

tour__optim__documentation

Huize Zhang

2/3/2020

Tourr: overview

The computation of a tour involves two major stages: the generating and interpolating stage. Given a current projection basis and its corresponding index value, the generator aims to try to find a better basis with higher index value through optimisation and the better basis found is called the target basis. The interpolator interpolates the current and target basis geodesically. Currently there are three optimisers in the tourr package: `search_better()`, `search_better_random()` and `search_geodesic()`.

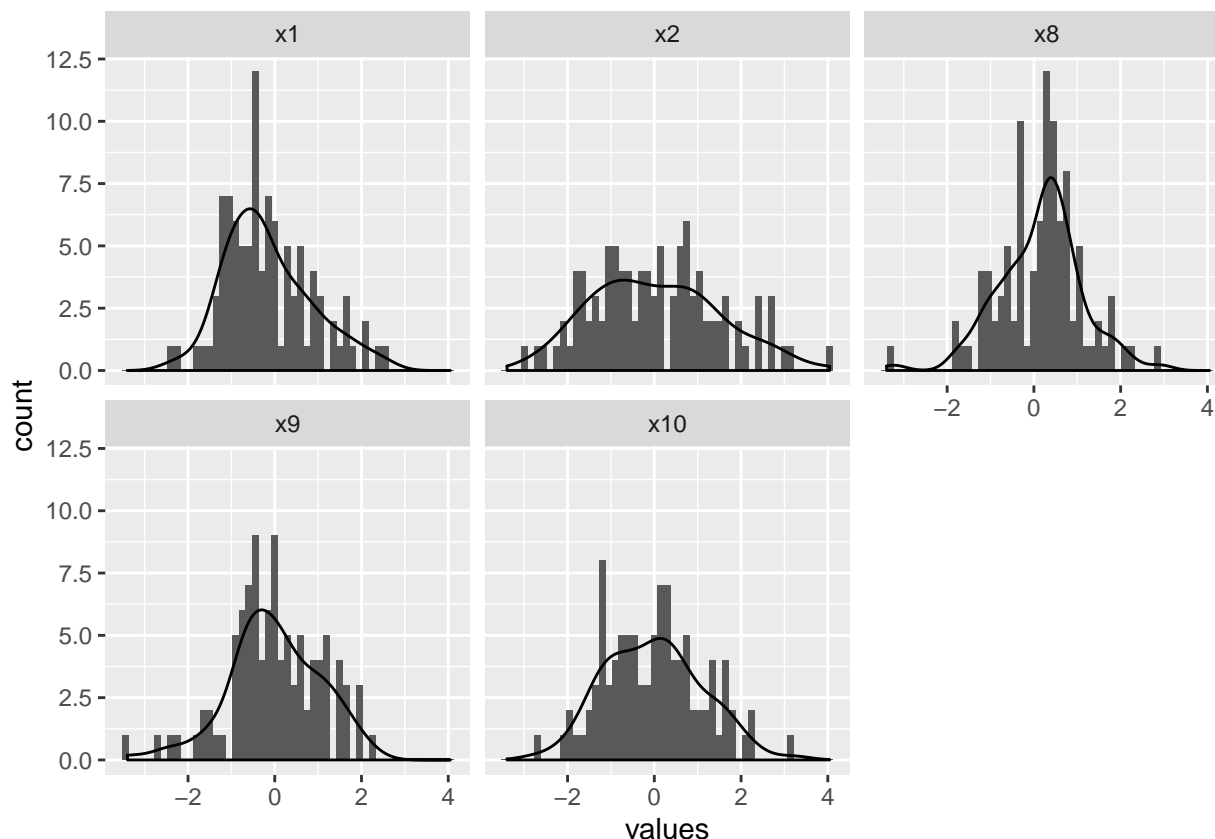
- `search_better()` randomly generates a nearby basis and compute its index value. If the index value is larger than the current index value the new basis will be accepted and used as the current basis for the next iteration. This generate-then-accept process repeats for a pre-determined number of iterations.
- `search_better_random()` modifies on top of `search_better()` to include a random component when the difference between the new index value and the current index value is less than a prescribed epsilon (ϵ). This stochastic component allows the optimisation to jump out of the local maximum.
- `search_geodesic()` uses a two-step procedure to find the basis with better index value. Five candidate bases are first generated randomly and the basis with the largest index value is chosen as the promising direction. This step is called the direction search. The line search step then optimises the basis (angle) along the most promising direction over a 90 degree window to find the target basis. This two-step procedure completes one iteration in the algorithm and the algorithm ceases when the percentage change on the index value exceeds 0.001 or the number of iteration exceeds a pre-determined `max.tries`. Note that the current basis and index value should be updated in each iteration once a better target basis is found and this has yet to be implemented in the `search_geodesic()` function.

Global Object

To understand the searching path of the basis in the optimisation, a global object that records all the bases, index value and auxiliary information would be useful. The `animate_dist()` function has been modified to result a tibble object contains all the information mentioned above, on top of its animated visualisation display.

Demo

Here I present a demonstration of how to use the modified version of tourr package and principle component analysis to understand the optimisation searching path. Simulated data is generated using 4 random variables (`x1`, `x8`, `x9`, `x10`) and a bi-modal distribution (`x2`). The histogram of the five variables are presented below.



Using the `search_geodesic()` optimiser, the first five rows of the global object generated from the tour is shown below.

```
## # A tibble: 6 x 6
##   basis          index_val tries info          col  method
##   <list>          <dbl> <dbl> <chr>          <chr> <chr>
## 1 <dbl[,1] [5 x 1]>    0.747     1 <NA>          x2    geodesic
## 2 <dbl[,1] [5 x 1]>    0.670     1 direction_search x2    geodesic
## 3 <dbl[,1] [5 x 1]>    0.750     1 direction_search x2    geodesic
## 4 <dbl[,1] [5 x 1]>    0.715     1 direction_search x2    geodesic
## 5 <dbl[,1] [5 x 1]>    0.725     1 direction_search x2    geodesic
## 6 <dbl[,1] [5 x 1]>    0.715     1 direction_search x2    geodesic
```

Visualisation tools could help to visually inspect the bases from the tour. Using principle component analysis, we obtain the 2D projection of the 5D basis. The next three visualisations plot the first two principle components and allow us to draw some insights on the `search_geodesic` routine.

- Colored by the stage in which the basis is generated, direction search bases are further away from the interpolating bases than the line search bases. This shows that the direction search provides a broad exploration of the vector space while line search bases are more clustered in a smaller space (plot 1).
- Coloring the interpolating bases by start, target and interpolation on top of the direction and line search bases allows us to observe how the bases updated during the optimisation (plot 2).
- Plot 3 shows the interpolating path only

