Simulations: Low Complexity Methods for Estimation After Selection

Nadav Harel, and Tirza Routtenberg
e-mail: nadavhar@post.bgu.ac.il, tirzar@bgu.ac.il

This document is attached to the simulations Matlab code for [1]. The description of all the simulations in this document appears in the paper. For each simulation, there is a main code that use different functions which will be describe hereinafter:

1 Linear Gaussian model

1.1 Ψ-bias and PSMSE Vs. the number of observations, N

The main `Main_sa_2b_N_2stages.m` file generates Figures 2.a. and 2.b and use the following functions:

- `RPB_estimator.m`: implementation for the described model of the extension of Cohen-Sackrowitz estimator for correlated population.
- `psml_2best.m` : implementation for the described model of our second-best PSML estimator.
- `sa68_iterative.m` implementation for the described model of our SA PSML estimator.
  - `g_sa_68.m`: the SA gradient for the SA-PSML
- `PSFIM_gaussian_sms.m`: returns the empirical PSFIM.

1.2 Probability comparison

In `prob_comparison_linGauss.m` we compared the probability of selection and the pairwise probability of selection. This file generates Fig. 3

1.3 Run-time Vs. the number of parameters, M

The main file is `Main_time_vs_M_2stage.m` and use the same functions as Section 1.1. This file generates Fig. 4.

1.4 Ψ-bias and PSMSE Vs. the number of Monte Carlo simulations, K

The main file is `Main_sa_2b_K_2stages.m` and use the same functions as Section 1.1. This file generates Fig. 5.a., 5.b., and 5.c.

2 Bernoulli model

The main file for this simulation is `bin_2stage_delta.m`. This file generates Figures 6.a. and 6.b and use the following functions:

- `bin_psml_2best.m` : implementation for the described model of our second-best PSML estimator.
- `sa_bin_iterative.m` implementation for the described model of our SA PSML estimator.
  - `g_bin_sa_68.m`: the SA gradient for the SA-PSML
- `PSFIM_binomial.m`: returns the empirical PSFIM.

3 Spectrum estimation after channel selection

The main file for this simulation is `CE_gauss_s_2stages_vs_N.m`. This file generates Figures 7.a. and 7.b and use the following functions:

- `CE2best_psml.m` : implementation for the described model of our second-best PSML estimator.
- `saCE68_iterative.m` implementation for the described model of our SA PSML estimator.
  - `g_se_sa_68.m`: the SA gradient for the SA-PSML
- `PSFIM_CE.m`: returns the empirical PSFIM.

4 Spectrum estimation with "black-box" selection rule

The main file for this simulation is `CE_knn_2stages_vs_N.m`. This file generates Figures 8.a. and 8.b and use the following functions:

- `data_set.m` this function builds the data set for the kNN selection rule
- `Psi_knn.m` implementation of the kNN selection rule
- `sa68_knn_iterative.m` implementation for the described model of our SA PSML estimator.
  - `g_se_knn_sa_68.m`: the SA gradient for the SA-PSML
- `PSFIM_knn.m`: returns the empirical PSFIM.

References

[1] N. Harel and T. Routtenberg, "Low-complexity methods for estimation after parameter selection," *arXiv preprint arXiv:1904.02500*, 2019.