RANKING ROBUSTNESS UNDER SUB-SAMPLING FOR THE SAT COMPETITION 2018

Katalin Fazekas, Daniela Kaufmann, Armin Biere

July 8, 2019, Lisbon

Johannes Kepler University Linz, Austria

JOHANNES KEPLER
UNIVERSITY LINZ

Motivation & Goals

- Benchmarks of SAT competition:
  - To rank solvers of competition
  - To show improvements in papers

Motivation & Goals

- Benchmarks of SAT competition:
  - To rank solvers of competition
  - To show improvements in papers
- Impact of benchmark selection on the final ranking?
  - Measure the robustness of the produced ranking of the competition
  - Range of sub-sampling strategies

INTRODUCTION

SAT Competition 2018

- 41 participating solvers (Main track)

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

Benchmarks:

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

Benchmarks:

- BYOB: bring your own benchmarks

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

Benchmarks:

- BYOB: bring your own benchmarks
- 400 brand new problems are selected

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

Benchmarks:

- BYOB: bring your own benchmarks
- 400 brand new problems are selected
- At most 20 from the same source

SAT Competition 2018

- 41 participating solvers (Main track)
- Ranking: ordering of the solvers by their PAR-2 scores in **increasing** order
- PAR-2 scores:

running times (< 5000 s) $+$ penalty (10000) for unsolved instances

Benchmarks:

- BYOB: bring your own benchmarks
- 400 brand new problems are selected
- At most 20 from the same source
- Same 400 instances used on Parallel and No-limits tracks

Random Sampling & Ranking Robustness

- Simple Random Samples:
  - Select randomly N problems from the 400 problem instances
  - N random numbers between 0 and 399
  - Sampling without replacement

Random Sampling & Ranking Robustness

- Simple Random Samples:
  - Select randomly N problems from the 400 problem instances
  - N random numbers between 0 and 399
  - Sampling without replacement
- Calculate ranking of solvers on that sample
  - How well estimates the original ranking? (Spearman's correlation)

Random Sampling & Ranking Robustness

- Simple Random Samples:
  - Select randomly N problems from the 400 problem instances
  - N random numbers between 0 and 399
  - Sampling without replacement
- Calculate ranking of solvers on that sample
  - How well estimates the original ranking? (Spearman's correlation)
- Ranking Robustness
  - Subsets of instances yield similar rankings

Spearman’s rank correlation

- Determine statistical dependency between rankings

Spearman's rank correlation

- Determine statistical dependency between rankings

$$\rho(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \text{ where } d_i = \mathrm{rank}(x_i) - \mathrm{rank}(y_i)$$

Spearman's rank correlation

- Determine statistical dependency between rankings

$$\rho(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \text{ where } d_i = \mathrm{rank}(x_i) - \mathrm{rank}(y_i)$$

- Indicates how well the statistical dependence between two rank variables can be described using a monotonic function

Spearman's rank correlation

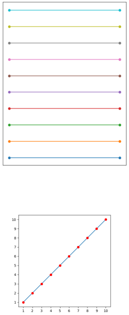
- Determine statistical dependency between rankings

$$\rho(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \text{ where } d_i = \mathrm{rank}(x_i) - \mathrm{rank}(y_i)$$

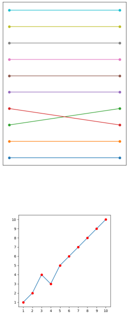
- Indicates how well the statistical dependence between two rank variables can be described using a monotonic function
- High when solvers have a similar rank

Spearman's correlation examples (1)

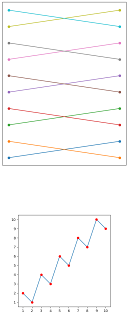
1.0



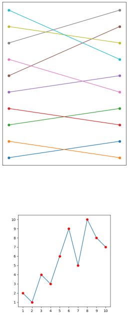
0.988



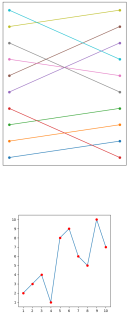
0.939



0.806

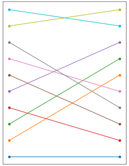


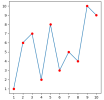
0.697



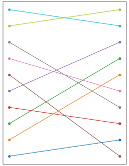
Spearman's correlation examples (2)

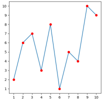
0.539



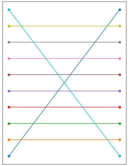


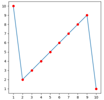
0.455



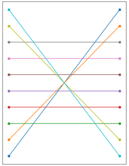


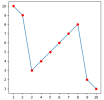
0.018



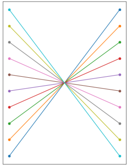


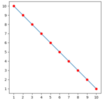
-0.576



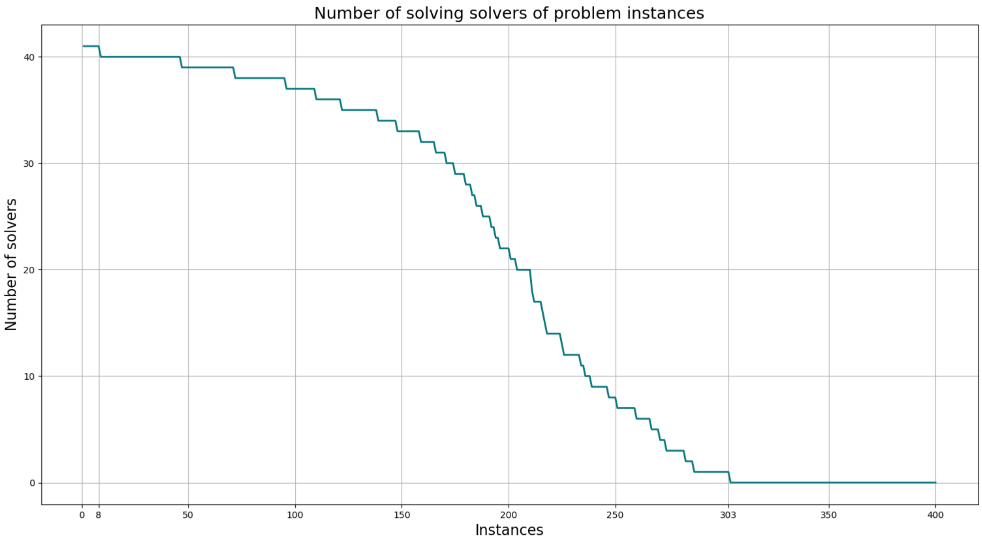


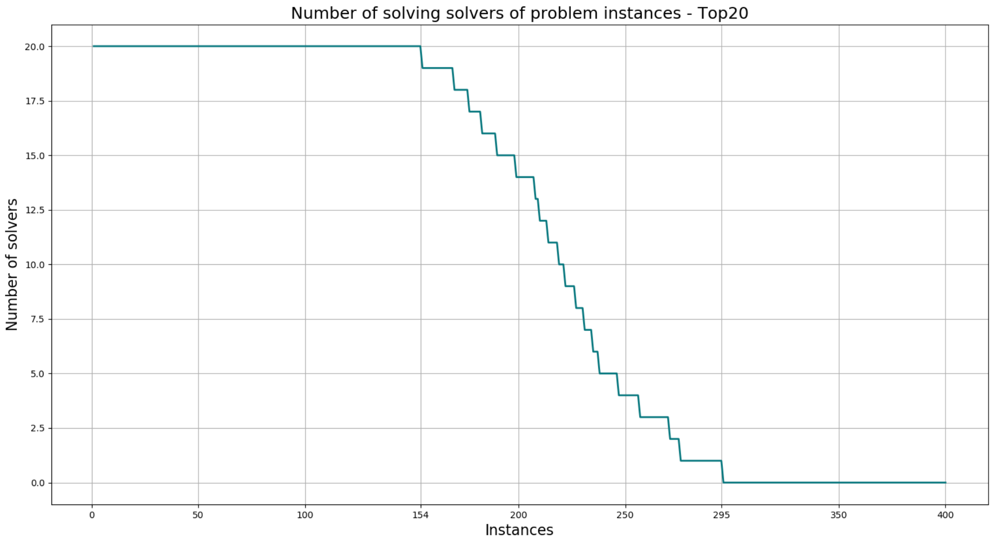
-1.0





RANKING ROBUSTNESS UNDER RANDOM SAMPLING


Number of solving solvers of problem instances
Number of solvers
Instances
0
10
20
30
40
0
8
50
100
150
200
250
303
350
400


Number of solving solvers of problem instances - Top20
Number of solvers
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
50
100
154
200
250
295
350
400
Instances

Preprocessing – Data cleaning

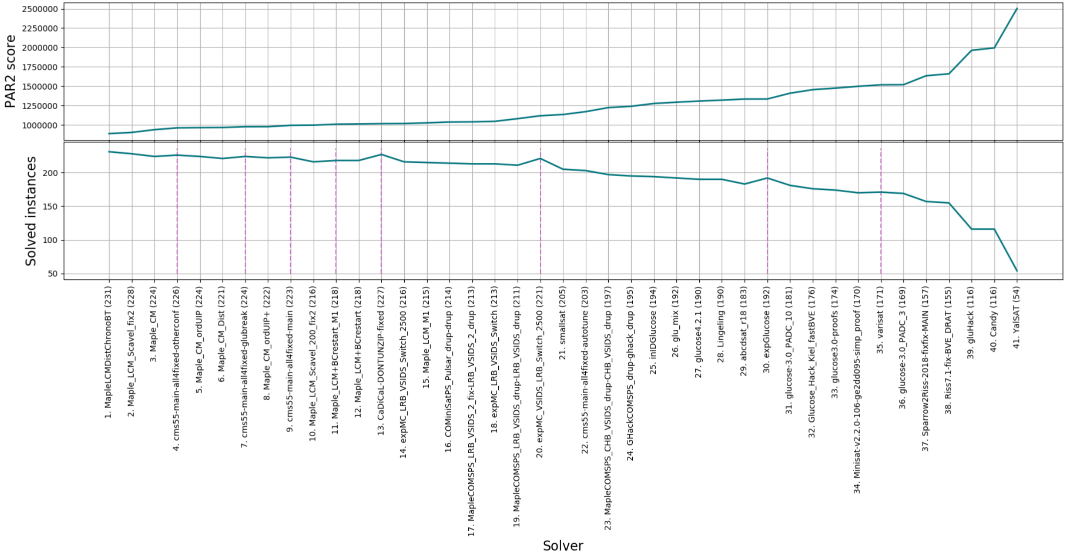
- Remove unsolved instances
  - their removal yields a rank of solvers with correlation 1 to the competition rank

Preprocessing – Data cleaning

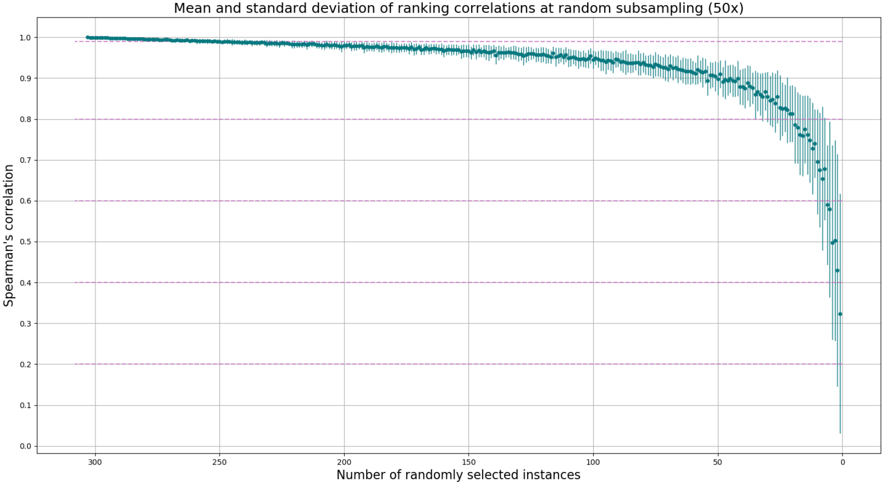
- Remove unsolved instances
  - their removal yields a rank of solvers with correlation 1 to the competition rank
- Fix answer for `sted1_0x1e3-100.cnf.bz2` as SAT
  - `smallsat` solved but did not answer satisfiability

Preprocessing – Data cleaning

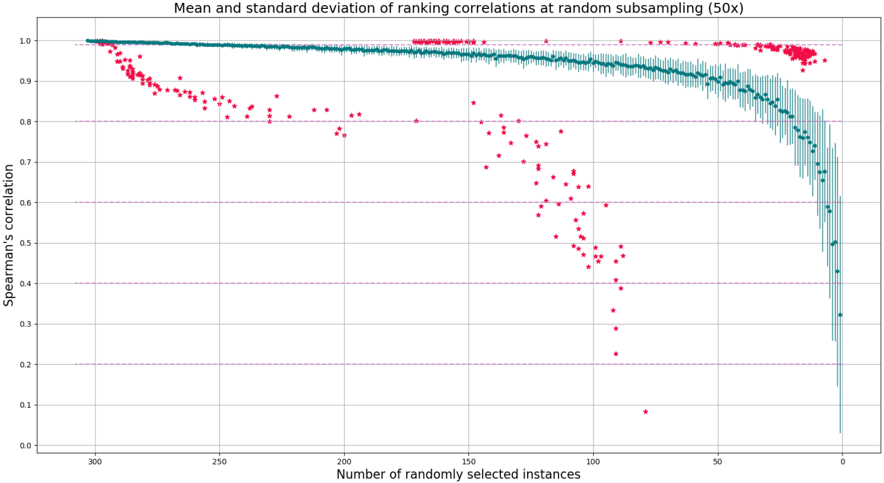
- Remove unsolved instances
  - their removal yields a rank of solvers with correlation 1 to the competition rank
- Fix answer for `sted1_0x1e3-100.cnf.bz2` as SAT
  - `smallsat` solved but did not answer satisfiability
- Penalized time out even when satisfiability is answered
  - 5 instances (on one verification failed)


PAR2 score
2500000
2250000
2000000
1750000
1500000
1250000
1000000
Solved instances
200
150
100
50
1. MapleLCMDistChronoBT (231)
2. Maple_LCM_Scavel_fix2 (228)
3. Maple_CM (224)
4. cms55-main-all4fixed-otherconf (226)
5. Maple_CM_ordUIP (224)
6. Maple_CM_Dist (221)
7. cms55-main-all4fixed-glubreak (224)
8. Maple_CM_ordUIP+ (222)
9. cms55-main-all4fixed-main (223)
10. Maple_LCM_Scavel_200_fix2 (216)
11. Maple_LCM+BCrestart_M1 (218)
12. Maple_LCM+BCrestart (218)
13. CaDiCaL-DONTUNZIP-fixed (227)
14. expMC_LRB_VSIDS_Switch_2500 (216)
15. Maple_LCM_M1 (215)
16. COMiniSatPS_Pulsar_drup-drup (214)
17. MapleCOMSPS_LRB_VSIDS_2_fix-LRB_VSIDS_2_drup (213)
18. expMC_LRB_VSIDS_Switch (213)
19. MapleCOMSPS_LRB_VSIDS_drup-LRB_VSIDS_drup (211)
20. expMC_VSIDS_LRB_Switch_2500 (221)
21. smallsat (205)
22. cms55-main-all4fixed-autotune (203)
23. MapleCOMSPS_CHB_VSIDS_drup-CHB_VSIDS_drup (197)
24. GHackCOMSPS_drup-ghack_drup (195)
25. inIDGlucose (194)
26. glu_mix (192)
27. glucose4.2.1 (190)
28. Lingeling (190)
29. abcdsat_r18 (183)
30. expGlucose (192)
31. glucose-3.0_PADC_10 (181)
32. Glucose_Hack_Kiel_fastBVE (176)
33. glucose3.0-proofs (174)
34. Minisat-v2.2.0-106-ge2dd095-simp_proof (170)
35. varisat (171)
36. glucose-3.0_PADC_3 (169)
37. Sparrow2Riss-2018-fixfix-MAIN (157)
38. Riss7.1-fix-BVE_DRAT (155)
39. gluHack (116)
40. Candy (116)
41. YalSAT (54)
Solver

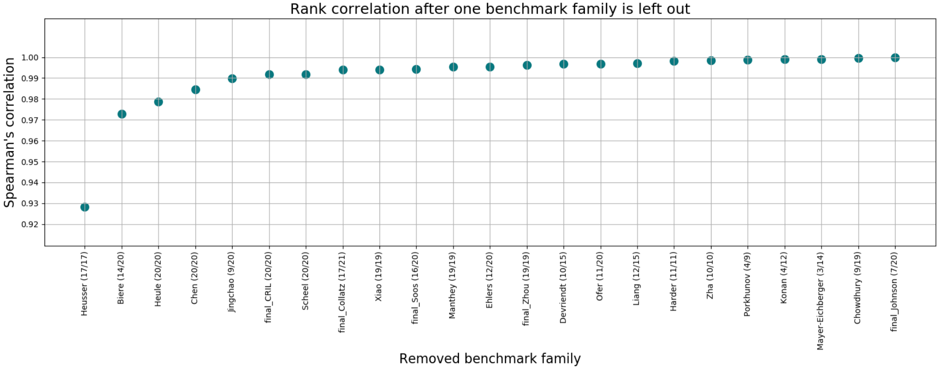
Ranking Robustness and Random sampling



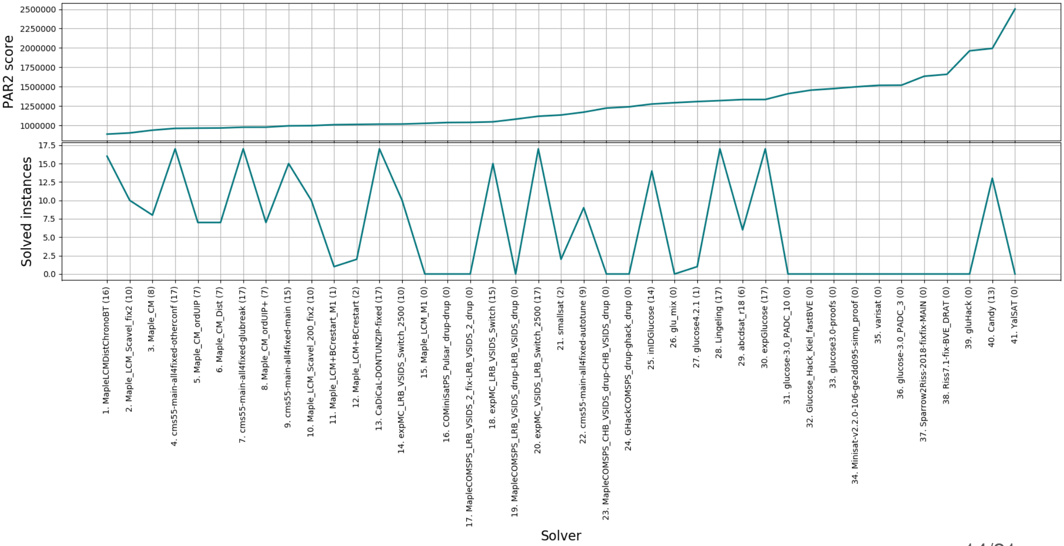
Ranking Robustness and GeneticAlg-Random sampling



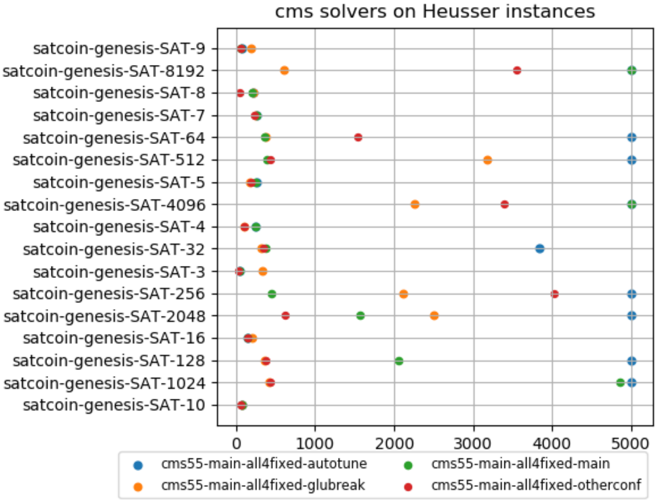
Ranking Robustness and Benchmark families



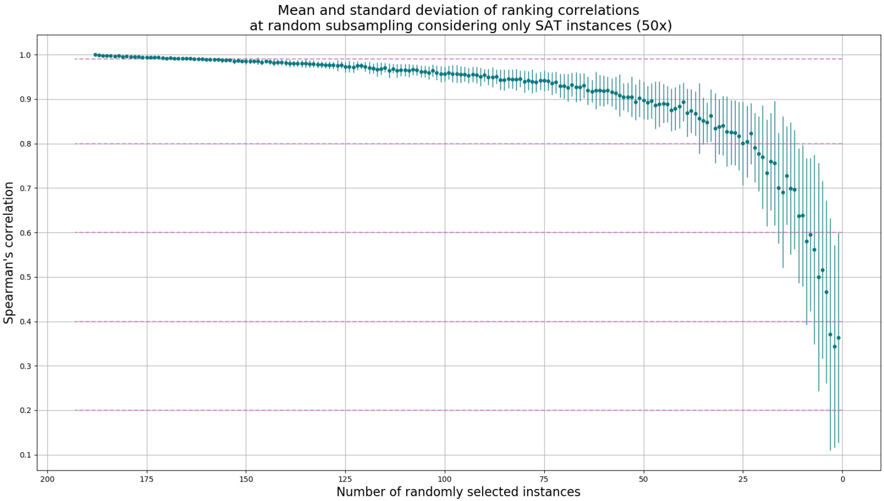
Ranking Robustness and Heusser benchmarks



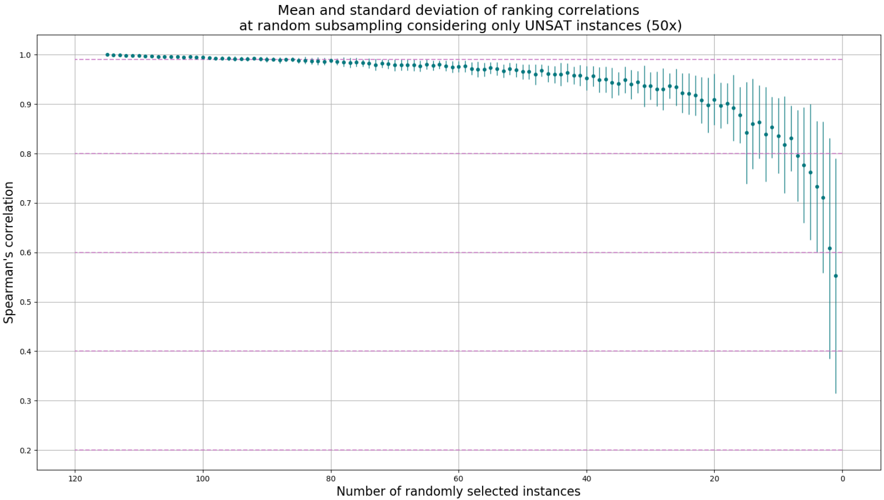
Ranking Robustness and Heusser benchmarks



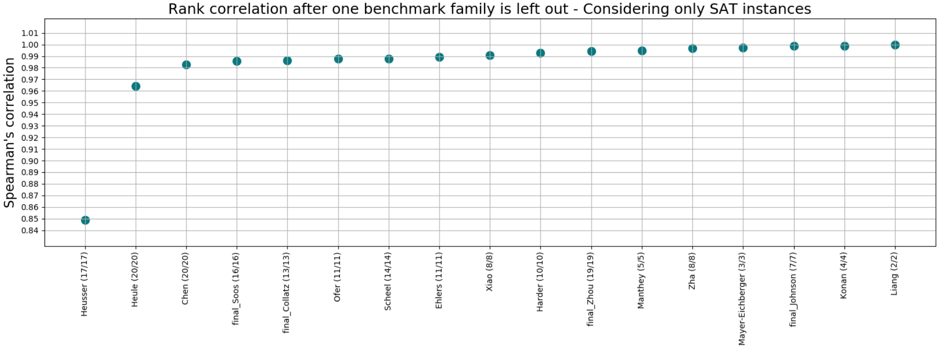
Ranking Robustness and Random sampling – SAT



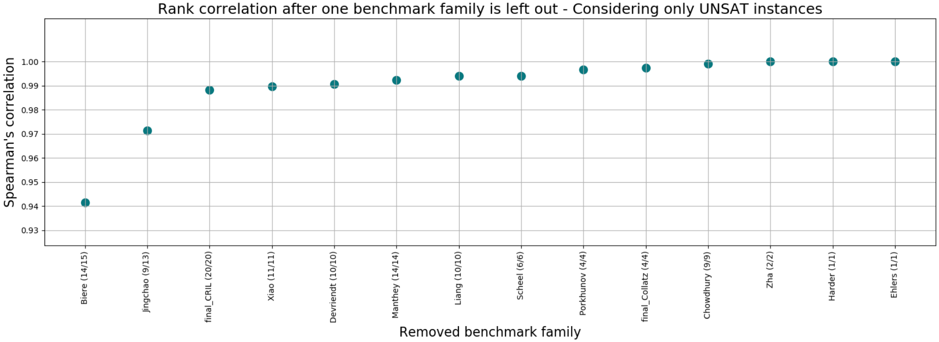
Ranking Robustness and Random sampling – UNSAT



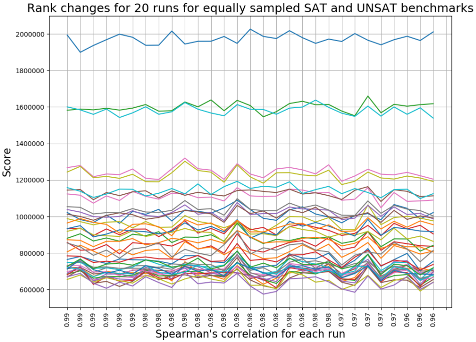
Ranking Robustness and Benchmark families – SAT

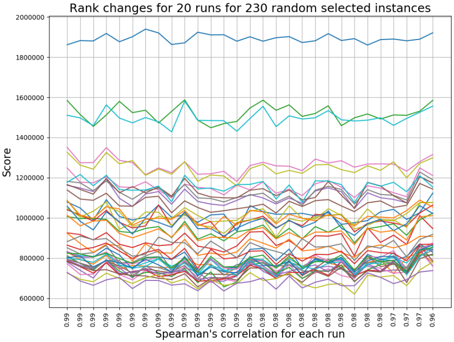


Ranking Robustness and Benchmark families – UNSAT



Ranking Robustness and Satisfiability





CONCLUSION

Summary

- Random sampling allows to remove a large fraction of benchmarks without changing the competition ranking much

Summary

- Random sampling allows to remove a large fraction of benchmarks without changing the competition ranking much
  - Benefits in regression testing, portfolio solving

Summary

- Random sampling allows to remove a large fraction of benchmarks without changing the competition ranking much
  - Benefits in regression testing, portfolio solving

- Is the selection so good or the solvers so robust?

Thank you for your attention!