Sequential Parameter Optimization for Symbolic Regression

Thomas Bartz-Beielstein Oliver Flasch Martin Zaefferer
{firstname.lastname}@fh-koeln.de

SPOT SEVEN
Cologne University of Applied Sciences
Faculty for Computer and Engineering Science

July 2012

SPOT SEVEN

Agenda

Goals

Introducing RGP

Introducing SPOT

Experiments with SPOT

Advanced SPOT Features

SPOT SEVEN

Goals of this Work

- general (conceptual) framework for empirical analysis of
  - GP system components and their ...
  - ... influence on GP system performance
- based on *Experimental Research in Evolutionary Computation* (Bartz-Beielstein, 2006)
  - principles for obtaining statistically validated results ...
  - ... of high reproducibility
- prototypic software implementation of this framework
  - automation of much of the necessary repeated work
  - standardized result analysis
  - available as open-source software based on the R environment

SPOT SEVEN

R: Programming Language for Statistics

- R is "GNU S", a freely available language and environment for statistical computing and graphics.
- R provides a wide variety of statistical and graphical techniques:
  - linear and nonlinear modelling,
  - statistical tests,
  - time series analysis,
  - classification,
  - clustering, etc.
- Useful platform for GP, providing:
  - flexible interactive environment
  - fast expression maniplutaion and evaluation
  - powerful visualization tools
  - tools for parallel computing
- See R project homepage `http://cran.r-project.org/` for further information.

SPOT SEVEN

RGP Overview

- modular GP implementation in R
  - *simplicity beats complexity*
  - *convention over configuration*
- large feature set
  - multiple search heuristics (Pareto GP, TinyGP, . . .)
  - multiple representations (tree GP and linear GP)
  - multiple sets of variation operators
  - support for strongly-typed GP

  ↪ complex parameterization
- performance-critical functions *also* implemented in C
- comprehensive documentation
- Freely available (GPL-2) on CRAN:
  `install.packages("rgp")`
- See RGP project homepage `http://rsymbolic.org/`
  for details and "bleeding edge releases".

SPOT SEVEN

GP Search Heuristics

- concrete search strategy employed by a GP system
- independent of the concrete GP search space

↪ decouple the search heuristic from the search space

- GP search heuristic components:
  - selection strategy
  - variation pipeline (order of variation operator application)
  - diversity preservation
- GP system components independent of the search heuristic:
  - GP individual representation
  - GP individual initialization and variation (mutation and crossover)
  - GP individual evaluation
- examples of GP search heuristics:
  - classical single-objective steady-state EAs with tournament selection
  - modern multi-objective steady-state heuristics

SPOT SEVEN

TinyGP

- popular small GP implementation mainly used in teaching
- steady-state single-objective search heuristic with tournament selection
- no direct means of diversity preservation
- included as a reference with well-known performance characteristics

Table : Parameters of the TinyGP search heuristic.

	Variable (Symbol)	*Domain*	*Default*
Population Size	`mu` (μ)	$\mathbb{N}$	300
Tournament Size	`tournamentSize` ($s_{\text{tournament}}$)	$\mathbb{N}$	2
Recombination Probability	`recombinationProbability` (p_{rec})	$[0, 1]$	0.9

Generational Multi-Objective GP (GMOGP)

- based on the well-known multi-objective generational $(\mu + \lambda)$ EA NSGAII
- coarsely scalable complexity through optional selection criteria: individual age, individual complexity
- diversity preservation through age-layering
- included as search heuristic with scalable complexity

Table : Parameters of the GMOGP search heuristic.

	Variable (Symbol)	*Domain*	*Default*
Population Size	`mu` (μ)	$\mathbb{N}$	300
Children per Generation	`lambda` (λ)	$\mathbb{N}$	20
Recombination Probability	`recombinationProbability` $(p_{\texttt{rec}})$	$[0, 1]$	0.1
Enable Complexity Criterion	`complexityCriterion`	$\mathbb{B}$	`true`
Enable Age Criterion	`ageCriterion`	$\mathbb{B}$	`true`
New Individuals per Generation	`nu` (ν)	$\mathbb{N}_0$	1

Experiment Setup

- *Research goal:* Quantify influence of RGP search heuristic parameters on algorithm performance (single algorithm, single problem).

Table : RGP parameters independent of the search heuristic.

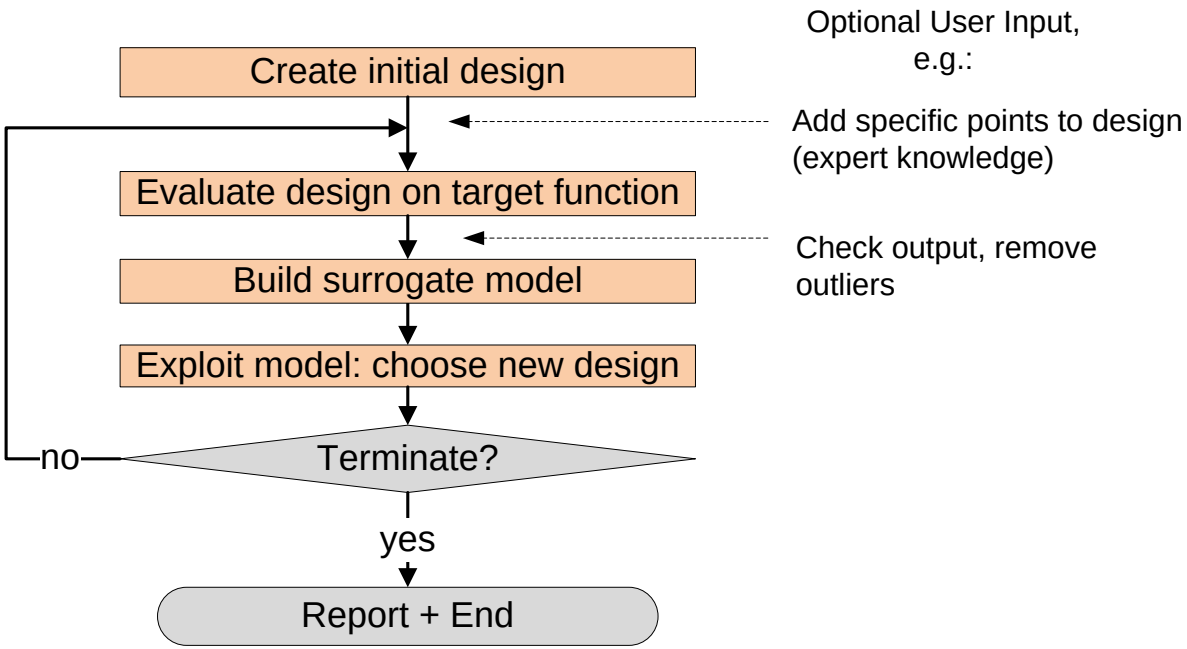
Problem	Symbolic Regression of $f(x) := \sin(x) + \cos(2 \cdot x)$
Fitness Cases	200 equidistant samples in $[0, 4 \cdot \pi]$
Error Measure	sample RMSE
Function Set	$\{+, -, \cdot, \div\}$
Input Variable Set	$\{x\}$
Constant Set	uniform random constants in $[-1, 1]$
Individual Size Limit	64
Mutation Operator Set	{ insert/delete subtree, change function/constant }
Crossover Operator Set	{ random subtree crossover }
Time Budget per GP Run	5 minutes
Initial Experiment Design Size	10
Number of Sequential GP Runs	100

SPOT SEVEN

SPOT Introduction

Sequential Parameter Optimization [?] Toolbox (SPOT[1])

- Based on statistical methods and Design of Experiment



[1]SPOT and all other used R packages can be retrieved from the CRAN homepage, i.e. http://cran.r-project.org.

SPOT Setup

Table : Parameters influencing SPOT performance

Initial Design Size	10
Initial Design Repeats	2
Maximum Repeats	5
Budget (GP-Runs)	100
Old Best Size	3
New Design Size	1
Budget Allocation	Linearly increasing
Surrogate Model	Kriging Model
Surrogate Optimization Method	CMA-ES
Surrogate Optimization Budget	1000

SPOT SEVEN

Interfacing RGP and SPOT I

```
> spotRgpTargetFunction <- function(x, time = 120) {
+   populationSize <- x[1]
+   tournamentSizePercentage <- x[2]
+   crossoverProbability <- x[3]
+
+   ## [...]  ## repair parameters as necessary
+
+   ## [...]  ## define problem data, to be solved by symb. regress.
+
+   ## [...]  ## run symbolic regression with parameters
+
+   ## [...]  ## calculate RMSE (fitness) of best individual in population
+
+   return (bestFitness)
+ }
```

Configuring and starting SPOT

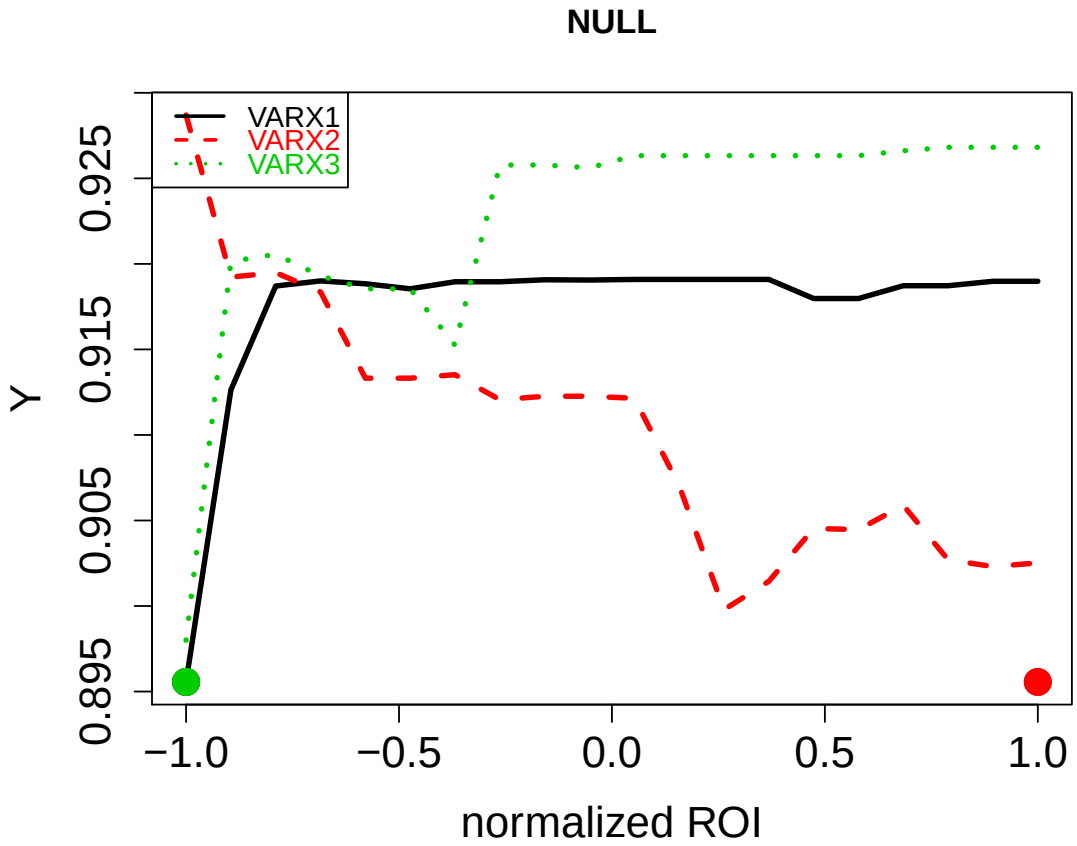
```
> conf <- list(alg.func = spotRgpTargetFunction,
+     alg.roi = spotROI(lower  = c(10,   0.0,     0.0),
+                       upper  = c(1000, 1.0,     1.0),
+                       type   = c("INT", "FLOAT", "FLOAT")),
+     alg.seed = 1,
+     spot.seed = 0,
+     seq.predictionModel.func = "spotPredictForrester",
+     seq.predictionOpt.func = "spotPredictOptMulti",
+     seq.predictionOpt.budget = 1000,
+     seq.predictionOpt.method = "cmaes",
+     io.verbosity = 3,
+     report.interactive = TRUE,
+     spot.ocba = FALSE, # no variance at some points causes OCBA to crash
+     init.design.size = 10,
+     init.design.repeats = 2,
+     seq.design.oldBest.size = 3,
+     seq.design.size = 1000,
+     seq.design.new.size = 1,
+     seq.design.maxRepeats = 5,
+     auto.loop.nevals = 100)

> time <- 600 # set per RGP run time budget (in seconds), default is 120
> res <- spot(spotConfig = conf, time = time)
```

SPOT SEVEN

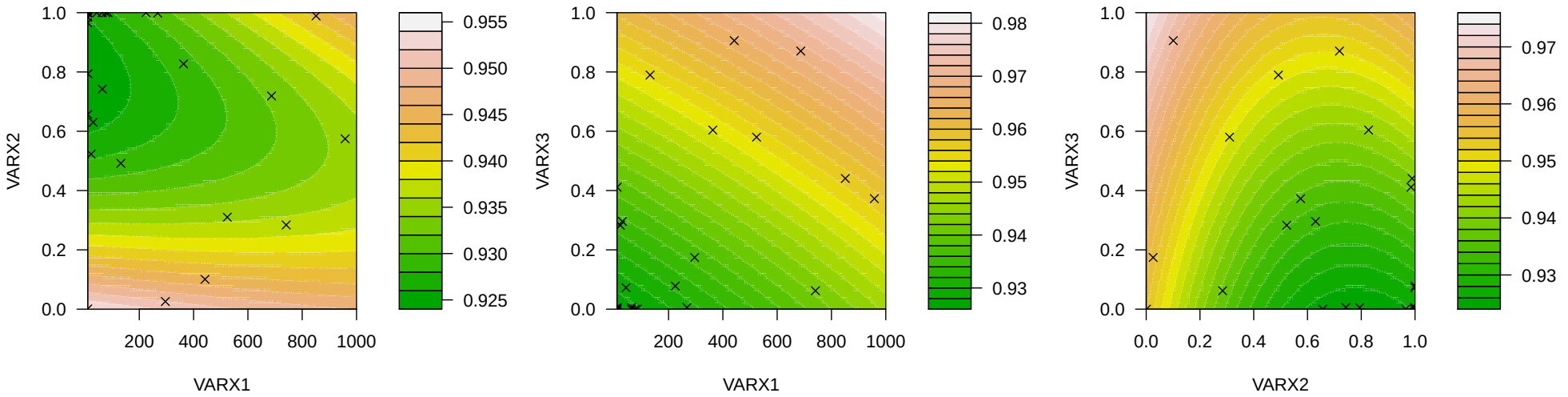
Results of the automated SPOT run

```
> spot(spotConfig=append(
+  list(report.func="spotReportSens"),
+  res),spotTask="rep")
```

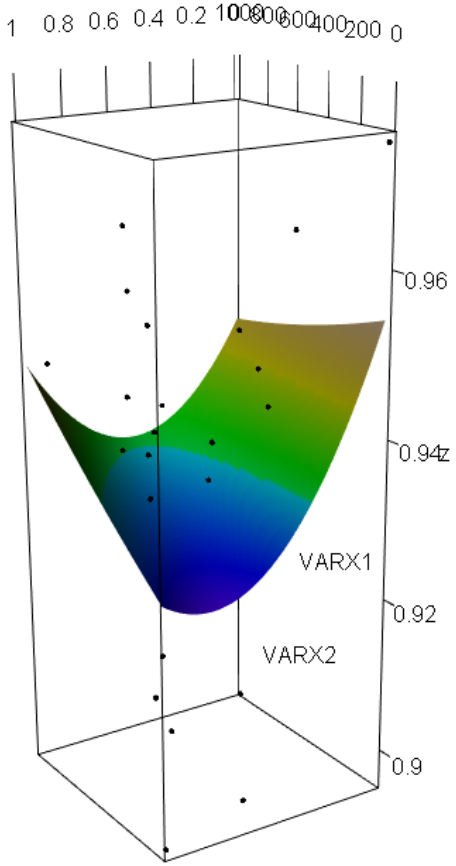


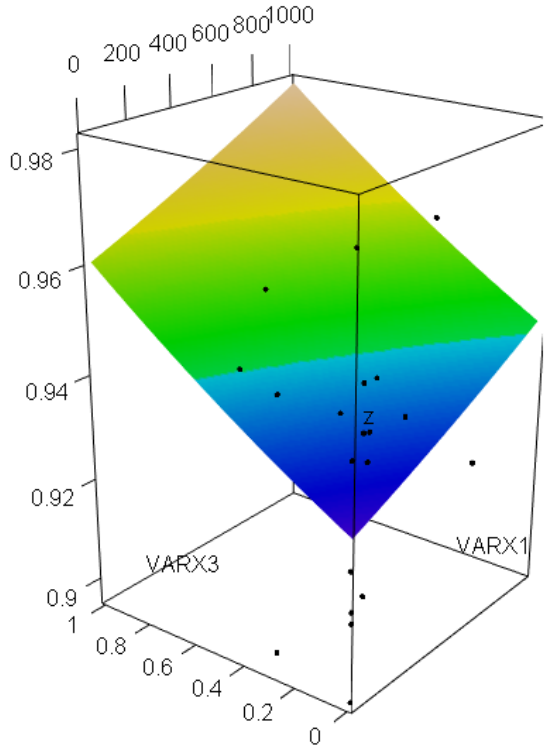
VARX1 (pop. size) VARX2 (tournament size %) VARX3 (crossover prob.)

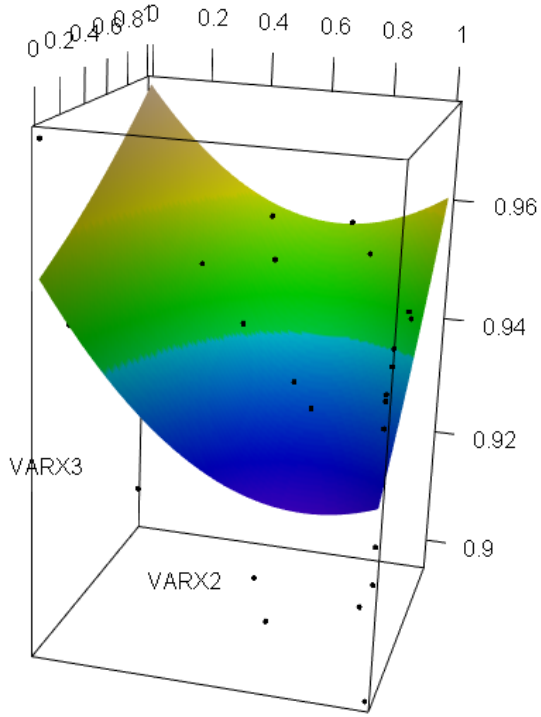
```
> spot(spotConfig=append(
+  list(report.func="spotReportContour",report.observations.all=T),
+  res),spotTask="rep")
```



```
> spot(spotConfig=append(
+  list(report.func="spotReport3d"),
+  res),spotTask="rep")
```







SEVEN

Factor Variables

- Categorical Variables
  - Enable Complexity Criterion
  - Enable Age Criterion
- Options
  - Special treatment: Mapping to $\mathbb{R}$ not adequate
  - Analyze each factor setting separately: Many extra runs
  - Random forests, MARS, ...: understanding, interpretability
  - Use linear model with dummy variables

SPOT SEVEN

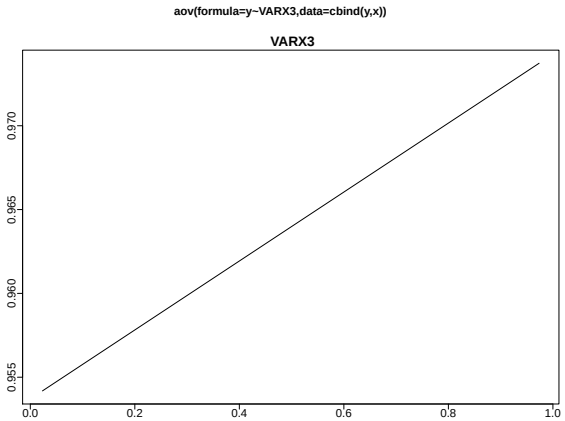
GMOGP Parameters, Linear Models with Factors

	Variable (Symbol)	*Domain*	*Default*
Population Size	`mu` (μ)	$\mathbb{N}$	300
Children per Generation	`lambda` (λ)	$\mathbb{N}$	20
Recombination Probability	`recombinationProbability` (p_{rec})	$[0, 1]$	0.1
Enable Complexity Criterion	`complexityCriterion`	$\mathbb{B}$	`true`
Enable Age Criterion	`ageCriterion`	$\mathbb{B}$	`true`
New Individuals per Generation	`nu` (ν)	$\mathbb{N}_0$	1

- Initial design: 20 design points $\times$ 2 repeats $=$ 40 GP runs
- 6 variables, 2 factors
- Linear model shows VARX3 (crossover prob.) has significant effect
- Refinement of the automated analysis:
  - Perform stepwise model selection by AIC

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
VARX3	1.000000	0.001389	0.001389	13.577399	0.000711
Residuals	38.000000	0.003887	0.000102		

Linear Models: Screening, Model Selection

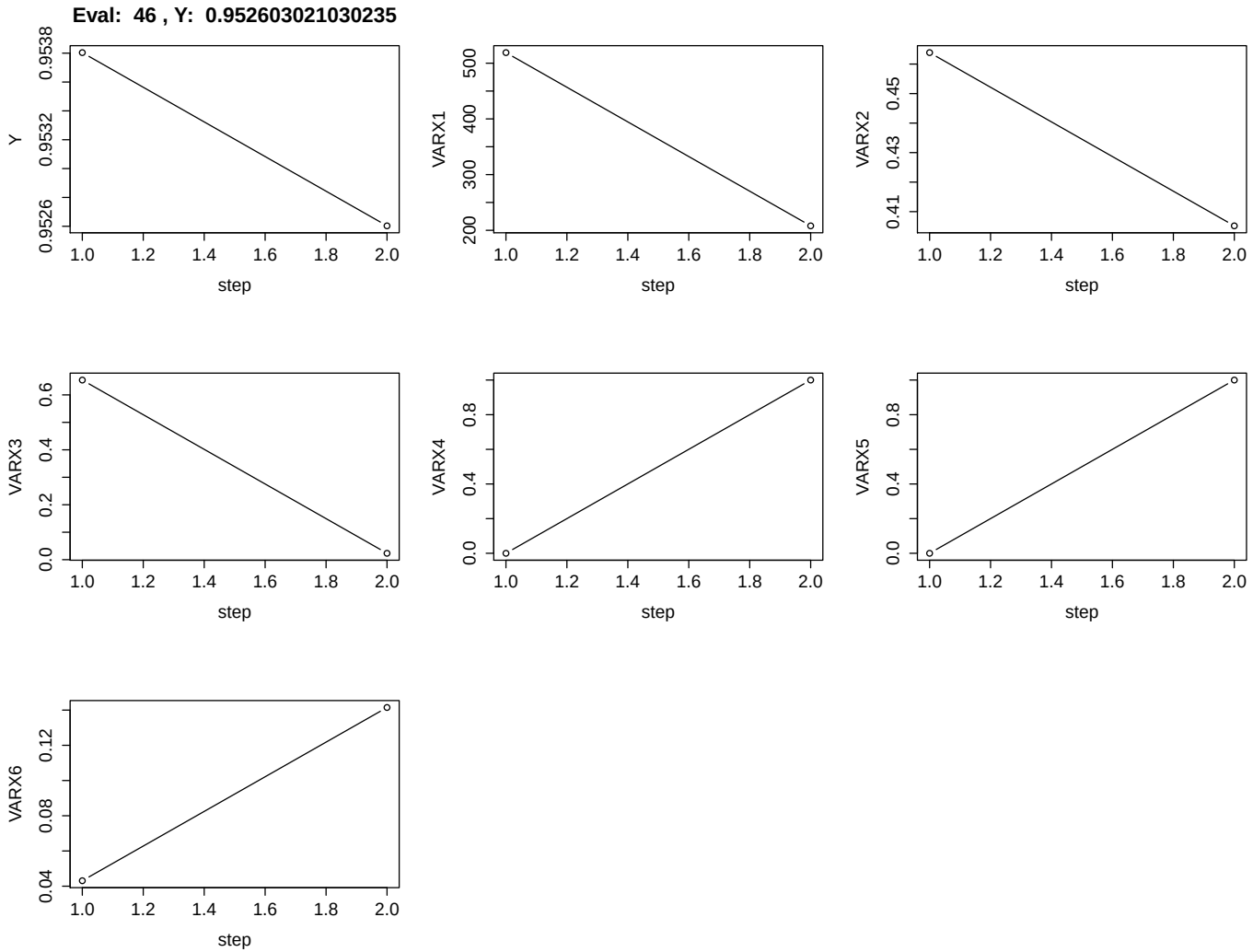


- Starting 6 additional GP runs to refine the model: 3 old models, 1 new with three repeats

VARX1	VARX2	VARX3	VARX4	VARX5	VARX6	CONFIG	REPEATS	STEP	SEED
519	0.46	0.65	0	0	0.04	2	1	1	3
208	0.40	0.02	1	1	0.14	9	1	1	3
108	0.26	0.49	0	0	0.96	4	1	1	3
75	0.24	0.00	0	0	0.37	21	3	1	1

SPOT SEVEN

Linear Models: Regression and Dummy Variables



- SPOT search path at step 2

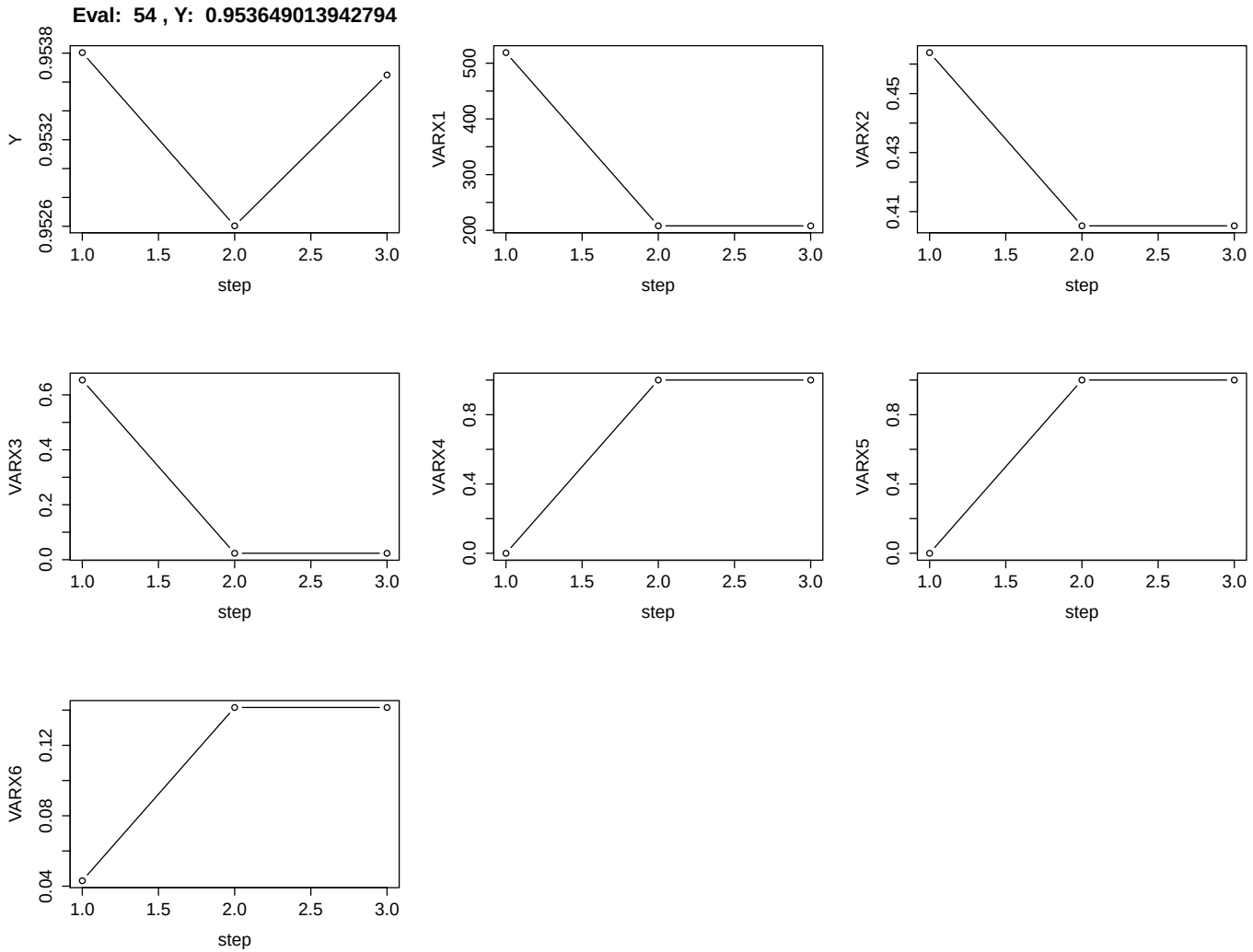
Linear Models: Regression Analysis

- Second step shows similar results
- Suggested design points after step 2 of the SPO:

VARX1	VARX2	VARX3	VARX4	VARX5	VARX6	CONFIG	REPEATS	STEP	SEED
208	0.40	0.02	1	1	0.14	9	1	2	4
108	0.26	0.49	0	0	0.96	4	1	2	4
555	0.54	0.50	1	0	0.91	15	2	2	3
464	0.46	0.00	0	1	0.29	22	4	2	1

SPOT SEVEN

Linear Models: Regression and Dummy Variables



- SPOT search path at step 3

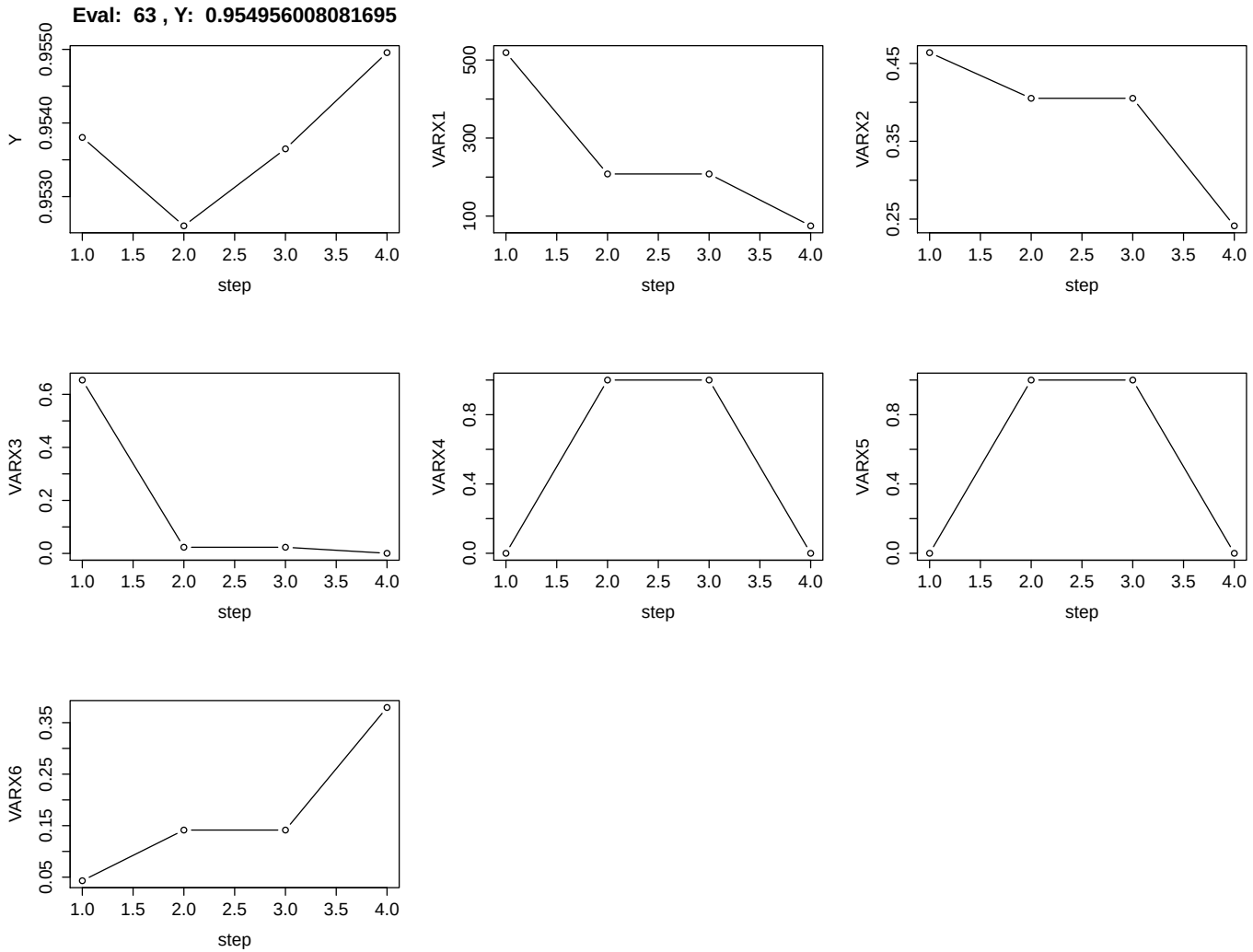
Linear Models: Regression Analysis

- Second step shows similar results
- Additional configurations proposed after step 3 of the SPO:

VARX1	VARX2	VARX3	VARX4	VARX5	VARX6	CONFIG	REPEATS	STEP	SEED
208	0.40	0.02	1	1	0.14	9	1	3	5
108	0.26	0.49	0	0	0.96	4	1	3	5
75	0.24	0.00	0	0	0.37	21	2	3	4
352	0.60	0.00	1	1	0.99	23	5	3	1

SPOT SEVEN

Linear Models: Regression and Dummy Variables



- SPOT search path at step 4

Linear Models with Factors after Step 4 of SPO

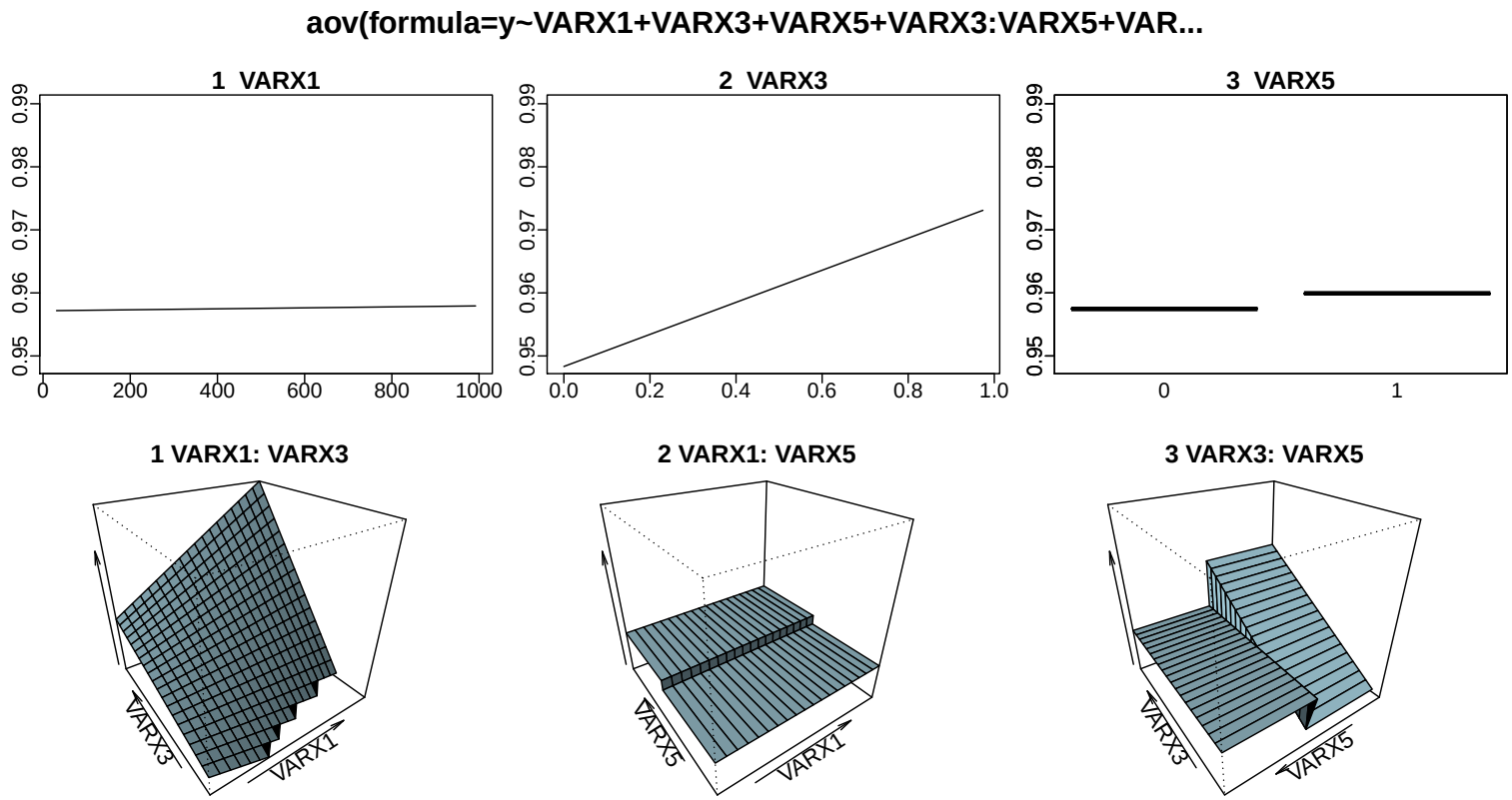
	Variable (Symbol)	*Domain*	*Default*
Population Size	`mu` (μ)	$\mathbb{N}$	300
Children per Generation	`lambda` (λ)	$\mathbb{N}$	20
Recombination Probability	`recombinationProbability` ($p_{\mathtt{rec}}$)	$[0, 1]$	0.1
Enable Complexity Criterion	`complexityCriterion`	$\mathbb{B}$	`true`
Enable Age Criterion	`ageCriterion`	$\mathbb{B}$	`true`
New Individuals per Generation	`nu` (ν)	$\mathbb{N}_0$	1

- Refinement of the automated analysis:
  - Perform stepwise model selection by AIC

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
VARX1	1.000000	0.000588	0.000588	9.403483	0.003309
VARX3	1.000000	0.001004	0.001004	16.056302	0.000180
VARX5	1.000000	0.000048	0.000048	0.765542	0.385272
VARX3:VARX5	1.000000	0.000290	0.000290	4.640992	0.035457
VARX1:VARX3	1.000000	0.000550	0.000550	8.790953	0.004414
Residuals	57.000000	0.003564	0.000063		

SPOT SEVEN

Linear Models: Regression and Dummy Variables



- VARX1 + VARX3 + VARX5 + VARX3:VARX5 + VARX1:VARX3
- Interactions between popsize (x1) and age (x5) and also between recombination prob. individuals (x3) and age (x5)

SPOT SEVEN

Summary: Regression Analysis

- Categorical parameters can be easily integrated into the SPOT tuning framework
- More complex models (MARS, GLM) can be used as well, however: Occam's razor
- Crossover prob. has significant impact, should be low
- Children per generation and age criterion: no effect
- Interactions
- Further steps: nested designs for complicated settings

SPOT SEVEN

Acknowledgments

- This work has been supported by the Federal Ministry of Education and Research (BMBF) under the grants FIWA (AIF FKZ 17N1009) and CIMO (FKZ 17002X11)

SEVEN