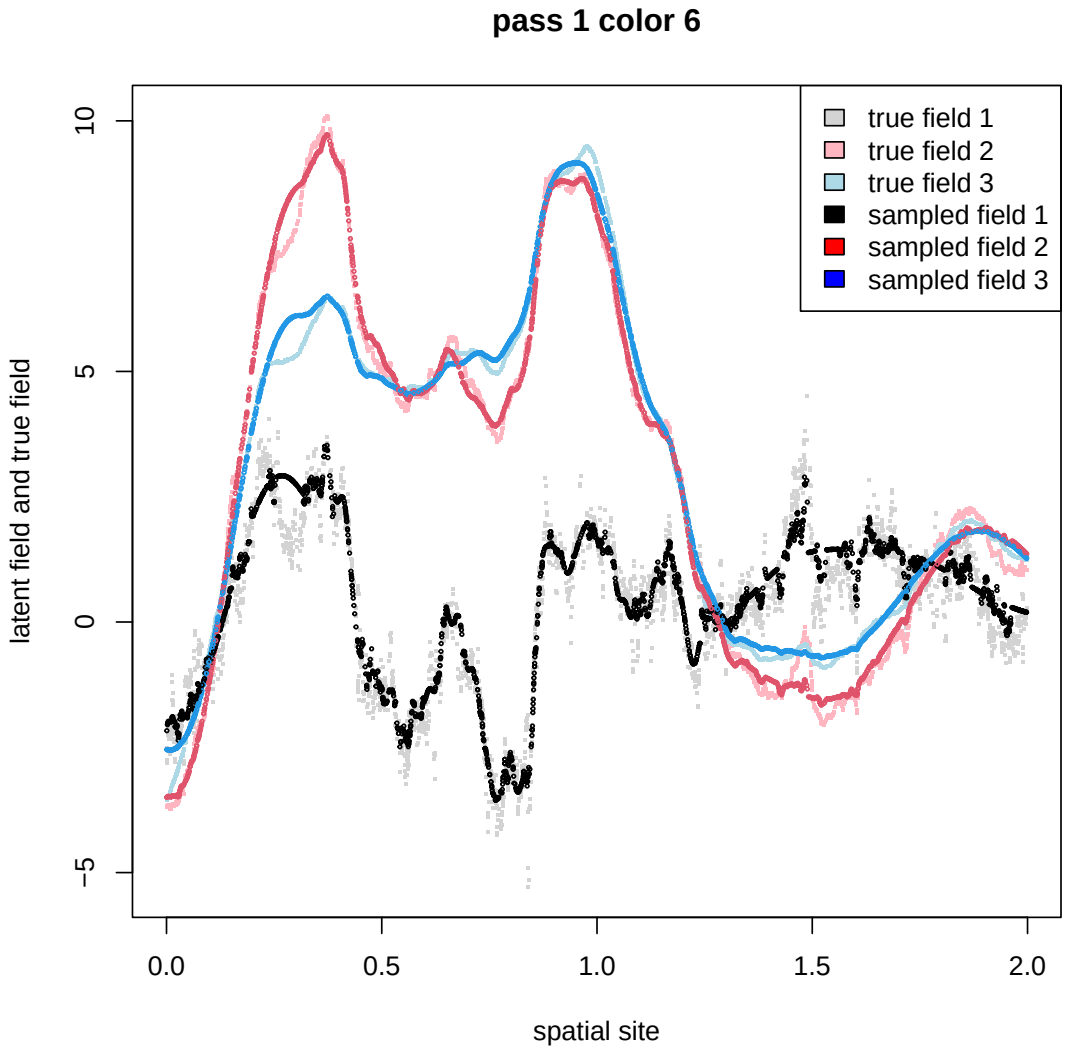
Blocks 'n' Bases



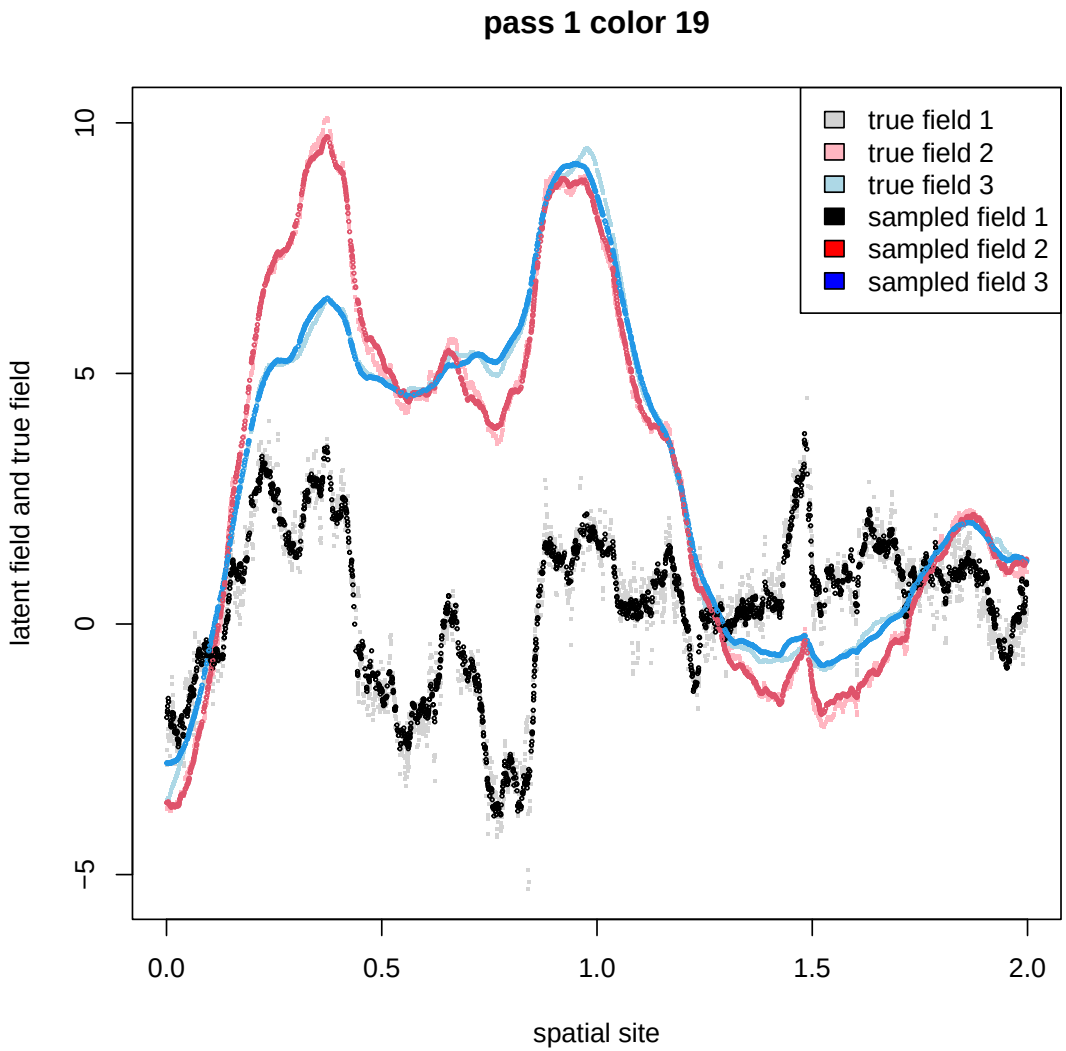
Blocks 'n' Bases



Blocks 'n' Bases



Blocks 'n' Bases



How does it run?

- Use of **parallelized** computing
- Moderate use of RAM

Figure 5: Sampling for a data set of 1.5 M observations

Bilbiography I

Allard, D., Clarotto, L., and Emery, X. (2022). Fully nonseparable gneiting covariance functions for multivariate space-time data.

Coube-Sisqueille, S. and Liquet, B. (2021). Improving performances of mcmc for nearest neighbor gaussian process models with full data augmentation. *Computational Statistics & Data Analysis*, page 107368.