

Evolutionary Algorithms for Robust Methods

Robin Nunkesser

Department of Computer Science, TU Dortmund

CFE/ERCIM 2008

Outline

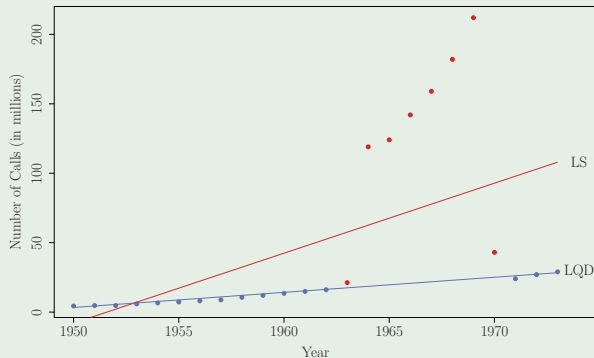
- 1 Introduction
 - Robust Regression
 - Evolutionary Computation
- 2 Robust Regression Algorithms
 - Evolutionary Algorithm
 - Compared Methods
- 3 Simulation Study
 - Data Model
 - Results

Motivation for Robust Regression

Definition (Donoho and Huber (1983))

The *(finite sample) breakdown point* is the smallest fraction of data points that need to be changed to have an unbounded effect on the estimate.

Number of international phone calls originated in Belgium



Drawback of Robust Regression

- Many robust methods are NP-hard to compute
- Fulfilling the exact-fit-property normally leads to NP-hardness

Theorem (Bernholt (2005))

The exact computation of LMS/LQS, LTS/LTA, MCD, MVE, all subset estimators, Constrained M estimation, Projection Depth, and Stahel-Donoho is NP-hard.

- Common assumption $P \neq NP \rightsquigarrow$ NP-hard estimators can not be computed exact for highly multivariate data
- Heuristics are used

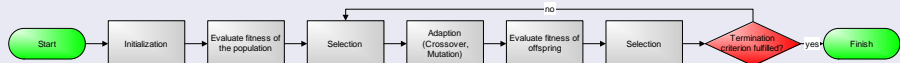
Evolutionary Computation (EC)

- General term for the usage of search heuristics inspired by natural evolution
 - possible solutions are represented by *individuals* (describing elements in a *search space*)
 - a set of individuals (a *population*) undergoes variation (*crossover* and *mutation*)
 - the *fitness* of the individuals is evaluated
 - a new generation is derived after a *selection* process
- Black-box optimization
- Often used for problems that are not easy to solve by conventional methods
 - Combinatorial optimization
 - Learning
 - Genetic association studies, robust regression, evolutionary clustering, time series modeling. . .

Layout of Algorithms for Evolutionary Computation

- 1 Create an initial random population.
- 2 Evaluate the fitness values of the population.
- 3 Perform the following steps on the current generation:
 - 1 Select individuals in the population based on a selection scheme.
 - 2 Adapt the selected individuals.
 - 3 Evaluate the fitness value of the adapted individuals.
 - 4 Select adapted individuals for the next generation according to a selection scheme.
- 4 If the termination criterion is fulfilled, then output the final population. Otherwise, set the next generation as current and go to step 3.

Graphical Representation



Evolutionary Algorithm for Robust Regression

- Subset of observations is suitable for many robust regression methods

Definition (Rousseeuw (1984), Croux et al. (1994))

The Least Median of Squares (LMS), Least Trimmed Squares (LTS), and Least Quartile Difference (LQD) estimator are defined by

$$\hat{\beta}_{\text{LMS}} = \arg \min_{\hat{\beta} \in \mathbf{R}^p} \text{med} r_i^2 ,$$

$$\hat{\beta}_{\text{LTS}} = \arg \min_{\hat{\beta} \in \mathbf{R}^p} \sum_{i=1}^h (r^2)_{i:n} ,$$

$$\hat{\beta}_{\text{LQD}} = \arg \min_{\hat{\beta} \in \mathbf{R}^p} \{ |r_i - r_j| \mid i < j \}_{(h/2):(n/2)} ,$$

i.e. they minimize the median, the sum of the h smallest, and a quartile of the absolute differences of the (squared) residuals.

Evolutionary Algorithm

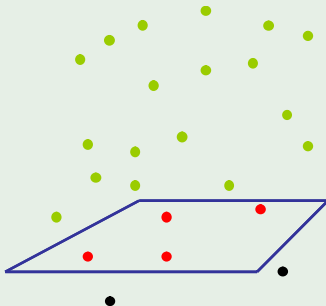
- ① Randomly select p observations.
- ② Perform the following steps on the current individual:
 - ① Conduct one of the following adaptations uniformly at random:
 - ① Exchange one of the selected observations with one of the not selected.
 - ② Conduct a problem specific adaptation.
 - ② Evaluate the fitness value of the (OLS adjusted) hyperplane defined by the selected observations (in general position), i.e.
 - ① $\text{med} r_i^2$,
 - ② $\sum_{i=1}^h (r^2)_{i:n}$,
 - ③ $\{|r_i - r_j| \mid i < j\}_{\binom{h}{2}: \binom{n}{2}}$,
 - ④ ...
 - ③ Proceed with the adapted individual, if it exceeds the original individual.
- ③ If the termination criterion is fulfilled, then output the final individual. Otherwise go to step 2.

Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.

Example

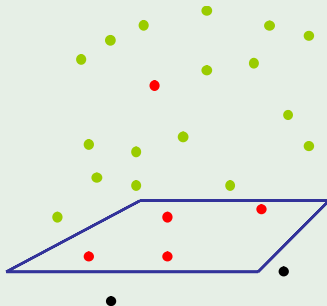


Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.

Example

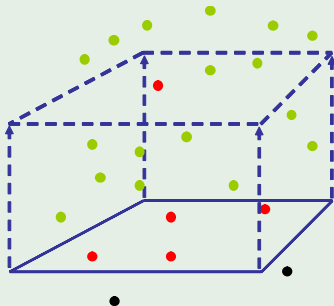


Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.
- 2 Move the hyperplane build by the p selected observations to the new observation.

Example

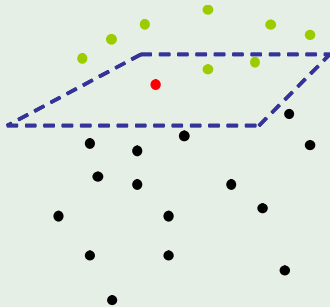


Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.
- 2 Move the hyperplane build by the p selected observations to the new observation.

Example

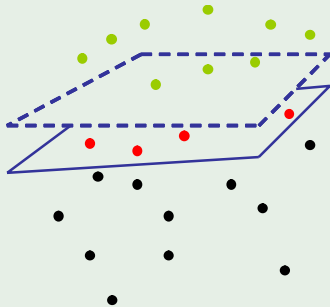


Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.
- 2 Move the hyperplane build by the p selected observations to the new observation.
- 3 Choose the p observations that are nearest to the new hyperplane.

Example

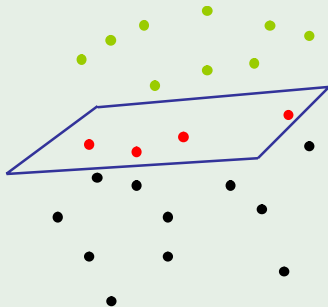


Specific Adaption

Needed to escape from local optima:

- 1 Choose one of the currently not selected observations.
- 2 Move the hyperplane build by the p selected observations to the new observation.
- 3 Choose the p observations that are nearest to the new hyperplane.

Example



Implementation

R Package RFreak

<http://cran.r-project.org/web/packages/RFreak/>

```
> LTSevol(stackloss[,4],stackloss[,1:3],adjust=TRUE)
```

Result obtained from FrEAK:

	Run	Generation	Objective value	Individual
1	1	983	-2.932391	000000100010000010100

Chosen subset:

```
[1] 7 17 6 11 19 5 12 9 18 10 8 15 16
```

Coefficients:

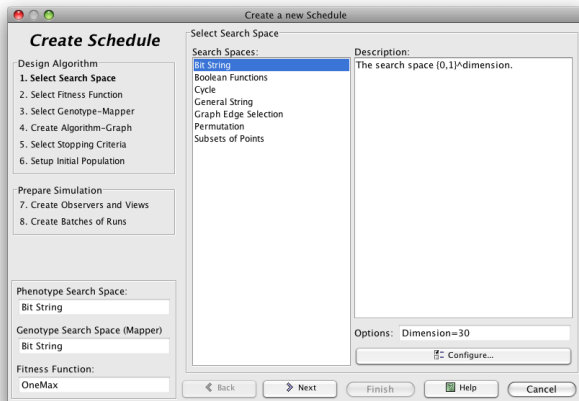
```
[1] -37.32332647 0.74092106 0.39152672 0.01113454
```

Criterion:

```
[1] 2.932391
```

Some Notes on RFreak

- Evolutionary Computation framework for R
- Modular layout (compare De Jong (2006)) for high reusability of code
- Growing number of application examples
- Easy to extend to further applications



Compared Methods (LMS and LQD)

LMS

- `lqs()` from the R package MASS
- Implementation of PROGRESS (Rousseeuw and Hubert, 1997)
- Based on hyperplanes defined by p observations
- Independent starts
- Intercept adjustment by computing univariate LMS

LQD

- No R implementation for highly multivariate data known to us
- May be computed with `lqs()` (quadratic blow up)
- Only LMS results are presented here

Compared Methods (LTS)

LTS

- `ltsReg()` from the R package `robustbase`
- Implementation of FAST-LTS (Rousseeuw and Van Driessen, 2006)
- Based on iteratively OLS adjusted hyperplanes defined by p observations
- Independent starts
- Intercept adjustment by computing univariate LTS

Data Model

General model

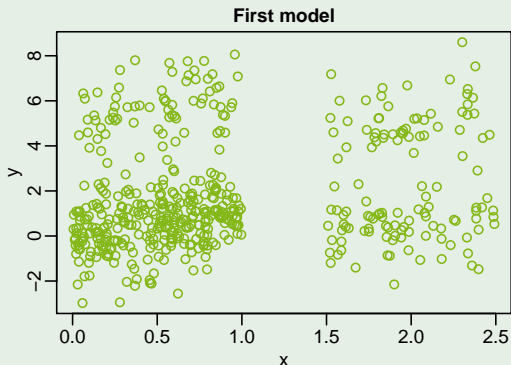
$$y_i = \beta_0 + \beta_1 x_{i,1} + \dots + \beta_{p-1} x_{i,p-1} + e_i \quad i = 1, \dots, 500$$

- We consider $1 \leq p - 1 \leq 30$
- Independent regressors $x_{\bullet 1}, \dots, x_{\bullet p-1}$ stem from a uniformly distributed random design on $(0, 1)$
- Outliers in x - and y -direction are constructed by adding a fixed value

First Model

- 25% outliers in x -direction (value 1.5)
- 25% outliers in y -direction (value 5)
- 10% outliers in x - and y -direction

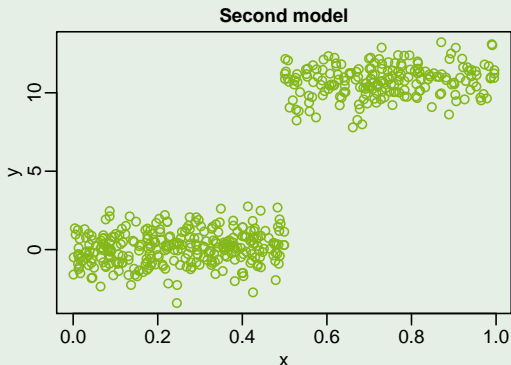
Example with one regressor



Second Model

- Simulate structural change
- $\beta_1 := 1$ and for $i > 1 : \beta_i := 0$
- $\beta_0 := 0$ for the first 300 data points, else $\beta_0 := 10$

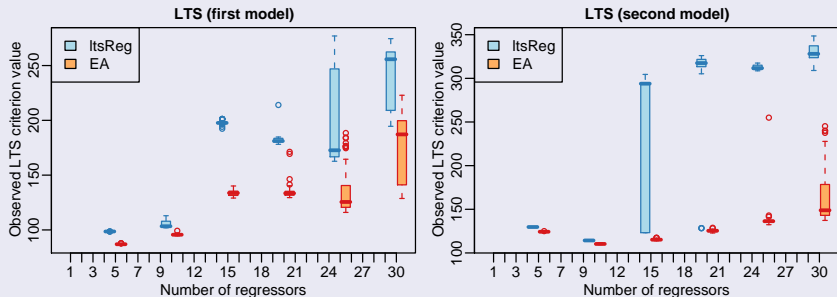
Example with one regressor



Results

- Simulations based on methods from design of experiments showed
 - better objective value
 - more computation time

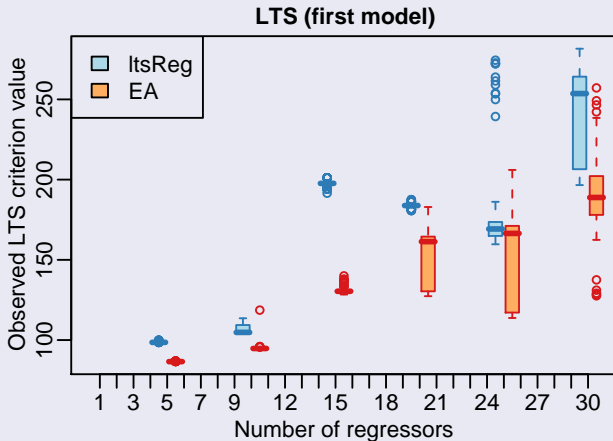
Objective value for an increasing number of regressors



- Here: results based on exactly the same computation time

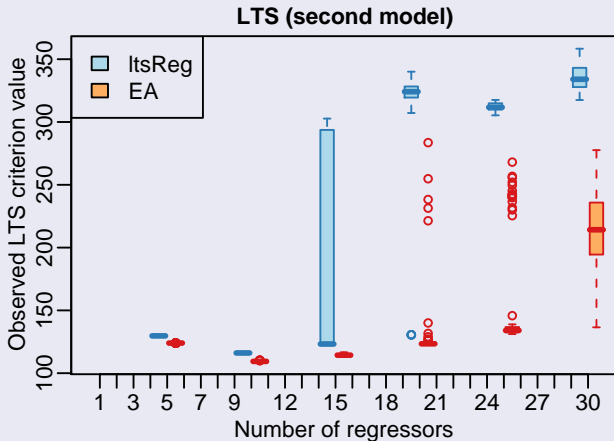
Results for LTS on the First Model

Objective value for an increasing number of regressors



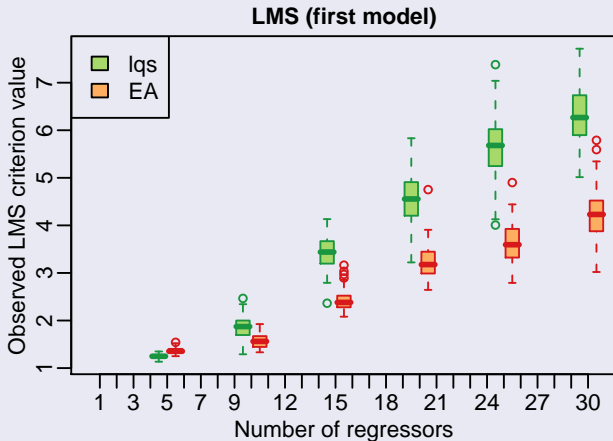
Results for LTS on the Second Model

Objective value for an increasing number of regressors



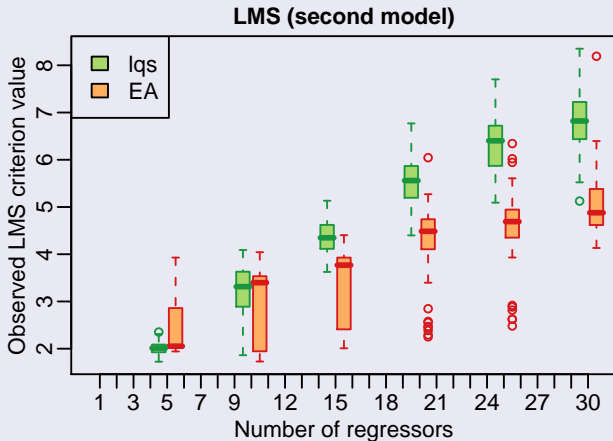
Results for LMS on the First Model

Objective value for an increasing number of regressors



Results for LMS on the Second Model






Objective value for an increasing number of regressors



Summary

- Evolutionary Algorithm suited for different robust methods
- Superior results on LTS and LMS in demanding data situations
- Same runtime
- Easy to extend to further methods

Bibliography

-  Bernholt, T., 2005. Robust estimators are hard to compute. Tech. Rep. 52/2005, SFB 475, Universität Dortmund.
-  Morell, O., Bernholt, T., Fried, R., Kunert, J., Nunkesser, R., 2008. An evolutionary algorithm for lts-regression: A comparative study. In: Proceedings of Compstat 2008. Accepted.
-  Nunkesser, R., 2008. Rfreak—an r package for evolutionary computation. Tech. rep., SFB 475, Technische Universität Dortmund.
-  Rousseeuw, P., Hubert, M., 1997. Recent developments in PROGRESS. In: Dodge, Y. (Ed.), L_1 -Statistical Procedures and Related Topics. Vol. 31 of Lecture Notes-Monograph Series. Institute of Mathematical Statistics, pp. 201–214.
-  Rousseeuw, P. J., Van Driessen, K., 2006. Computing lts regression for large data sets. Data Min. Knowl. Discov. 12 (1), 29–45.