Comput Manag Sci (2011) 8:103–123
DOI 10.1007/s10287-009-0103-x

ORIGINAL PAPER

Least median of squares estimation by optimization heuristics with an application to the CAPM and a multi-factor model

Peter Winker · Marianna Lyra · Chris Sharpe

Received: 4 August 2008 / Accepted: 11 March 2009 / Published online: 19 September 2009

Abstract For estimating the parameters of models for financial market data, the use of robust techniques is of particular interest. Conditional forecasts, based on the capital asset pricing model, and a factor model are considered. It is proposed to consider least median of squares estimators as one possible alternative to ordinary least squares. Given the complexity of the objective function for the least median of squares estimator, the estimates are obtained by means of optimization heuristics. The performance of two heuristics is compared, namely differential evolution and threshold accepting. It is shown that these methods are well suited to obtain least median of squares estimators for real world problems. Furthermore, it is analyzed to what extent parameter estimates and conditional forecasts differ between the two estimators. The empirical analysis considers daily and monthly data on some stocks from the Dow Jones Industrial Average Index.

Keywords Least median of squares · CAPM · Multi-factor model · Differential evolution · Threshold accepting

We are indebted to two referees for their helpful and constructive comments which helped to improve the paper. Financial support from the EU Commission through MRTN-CT-2006-034270 COMISEF is gratefully acknowledged.

P. Winker (✉) · M. Lyra · C. Sharpe
Department of Economics, University of Giessen, Licher Str. 64, 35394 Giessen, Germany
e-mail: Peter.Winker@wirtschaft.uni-giessen.de

M. Lyra
e-mail: Marianna.Lyra@wirtschaft.uni-giessen.de

C. Sharpe
e-mail: Chris.Sharpe@wirtschaft.uni-giessen.de

1 Introduction

The estimation and analysis of the capital asset pricing model (CAPM) and other models with more than one factor is often complicated by the fact that the distribution of the error terms cannot be assumed to be independently identically normal. Consequently, different robust estimation approaches have been considered (see, e.g., Chan and Lakonishok 1992; Knez and Ready 1997; Martin and Simin 2003; Ronchetti and Genton 2008).

In this contribution, we consider the classical least median of squares (LMS) estimator (Rousseeuw 1984). Although this estimator exhibits nice properties with regard to robustness, it is not used frequently. One possible reason is that estimation requires to solve a complex optimization problem. In particular, the objective function landscape is not smooth and exhibits many local optima. Consequently, traditional optimization methods will fail. One alternative consists in exploiting the inherent discrete nature of the optimization problem and resorting to a full enumeration of all potential solutions. An algorithm built on this approach is PROGRESS proposed by Rousseeuw and Leroy (1987).[1] However, the complexity of this approach grows at a rate of T^2 in the sample size T for the bivariate regression. If more than one factor has to be considered, the complexity grows dramatically. Furthermore, the technique does not allow for a simple implementation of nonlinear or constraint estimation. Optimization heuristics have the potential to overcome these shortcomings.

Heuristic optimization techniques have been successfully applied to a variety of problems in statistics and economics for well over a decade (see Gilli et al. (2008) and Gilli and Winker (2009) for recent overviews). However, applications to estimation problems are still rare. Fitzenberger and Winker (2007) consider Threshold Accepting (TA) for censored quantile regression, a problem similar to the LMS estimator.[2] Maringer and Meyer (2008) and Yang et al. (2007) also use TA for model selection and estimation of smooth transition autoregressive models. In contrast, several optimization heuristics have been used in other fields of research in finance, e.g., portfolio optimization (Dueck and Winker 1992; Maringer 2005; Winker and Maringer 2007a; Specht and Winker 2008) and credit risk bucketing (Krink et al. 2007).

We present an application to the LMS estimator. In particular, we propose implementations of TA and Differential Evolution (DE) for obtaining the LMS estimator. We purposely select a population based search method, DE, and a local search method, TA, to compare their efficacy on a continuous search space.[3] We provide some evidence on the tuning of both algorithms and the relative performance for this problem. It turns out that the LMS estimator can be obtained quite reliably using optimization heuristics despite of its high inherent complexity.

[1] Barreto and Maharry (2006) propose a generalization for the bivariate regression without a constant. The approach might be considered as an application of elemental subset regression (Mayo and Gray 1997).

[2] In fact, Fitzenberger and Winker (2007) exploit the elemental subset properties of quantile regression for their approach.

[3] Note, that we do not take into account the implicit discrete structure of the optimization problem related to elemental subset regression. This might reduce the performance of TA, but allows to introduce constraints and nonlinear components in future research.

Finally, we apply the estimator to the CAPM and a three factor model for a large set of rolling window samples for some of the stocks comprising the Dow Jones Industrial Average Index (DJIA). The estimates differ substantially for some stocks and time periods from those obtained by ordinary least squares (OLS). We also calculate the conditional forecasts based on the model and the actual factor values. These conditional forecasts are compared with those obtained from the OLS estimates. It is also analyzed to what extent a combination of both forecasts might reduce the forecasting errors.

The rest of the study is organized as follows. Section 2 shortly reviews the theoretical background to the underlying models of the financial market and the LMS estimator. Section 3 reports on heuristic strategies, describes the optimization problem and the algorithms used. The specific application and the empirical results are presented in Sect. 4. In Sect. 5, we provide evidence on the rate of convergence of the two heuristics, while Sect. 6 summarizes the main findings and provides an outlook to further research.

2 Theoretical background

2.1 CAPM and multi-factor models

The CAPM provides a method for estimating the risk-return equilibrium. The pioneering work by Markowitz (1952) has set the foundation of modern portfolio management and was employed later by Sharpe (1964), Lintner (1965) and Mossin (1966) to develop the CAPM. The CAPM describes a linear relationship between the risk premium on individual securities relative to the risk premium on the market portfolio. It is modeled by

$$r_{i,t} - r_t^s = \alpha + \beta \left(r_{m,t} - r_t^s \right) + \varepsilon_{i,t} \, , \tag{1}$$

where $r_{i,t}$ is the rate of return at time t for asset i; r_t^s the risk-free rate of return at time t; α, β the parameters of CAPM; $r_{m,t}$ the market rate of return at time t and $\varepsilon_{i,t}$ is the residual at time t for asset i.

The simplicity of the CAPM, i.e., the concentration on a single risk factor, is one of the reasons why the explanatory power of the model is limited. One extension to the model has been proposed by Fama and French (1992). The authors emphasize the multi-dimensionality of risks. In particular, they consider the effects of firm size and book value to equity in explaining the cross-section of average stock returns. In another paper, Fama and French (1993) introduce the three-factor model. They conclude that the market factor together with a size and a book-to-market factor can explain 95% of the variation in excess stock returns. A key finding is that the difference between small and big firms and the difference between high and low book-to-equity value captures variation through time.[4]

[4] For a critical assessment of the empirical performance of the model, see, e.g., Knez and Ready (1997). Other non parametric pricing models have also been proposed in the literature, e.g., Ince (2006) introduces technical indicators composed of stock price and volume over time.

The final form of the model used in our application is given by Eq. (2). The market risk $r_{m,t} - r_t^s$ is as for the CAPM given by the difference between the market rate of return and the risk free rate. *SMB* is defined as the average return on three portfolios comprising small firm stocks minus the average return on three portfolios comprising larger firms. Finally, *HML* is the average return on two portfolios comprising firms with high book-to-market value minus the average return on the two so called growth portfolios with low book-to-market values.

$$r_{i,t} - r_t^s = \alpha + \beta_1 \left(r_{m,t} - r_t^s \right) + \beta_2 SMB_t + \beta_3 HML_t + \varepsilon_{i,t}, \tag{2}$$

where $r_{m,t} - r_t^s$ is the factor accounting for market risk premium at time (t); SMB_t the factor accounting for size premium at time t; HML_t the factor accounting for value premium at time t and β_1, β_2, and β_3 are the exposure levels to the corresponding risk factors.

2.2 LMS

A substantial amount of research in financial market economics has focused on estimating the parameters of CAPM and multi-factor models.[5] OLS estimation can be problematic due to its lack of robustness (Rousseeuw and Wagner 1994; Ronchetti and Genton 2008). In particular, outliers can have a strong effect on the estimated coefficients.[6] The smallest percentage of influential observations that can change the parameters of the regression line is called breakdown point (Rousseeuw 1984).

In order to achieve a higher breakdown point, a number of robust techniques have been suggested in the statistical literature (see, e.g., Maronna et al. (2006) for an introduction and overview). These include least absolute deviations (LAD), also called minimum absolute deviations (MAD) suggested by Sharpe (1971); Cornell and Dietrich (1978), and later by Chan and Lakonishok (1992). The former group of authors applied also trimmed regression quartile (LTQ) estimators. Huber (1973) introduced M-estimators applied by Martin and Simin (2003) in the context of the CAPM.

In the present application, we concentrate on the LMS estimator proposed by Rousseeuw (1984) as it is one of the earliest highly robust methods that has also been applied in econometrics (Zaman et al. 2001). More details on the approach can be found in Rousseeuw and Leroy (1987) and Rousseeuw and Wagner (1994). Other robust estimates, such as M-estimators, might exhibit higher efficiency. Therefore, it should be noted that the implementation presented here does not aim to demonstrate the superiority of LMS estimates, e.g., with regard to predictive performance,[7] but to provide a proof of concept, i.e., that LMS estimates obtained by means of heuristic

[5] We will not comment on the difficulties related to the definition of the variables, in particular the risk free rate of return and—even more difficult—the market rate of return which should summarize all available investment opportunities.

[6] In the application to financial market data, the meaning of "outliers" is not obvious. In fact, they might provide relevant information and should not be discarded from the analysis (Knez and Ready 1997).

[7] For a discussion of the predictive performance of CAPM and multi-factor models based on OLS estimates see Simin (2008).

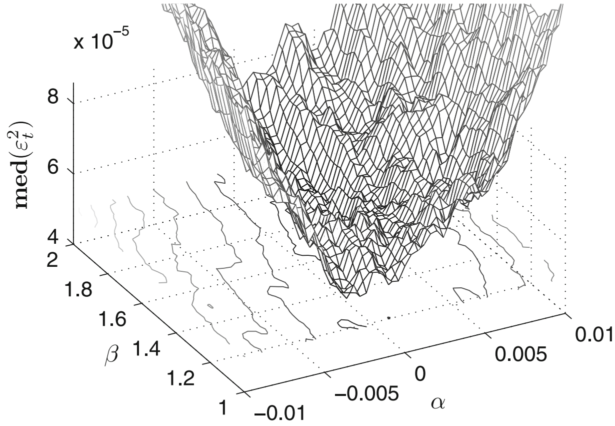

Fig. 1 Median of squared residuals as a function of α and β

optimization can be used for real life applications. This will allow for further extensions in the future, e.g., taking into account constraints or nonlinear relationships.[8]

The LMS estimator for the CAPM is defined as the solution to the following optimization problem:

$$\min_{\alpha,\beta} \left(\text{med}(\varepsilon_{i,t}^2) \right), \tag{3}$$

where $\varepsilon_{i,t} = (r_{i,t} - r_t^s) - \alpha - \beta(r_{m,t} - r_t^s)$ according to Eq. (1) above. This results in a highly complex objective function as illustrated for one problem instance in Fig. 1.

For the multi-factor model (2), the optimization problem becomes

$$\min_{\alpha,\beta_1,\beta_2,\beta_3} \left(\text{med}(\varepsilon_{i,t}^2) \right), \tag{4}$$

with $\varepsilon_{i,t} = (r_{i,t} - r_t^s) - \alpha - \beta_1(r_{m,t} - r_t^s) - \beta_2 SMB_t - \beta_3 HML_t$.

3 Heuristic strategies

Recent research[9] suggests that even an apparently simple optimization problem might result in an objective function which does not allow for the successful application of standard numerical approximation algorithms. In this vein, minimizing the median of squared residuals results in a search space containing many local minima, where traditional optimization methods can not provide an exact solution. As an example, Fig. 1 provides an illustration (over a finite grid of value pairs for α and β) of the objective function for the CAPM using the 200 daily stock returns of the IBM stock starting

[8] Roko and Gilli (2008) applied successfully another technique (classification trees) to model the non-linear relationship between expected stock returns and financial and economic factors for the S&P 500.

[9] E.g., Gilli and Winker (2007) and the papers in that special issue.

Algorithm 1 Threshold Accepting Algorithm

```
1: Initialise n_R, n_S, and τ_r, r = 1, 2, ..., n_R
2: Generate at random a solution x^0 ∈ [α_l α_u] × [β_l β_u]
3: for r = 1 to n_R do
4:   for i = 1 to n_S do
5:     Generate neighbor at random, x^1 ∈ N(x^0)
6:     if f(x^0) − f(x^1) < τ_r then
7:       x^0 = x^1
8:     end if
9:   end for
10: end for
```

on 2 January, 1970.[10] Note that the illustration might be misleading as it seems to suggest that the function values for the grid points are connected by a smooth surface. However, even between the grid points, further local minima might exist.

In principle, it is possible to provide an exact solution to this optimization problem by exploiting the inherent discrete structure of the problem. However, this comes at high computational cost which becomes a binding constraint when additional factors are considered. Furthermore, the technique based on elemental subset regression proposed by Rousseeuw and Leroy (1987) can not easily be generalized for nonlinear models or when some constraints are imposed on the parameter space.[11] Heuristic optimization methods are well suited to handle such problems. If traditional methods fail due to the existence of many local optima, the performance of optimization heuristics will typically dominate them in terms of solution quality. In the following, we will analyze the performance of two heuristic methods for the LMS estimation problem, TA and DE.

3.1 Threshold accepting

Originally devised by Dueck and Scheuer (1990), TA has proven to be a simple, powerful search tool for many types of optimization problems. A key advantage of TA is that it enables the search to escape local minima. Here we present a modified version of the standard TA algorithm for the LMS estimation problem.[12] Algorithm 1 provides the general outline. First, the number of rounds n_R and the number of steps per round n_S are initialized as well as the threshold sequence τ_r. Next, a random solution x^0 is chosen (2:), which corresponds to a vector of parameter values (α, β) for the CAPM. Then, in each round r, n_S local search steps are executed for a fixed value of the threshold τ_r, which determines (6:) to what extent not only local improvements, but also local impairments are accepted. In each search step i, a random neighbor x^1

[10] In passing note that the sample size used for the estimation in our application is less than one year. Thus, it is substantially lower than the standard practice of 5 years used in industry (Simin 2008, p. 358). However, repeating our analysis with larger samples does not affect the qualitative findings.

[11] One example is the constraints for the CAPM coefficients resulting from the international CAPM as discussed by Engel and Rodrigues (1993).

[12] A description of the general form of the algorithm and its behavior is given by Winker and Maringer (2007b). For a further, comprehensive overview see Winker (2001).

Algorithm 2 Data Driven Generation of Threshold Sequence

1: Initialize n_R, lower quantile α, $n_D = [n_R/\alpha]$
2: **for** $r = 1$ to n_D **do**
3: Generate at random a solution $x_r^c \in [\alpha_l \alpha_u] \times [\beta_l \beta_u]$
4: Generate at random a near neighbor solution $x_r^n \in \mathcal{N}(x_r^c)$
5: Calculate $\Delta_r = |f(x_r^n) - f(x_r^c)|$
6: **end for**
7: Sort $\Delta_1 \leqslant \Delta_2 \leqslant \cdots \leqslant \Delta_{n_D}$
8: Use $\Delta_{n_R}, \ldots, \Delta_1$ as threshold sequence

of the current solution x^0 is generated by minor random modifications of α and β. The objective function f returns the median of the squared residuals. A comparison of the two solutions determines whether the current solution is kept or is replaced by x^1 if the new solution is better or at least not worse by more than the current threshold τ_r (7:). The algorithm terminates after an a priori fixed number of $n_R \times n_S$ iterations.

Thereby, the threshold sequence τ_r determines the search behavior of the algorithm. Broadly speaking, the TA search behaves like a random search in the initial stages, for large values of τ, and gradually transforms into a greedy search, as $\tau \to 0$. The degree of randomness in the initial stages depends on the starting value of the threshold sequence; the degree of 'greediness' in the latter stages depends on how the threshold is reduced. Rather than guess the first and last values of the threshold sequence and refine the values by trial and error, Winker and Fang (1997) suggested a data driven approach (see also Winker and Maringer 2007b). This method is used in our implementation. The pseudocode for the data driven generation of the threshold sequence is provided in Algorithm 2.

Returning to the main TA algorithm, we select reasonable boundaries for the search space $[\alpha_l \alpha_u] \times [\beta_l \beta_u]$ and generate a starting point x^0 at random within this area (2:). Then, in each step i, a further random solution is generated within a neighborhood of the current solution, $N(x^0)$. There are several options for the shape of the neighborhood in a two dimensional search space. However, a hyper-rectangle offers the advantage of small computational overhead to recalculate its dimensions.[13] We set the initial dimensions of the hyper-rectangle to the boundaries of the search space and reduce the dimensions proportionally with the number of rounds. Whether we choose a linear or geometric reduction of the neighborhood dimensions does not appear to make much of a difference for the quality of the results. The number of reductions to the hyper-rectangle is n_R, equal to the number of values in the threshold sequence. The rationale behind reducing the neighborhood is closely connected to the behavior of the search and the threshold sequence. The first neighborhood allows for a general exploration of the full search space; at the end there is a limited, concentrated exploration of a small area, where we assume good quality solutions lie.

One issue that needs to be dealt with is reconstructing the neighborhood if it exceeds the bounded search space. A straight forward solution is to shift the whole neighborhood in a vertical, horizontal, or diagonal direction by the amount it has

[13] See Winker (2001) and Gilli et al. (2008) for a general discussion on neighborhoods in higher dimensional search spaces.

exceeded the bounded search space. It should be emphasized that this would be a less trivial, computationally more expensive operation with other neighborhood shapes.

3.2 Differential evolution

DE is a population based optimization technique for continuous objective functions developed by Storn and Price (1997). The algorithm starts with a randomly initialized set of candidate solutions $P_{j,i}^{(1)}$ each corresponding to a parameter vector (α, β) for the CAPM and $(\alpha, \beta_1, \beta_2, \beta_3)$ for the multi-factor model, respectively. Then, for a predefined number of generations, the elements of the population are updated by generating linear combinations of existing elements and random crossover. Finally, the objective function value of the new candidate solution is compared with that of the original element. If it is lower, the new candidate solution replaces the old one. Algorithm 3 provides the pseudocode of our implementation.

Algorithm 3 Differential Evolution Algorithm

1: Initialize parameters n_p, n_G, F *and* CR
2: Initialize population $P_{j,i}^{(1)}$, $j = 1, \ldots, \mathrm{d}, \mathrm{i} = 1, \ldots, n_p$
3: **for** $k = 1$ to n_G **do**
4: $\quad P^{(0)} = P^{(1)}$
5: $\quad$ **for** $i = 1$ to n_p **do**
6: $\quad\quad$ Generate $r_1, r_2, r_3 \in 1, \ldots, n_p, r_1 \neq r_2 \neq r_3 \neq i$
7: $\quad\quad$ Compute $P_{.,i}^{(\upsilon)} = P_{.,r_1}^{(0)} + F \times (P_{.,r_2}^{(0)} - P_{.,r_3}^{(0)})$
8: $\quad\quad$ **for** $i = 1$ to d **do**
9: $\quad\quad\quad$ Generate $u \sim U(0, 1)$
10: $\quad\quad\quad$ **if** $u < CR$ **then**
11: $\quad\quad\quad\quad P_{j,i}^{(u)} = P_{j,i}^{(\upsilon)}$
12: $\quad\quad\quad$ **else**
13: $\quad\quad\quad\quad P_{j,i}^{(u)} = P_{j,i}^{(0)}$
14: $\quad\quad\quad$ **end if**
15: $\quad\quad$ **end for**
16: $\quad\quad$ **if** $f(P_{.,i}^{(u)}) < f(P_{.,i}^{(0)})$ **then**
17: $\quad\quad\quad P_{.,i}^{(1)} = P_{.,i}^{(u)}$
18: $\quad\quad$ **else**
19: $\quad\quad\quad P_{.,i}^{(1)} = P_{.,i}^{(0)}$
20: $\quad\quad$ **end if**
21: $\quad$ **end for**
22: **end for**

As mentioned above, the initial population of n_p elements is randomly chosen (2:). Note that the performance of proper implementations should not depend on the specific choice of starting values in contrast to standard gradient methods. However, they introduce a stochastic component which can be analyzed when running the algorithm repeatedly on the same problem instance (see Sect. 5). Therefore, the values for the initial population are generated as random realizations of uniform random numbers on the intervals $[-0.1, 0.1]$ for α, $[-1, 2]$ for β (β_1), $[0, 1]$ for β_2, and $[0, 1]$ for β_3.

Returning to the DE algorithm, for a predefined number of generations n_G, the algorithm performs the following procedure: Each element of the population is updated by means of differential mutation (6:-7:) and crossover (8:-15:). The superscript υ indicates the mutated vector for every element of the population. Particularly, differential mutation constructs new parameter vectors by adding the scaled difference of two randomly selected vectors to a third one (7:). F is the scale factor that determines the speed of shrinkage in exploring the search space. During crossover, DE generates for each component a uniform random number u (9:). This is compared with the crossover probability CR (10:) and determines which initial elements $P^{(0)}_{j,i}$ will be replaced with mutant ones $P^{(\upsilon)}_{j,i}$ resulting in a new trial vector $P^{(u)}_{j,i}$. Finally, the value of the objective function of the trial vector is compared with that of the initial element (16:). Only if the trial vector results in a better value of the objective function, it replaces the initial element in the population (17:). The above process repeats until all elements of the population have been considered. Then, the process restarts for the next generation.

Calibration issues. Price et al. (2005) report that, although the scale factor F has no upper limit and the crossover parameter CR is a fine tuning element, both are problem specific. In an attempt to improve the tuning of the algorithm, we conducted repeated runs for different values of the population size n_p and the number of generations n_G. During this initial phase we did not tune the weighting (scaling) factor (F) and the crossover probability (CR). In order to achieve convergence, we increased the population size n_p to more than ten times the number of parameters.[14] We observe that when the best value is found repeatedly for several runs of the algorithm, a further increase in the number of generations (to more than 100) does not improve the results, while the computational time increases. With a population size of $n_p = 20$, which is ten times the number of parameters for the CAPM (2), a number of generations of $n_G = 100$, and the constants set to $F = 0.8$ and $CR = 0.9$, the algorithm typically converges to the same results in several replications. By increasing the population size to $n_p = 50$, the algorithm consistently provides identical outcomes in each repetition.

For fine tuning the technical parameters, the algorithm has been run for different combinations of F and CR. The procedure is illustrated in Algorithm 4 for parameter values ranging from 0.5 to 0.9.

Algorithm 4 Calibration of Crossover and Scaling Factor Parameters

```
Initialize parameters n_p, n_G
Initialize population P^(1)_{j,i}, j = 1, ..., d, i = 1, ..., n_p
for F = 0.5, ..., 0.9 do
  for CR = 0.5, ..., 0.9 do
    Repeat Algorithm 3 from line 3-22
  end for
end for
```

Figure 2 exhibits the dependence of the best value of the objective function obtained for different combinations of F and CR always for the same problem instance

[14] A practical advice for optimizing objective functions with DE is given on www.icsi.berkeley.edu/~storn/.

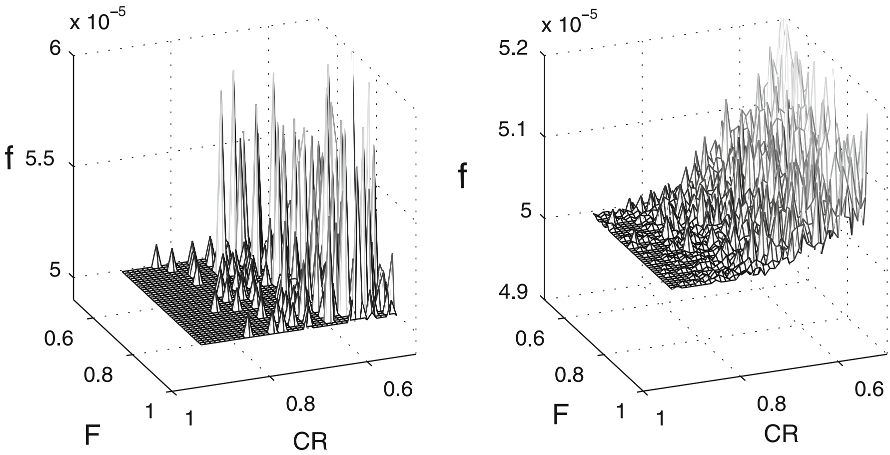

Fig. 2 Calibration of technical parameters

(first 200 observations of the IBM stock in our sample). The population size n_p and the number of generations n_G are set to 50 and 100, respectively. The left side of Fig. 2 presents the results for a single run of the algorithm, while the right side shows the mean over 30 restarts. Although the surface is full of local minima for CR below 0.7, it becomes smoother as CR reaches 0.8 independent of the choice of F. The results clearly indicate that for higher values of CR, results improve, while the dependency on F appears to be less pronounced. Based on these results, we use $F = 0.8$ and $CR = 0.9$ for estimating the parameters of the models in the next section.

4 Empirical findings

4.1 Implementation details

For the application of LMS to the CAPM, we consider daily data from the sample of publicly traded firms comprising the DJIA for the period between 1970 and 2006. We select six companies, IBM, ExxonMobil (XOM), General Electric (GE), Merck (MRK), General Motors (GM) and Boeing (BA).

In order to estimate the parameters of the CAPM over a sensible time period, we use a rolling window of 200 days length moving from 1970 to 2006 day by day. For each given sample, the parameters α and β are obtained by LMS estimation using ten restarts of each heuristic.[15] The estimates corresponding to the best value of the objective function are kept.

Next, for given parameter estimates, we forecast the excess return conditional on the actual market return for the next trading day. In Eq. (5), $\hat{r}^a_{m,t+1}$ denotes the actual

[15] We only report results for the DE implementation as it appears to be more efficient for the specific problem as discussed in Sect. 5 below. For the DE implementation, we use $n_P = 50$ and $n_G = 100$. The computation time for 10 restarts on a given sample amounts to about 1.7 seconds using Matlab 7.4 on a PC with Intel Duo Core processor operating at 2.39 GHz, running Windows XP OS.

market return for the next trading day. Then, the conditional forecasts of the excess returns for stock i is defined as

$$\hat{r}^{f}_{i,t+1} = \hat{\alpha} + \hat{\beta}\hat{r}^{a}_{m,t+1} + \nu_{t+1}\,, \tag{5}$$

The same calculation is done based on the OLS estimates. Finally, both forecast errors are compared. Algorithm 5 summarizes the procedure.

Algorithm 5 Rolling window estimation

```
Initialize parameters
for t = 1 to 9113 do
    Run optimization heuristic 10 times for sample [t ... t + 199]
    LMS estimates α_t^LMS and β_t^LMS correspond to best value of objective function
    OLS estimates α_t^OLS and β_t^OLS
    Calculate one-day-ahead conditional forecasts
end for
```

For the Fama/French multi-factor model we use monthly data for the period between 1962 and 2008, except for XOM and MRK stocks for which the sample starts only in 1970.[16] The length of the rolling window is fixed to 10 months. Otherwise, the procedure is identical to that used for the CAPM.

4.2 Estimation results

The results of the rolling windows estimation for the CAPM are illustrated in Figs. 3 and 4, for IBM and XOM, respectively. In both figures the actual stock returns are presented in the top graph. The β estimates using OLS and LMS are presented in the middle and the bottom graphs, respectively.

For a few samples we compared our results for the CAPM with those obtained from the R implementation of LMS in the MASS-package which should allow to derive exact solutions by full enumeration based on subset elemental regression.[17] Typically, the estimates obtained by DE are almost identical to those results. However, for a few cases, we found slightly better values for the estimates from DE which might result from a different definition of the median in the MASS-package.

When looking at the results, the repeated zero values for XOM at the beginning of the sample appear surprising. This is a result of the LMS approach given that the dataset contains a large number of zero returns for this period, i.e., no change in the stock price from day to day. However, apart from these unusual periods, the LMS estimators appear not to be smoother than the OLS estimators. This might indicate that the kind of extreme events that make the use of robust estimators preferable are

[16] The data for the factors is taken from Kenneth R. French's website http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html.

[17] It should be noted that this approach becomes computationally infeasible as soon as the sample size becomes large and/or additional regressors are taken into account in a multi-factor model. Furthermore, it does not allow for imposing constraints on the parameter space in a straightforward way.

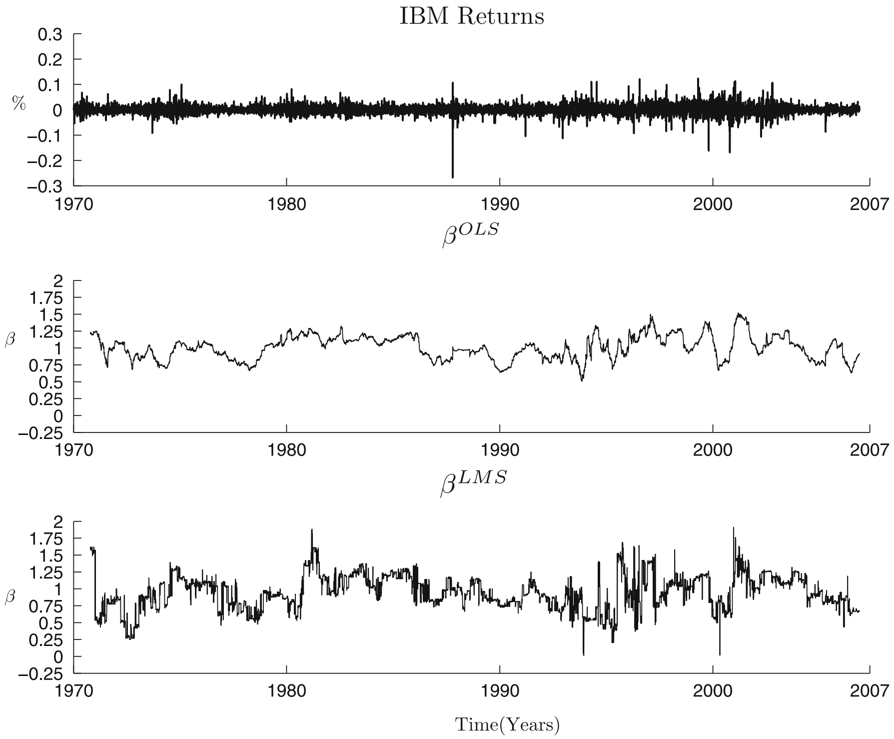

Fig. 3 Actual returns and estimates of β for IBM (1971–2006)

rather rare in our sample. Similar results are found for the other stocks in our analysis.

Figure 5 provides scatter plots of the β estimates obtained by OLS and LMS, respectively, for the samples including data from 1980 to 2006. Thereby, we exclude the unusual results for the early 1970s. Obviously, both estimates exhibit a marked positive correlation, in particular when excluding the zero estimates for XOM. Nevertheless, it also becomes obvious that the correlation is far from being perfect. Therefore, a comparison of the relative forecasting performance of both approaches will be provided in Sect. 4.3.

Of course, these results should not be interpreted as general findings on the relative performance of OLS and LMS. In particular, we have to study in more detail the effect of the number of extreme events in the period considered. Furthermore, only the stocks of large companies over a long time period have been considered. It is certainly worth considering alternative stocks, e.g., high-tech stocks with small capitalization, and to identify sub-periods for which the relative performance of OLS and LMS differs most. Finally, we also considered the three factor model proposed by Fama and French (1992). The estimation results are available on request. We will refer to this model with regard to its forecasting performance in the next subsection.

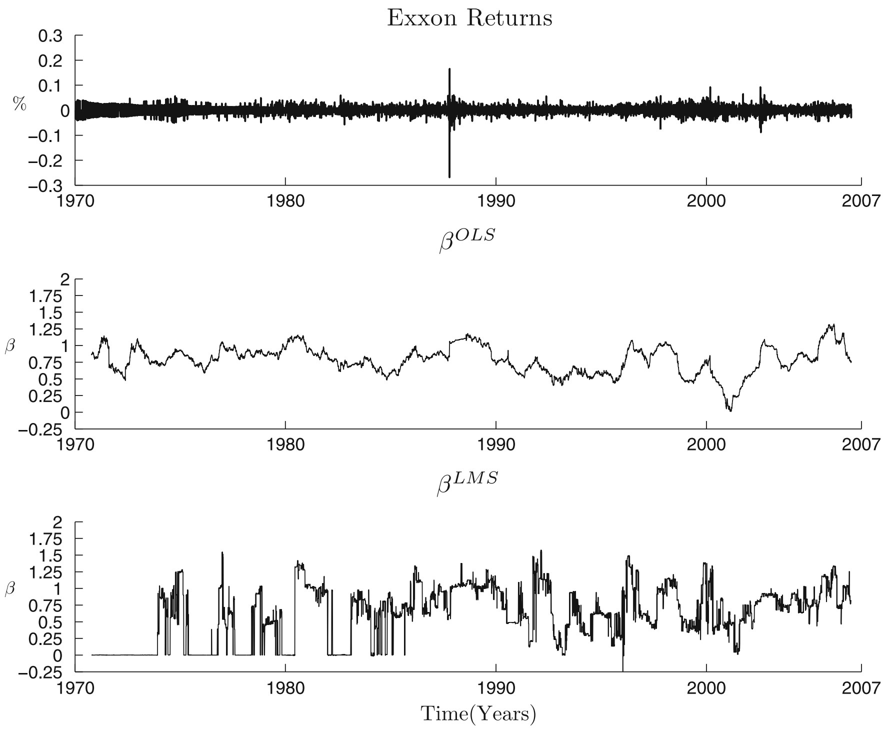

Fig. 4 Actual returns and estimates of β for Exxon (1971–2006)

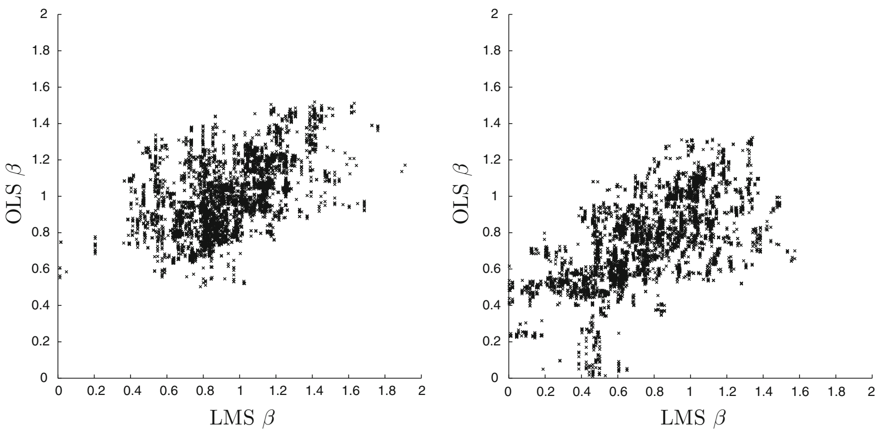

Fig. 5 Scatter plots of β estimates by OLS and LMS for IBM and Exxon

4.3 Forecasting performance

Although the estimates of the CAPM and the multi-factor model might be of interest on their own, the typical application consists in using them for (conditional) forecasting

Table 1 Forecast errors for LMS and OLS estimates of the CAPM

Stock	LMS	OLS
IBM		
MSE	0.1835×10^{-3}	0.1784×10^{-3}
MAE	0.0092	0.0090
XOM		
MSE	0.1628×10^{-3}	0.1882×10^{-3}
MAE	0.0092	0.0093
GE		
MSE	0.1760×10^{-3}	0.1409×10^{-3}
MAE	0.0090	0.0087
MRK		
MSE	0.2236×10^{-3}	0.1974×10^{-3}
MAE	0.0107	0.0099
GM		
MSE	0.2233×10^{-3}	0.2115×10^{-3}
MAE	0.0106	0.0103
BA		
MSE	0.3903×10^{-3}	0.3471×10^{-3}
MAE	0.0136	0.0133

(Simin 2008). Given the marked differences between LMS and OLS estimates for both models, it is of interest to see how these differences affect their predictive performance. To this end, we calculated the mean squared forecast error (MSE) and the mean absolute forecast error (MAE) for the one-period-ahead conditional forecasts for 9113 days for the CAPM and up to 555 months for the multi-factor model. Table 1 summarizes the findings for the CAPM.

For most stocks, the conditional forecasts based on OLS estimates exhibit both smaller MSE and MAE. The only exception is the XOM stock, for which the MSE can be reduced when using LMS instead of OLS. The first finding might have been expected for in sample forecasts as MSE is the objective function for OLS. Out of sample it does not have to hold, but in the absence of a large number of extreme observations it will. It is slightly more surprising that OLS based forecasts also dominate the MAE. Again, this might be a result of too few extreme events or outliers. Therefore, future work will be oriented towards identifying periods when the relative performance of LMS or other robust estimators might be expected to be better.

Table 2 reports the MSE and the MAE for the multi-factor model. Given that the model is based on monthly data, the number of extreme events is expected to be even smaller than for daily data. Consequently, no advantage of using a robust method like LMS might be expected under normal market conditions.[18]

So far, we might see our results as a further contribution to the rather disappointing evidence regarding the predictive performance of factor models (Simin 2008).

[18] When using a very short rolling window of length 25 months the difference between MSE and the MAE using LMS and OLS become smaller, but still the predictive performance of OLS appears to be superior.

Table 2 Forecast errors for LMS and OLS estimates of a multifactor model

Stock	LMS	OLS
IBM		
MSE	26.3819	9.7380
MAE	3.3231	2.3399
XOM		
MSE	11.7523	4.8583
MAE	2.4383	1.6077
GE		
MSE	16.6303	6.1978
MAE	2.9205	1.9088
MRK		
MSE	24.8864	21.9948
MAE	3.4414	3.5744
GM		
MSE	24.0990	17.5200
MAE	3.3519	2.7220
BA		
MSE	39.3633	21.4275
MAE	4.5106	3.4279

In particular, no improvement over conventional OLS based forecasts is apparent. However, these findings do not exclude that the LMS-based forecasts outperform the OLS approach at least under specific market conditions, e.g., high versus low volatility regimes. An analysis of this aspect is left for future research.

Finally, the LMS-based forecasts might still contain additional explanatory power which could be a reward for the high computational cost incurred. To analyze this possibility, we apply the test proposed by Chong and Hendry (1986) to the forecasts obtained from LMS and OLS, respectively. Let $\hat{r}^e_{LMS,t}$ and $\hat{r}^e_{OLS,t}$ denote the conditional forecasts of the excess returns from the LMS and OLS estimates, and r^e_t the actual excess return in period t. Then, estimation of the model

$$r^e_t = \gamma_0 + \gamma_1 \hat{r}^e_{LMS,t} + \gamma_2 \hat{r}^e_{OLS,t} + \nu_t \tag{6}$$

allows to test several hypotheses. If either γ_1 or γ_2 are equal to zero and $\gamma_0 = 0$, the forecasts associated with the nonzero γ_i is unbiased. If $\gamma_1 = 0$, the forecasts based on the OLS estimation dominate, i.e., the LMS based forecasts do not provide additional information. For $\gamma_2 = 0$ we obtain dominance for the LMS based forecasts. If both $\gamma_1 = 0$ and $\gamma_2 = 0$ has to be rejected, a combined forecast improves forecasting performance.

The results for the CAPM are again mixed. Only for the IBM stock, the null hypothesis that the forecasts based on the LMS estimates have no additional informational content can be rejected at the 5% level, while for the other stocks the OLS based forecasts are found to dominate. However, one has to keep in mind that the test is linked to MSE, for which the OLS estimators should be more suitable. Thus, finding relevant additional information content (the parameter γ_1 has a value close to 0.2 for IBM) demonstrates that considering alternative estimators might at least sometimes improve the predictive performance. The evidence changes when considering the multi-factor model. There, for all stocks considered both γ_1 and γ_2 are significantly different from zero. Although the parameter values are typically smaller for γ_1, we find a clear evidence that the conditional forecasts can be improved by combining the OLS-based forecasts with the LMS-based forecasts. Future research will focus on identifying the driving forces of this result and its robustness.

5 Rate of convergence

The results show that both optimization heuristics are able to provide high quality approximations of the LMS estimator. We are also interested in comparing their rate of convergence, i.e., the quality of the approximation as a function of computational time.[19] Typically, limits in time and computational resources make it unfeasible to obtain the global optimum in each run with certainty. However, by analyzing the distribution of outcomes for different parameter settings we can draw some conclusions on the convergence properties. Tables 3 and 4 provide a statistical summary of the results obtained by TA and DE for various parameter settings. Again, we consider the first 200 observations for the IBM stock and the CAPM as our test case.

In our experiments, we calculate the LMS estimators by DE for nine combinations of population size $n_p = \{20, 50, 100, 200\}$ and number of generations $n_G = \{50, 100, 200, 1000\}$. The scaling factor F and the crossover rate CR are kept constant at 0.8 and 0.9, respectively. In the case of TA, the first parameter represents the number of rounds (corresponding to neighborhood/threshold reductions), n_R, and the second parameter represents the number of neighborhood search iterations n_S per round. Nine combinations of n_R and n_S are selected from $n_R = \{20, 50, 100, 200\}$ and $n_S = \{50, 100, 200, 1000\}$.

The implementation process of all heuristics is subject to random effects—the TA algorithm generates a random solution from a neighborhood, and the DE algorithm generates an initial population of random solutions. Furthermore, the selection of candidate solutions in each search step is random. In order to obtain some information on the effect of these stochastic components on the results, we repeat both algorithms 1,000 times for each set of parameter values and report the best value, the median, the worst value, the variance, the 5th percentile, the 90th percentile, and the frequency of the best value occurring in all 1,000 repetitions.

Looking first at the TA results in Table 3, we can see that they improve significantly as the number of threshold reductions, n_R, and neighborhood iterations, n_S, increase. However, the best results obtained by TA is not better than any of the best DE results. Moreover, we observe that typically, the best results for a given parameter setting is found only a few times out of 1,000 restarts in each TA experiment.

[19] For a combined analysis of the rate of convergence of an estimator itself and the convergence of the approximation by a heuristic optimization algorithm, the reader is referred to Maringer and Winker (2009).

Table 3 Descriptive statistics for 1,000 repetitions of the LMS estimation using TA

n_R	n_S	Best value	Median	Worst value	Var	q5%	q90%	Freq
20	50	4.9935×10^{-5}	5.0557×10^{-5}	5.1863×10^{-5}	1.7801×10^{-13}	4.9935×10^{-5}	5.1175×10^{-5}	1
20	100	4.9936×10^{-5}	5.0253×10^{-5}	5.1299×10^{-5}	8.3278×10^{-14}	4.9936×10^{-5}	5.0761×10^{-5}	1
20	200	4.9935×10^{-5}	5.0123×10^{-5}	5.0963×10^{-5}	3.1553×10^{-14}	4.9935×10^{-5}	5.0408×10^{-5}	1
20	1,000	4.9935×10^{-5}	4.9980×10^{-5}	5.0360×10^{-5}	2.6529×10^{-15}	4.9935×10^{-5}	5.0065×10^{-5}	11
50	50	4.9935×10^{-5}	5.0234×10^{-5}	5.1245×10^{-5}	7.3395×10^{-14}	4.9935×10^{-5}	5.0689×10^{-5}	1
50	100	4.9936×10^{-5}	5.0102×10^{-5}	5.1108×10^{-5}	2.6521×10^{-14}	4.9936×10^{-5}	5.0363×10^{-5}	1
100	50	4.9935×10^{-5}	5.0115×10^{-5}	5.0995×10^{-5}	2.8602×10^{-14}	4.9935×10^{-5}	5.0378×10^{-5}	3
100	100	4.9935×10^{-5}	5.0040×10^{-5}	5.0548×10^{-5}	7.4462×10^{-15}	4.9935×10^{-5}	5.0177×10^{-5}	2
200	200	4.9935×10^{-5}	4.9963×10^{-5}	5.0127×10^{-5}	9.4226×10^{-16}	4.9935×10^{-5}	5.0013×10^{-5}	11

Table 4 Descriptive statistics for 1,000 repetitions of the LMS estimation using DE

n_p	n_G	Best value	Median	Worst value	Var	q5%	q90%	Freq
20	50	4.9935×10^{-5}	4.9935×10^{-5}	5.6897×10^{-5}	1.1849×10^{-13}	4.9935×10^{-5}	4.9972×10^{-5}	534
20	100	4.9935×10^{-5}	4.9935×10^{-5}	5.5428×10^{-5}	1.0954×10^{-13}	4.9935×10^{-5}	4.9935×10^{-5}	949
20	200	4.9935×10^{-5}	4.9935×10^{-5}	5.5428×10^{-5}	8.0223×10^{-14}	4.9935×10^{-5}	4.9935×10^{-5}	947
20	1,000	4.9935×10^{-5}	4.9935×10^{-5}	5.6897×10^{-5}	2.2572×10^{-13}	4.9935×10^{-5}	4.9935×10^{-5}	940
50	50	4.9935×10^{-5}	4.9935×10^{-5}	5.0934×10^{-5}	1.0532×10^{-15}	4.9935×10^{-5}	4.9938×10^{-5}	742
50	100	4.9935×10^{-5}	4.9935×10^{-5}	5.0934×10^{-5}	1.9939×10^{-15}	4.9935×10^{-5}	4.9935×10^{-5}	998
100	50	4.9935×10^{-5}	4.9935×10^{-5}	4.9936×10^{-5}	3.2327×10^{-29}	4.9935×10^{-5}	4.9935×10^{-5}	1,000
100	100	4.9935×10^{-5}	4.9935×10^{-5}	4.9935×10^{-5}	3.2328×10^{-29}	4.9935×10^{-5}	4.9935×10^{-5}	1,000
200	200	4.9935×10^{-5}	4.9935×10^{-5}	4.9935×10^{-5}	2.7964×10^{-37}	4.9935×10^{-5}	4.9935×10^{-5}	1,000

The results obtained for DE, which are shown in Table 4, contrast markedly with the results for TA. Here, we observe that DE converges close to the global optimum, in almost every restart when population size and generations are at least 50 and 100, respectively. The number of generations is the parameter which controls the consistency of the algorithm. Even with a population size of 20 the algorithm exhibits a high frequency of convergence when the number of generations is 100 or more. In fact, in this case, the best value is found in more than 90% of the repetitions as indicated by the 90%-quantile (q90%).

The superior performance of DE is attributed to the fact that the search is run on a continuous search space. Given that it is a population based approach, it is quite efficient in providing good approximations once the region of the global optimum is identified. A simple local search heuristic as TA, by contrast, will have to spend a large number of search steps with decreasing threshold values in such a region before it eventually approximates the global optimum up to several digits precision. However, as pointed out above, the LMS estimation problem could also be interpreted as the search on a discrete search space making use of ideas from elemental subset regression. Then, the relative performance of TA might improve substantially as the results by Fitzenberger and Winker (2007) for the related problem of censored quantile regression show.

Nevertheless, it is useful to report how much more efficient DE is than TA for the given problem formulation. DE and TA are different search heuristics, for which the main computational burden is the repeated calculation of the objective function. Therefore, we use this as the measure for efficiency. In each TA paired DE experiment, i.e., in the corresponding lines of Tables 3 and 4, we calculated the objective function the same number of times. Considering, e.g., for TA the experiment with $n_R = 20$ and $n_S = 1000$ (line 4), we find that it has a lower variance than the paired application of DE. Nevertheless, the best result for TA is still slightly worse than the best result obtained by DE. Again, this difference in efficiency might be attributed to the difficulties TA faces in the fine tuning of continuous variables. Looking at the 5th percentile we can state that in at least 5% of the repetitions both heuristics find the best solution.

To summarize, the results we obtain indicate the superiority of DE in terms of consistency and efficiency for LMS estimation, at least for the specific applications to the CAPM and multi-factor model and the datasets considered.

6 Conclusion and further work

The LMS estimator is considered for obtaining robust estimators of the parameters of the CAPM and a multi-factor model. It is shown that optimization heuristics like TA and DE are suitable to solve the resulting optimization problem. From our results, DE appears to be more efficient than TA, at least when the problem is considered as an optimization problem on a continuous parameter space. Making use of the ideas from elemental subset regression would allow to constrain the search on a discrete subset. Then, the relative performance of TA might improve. Such a comparison is left to future research.

In fact, efficient implementations of DE allow a fast and reliable estimation of the parameters of both models by LMS. This is demonstrated by a rolling window analysis on a sample of six publicly traded firms with daily data for the CAPM (1970–2006) and monthly data for the multi-factor model (1970–2008). The results indicate that the estimates obtained by LMS differ substantially from those resulting from OLS. However, the LMS estimates do not exhibit less variation as might have been expected from the outlier related argument. Furthermore, the relative performance of both estimators in a simple one-period-ahead conditional forecasting experiment is mixed. In most cases, both the MSE and the MAE are smaller for the conditional forecasts based on the OLS estimation. However, it is shown that the forecasts based on the LMS estimators provide additional informational content. Thus, it might be justified to incure the additional computational load for obtaining the LMS estimators.

Some extensions of the paper are straightforward based on the results presented. First, the method should be applied to different data sets, e.g., stock returns from other stock indices or stock markets. Furthermore, it would be of interest to identify in more detail the situations where the estimation and forecasts based on LMS outperform OLS and vice versa.

References

Barreto H, Maharry D (2006) Least median of squares and regression through the origin. Comput Stat Data Anal 50(6):1391–1397
Chan LKC, Lakonishok J (1992) Robust measurement of beta risk. J Financ Quant Anal 27:265–282
Chong YY, Hendry DF (1986) Econometric evaluation of linear macro-economic models. Rev Econ Stud 53(175):671–690
Cornell B, Dietrich JK (1978) Mean-absolute-deviation versus least-squares regression estimation of beta coefficients. J Financ Quant Anal 13:123–131
Dueck G, Winker P (1992) New concepts and algorithms for portfolio choice. Appl Stoch Models Data Anal 8:159–178
Dueck G, Scheuer T (1990) Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. J Comput Phys 90:161–175
Engel CM, Rodrigues AP (1993) Tests of mean-variance efficiency of international equity markets. Oxford Econ Papers 45:403–421
Fama EF, French KR (1992) The cross-section of expected stock returns. J Finance 47:427–465
Fama EF, French KR (1993) Common risk factors in the returns on stocks and bonds. J Financ Econ 33:3–56
Fitzenberger B, Winker P (2007) Improving the computation of censored quantile regressions. Comput Stat Data Anal 52(1):88–108
Gilli M, Winker P (2007) Editorial—2nd special issue on applications of optimization heuristics to estimation and modelling problems. Comput Stat Data Anal 52(1):2–3
Gilli M, Winker P (2009) Heuristic optimization methods in econometrics. In: Kontoghiorghes E, Belsley D (eds) Handbook on computational econometrics, chap 3. Wiley, Chichester, pp 81–119
Gilli M, Maringer D, Winker P (2008) Applications of heuristics in finance. In: Seese D, Weinhardt C, Schlottmann F (eds) Handbook of information technoloy in finance, chap 26, pp.635–653. International Handooks on Information Systems. Springer, Berlin
Huber PJ (1973) Robust regression: asymptotics, conjectures and Monte Carlo. Ann Stat 1:799–821
Ince H (2006) Non-parametric regression methods. Comput Manage Sci 3:161–174
Knez PJ, Ready MJ (1997) On the robustness of size and book-to-market in cros-sectional regressions. J Finance 52(4):1355–1382
Krink T, Paterlini S, Restic A (2007) Using differential evolution to improve the accuracy of bank rating systems. Comput Stat Data Anal 52(1):68–87
Lintner J (1965) The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. Rev Econ Stat 47(1):13–37

Maringer D (2005) Distribution assumptions and risk constraints in portfolio optimization. Comput Manage Sci 2(2):139–153
Maringer DG, Meyer M (2008) Smooth transition autoregressive models—new approaches to the model selection problem. Stud Nonlinear Dyn Econom 12(1):Article 5
Maringer D, Winker P (2009) The convergence of estimators based on heuristics: theory and application to a GARCH model. Comput Stat 24:533–550
Markowitz HM (1952) Portfolio selection. J Finance 7(1):77–91
Maronna RA, Martin RD, Yohai VJ (2006) Robust statistics: theory and methods. Wiley, Chichester
Martin RD, Simin T (2003) Outlier resistant estimates of beta. Financ Anal J 59:56–69
Mayo MS, Gray JB (1997) Elemental subsets: the building blocks of regression. Am Stat 51(2):122–129
Mossin J (1966) Equilibrium in a capital asset market. Econometrica 34(4):768–783
Price KV, Storn RM, Lampinen JA (2005) Differential evolution: a practical approach to global optimization. Springer, Germany
Roko I, Gilli M (2008) Using economic and financial information for stock selection. Comput Manage Sci 5:317–335
Ronchetti E, Genton M (2008) Robust prediction of beta. In: Kontoghiorghes E, Rustem B, Winker P (eds) Computational methods in financial engineering. Springer, Berlin, pp 147–161
Rousseeuw PJ (1984) Least median of squares regression. J Am Stat Assoc 79:871–880
Rousseeuw PJ, Leroy AM (1987) Robust regression and outlier detection. John Wiley & Sons, New York
Rousseeuw PJ, Wagner J (1994) Robust regression with a distributed intercept using least median of squares. Comput Stat Data Anal 17(1):65–76
Sharpe W (1971) Mean-absolute deviation characteristic lines for securities and portfolios. Manage Sci 18:1–13
Sharpe WF (1964) Capital asset prices: a theory of market equilibrium under conditions of risk. J Finance 19(3):425–442
Simin T (2008) The poor predictive performance of asset pricing models. J Financ Quant Anal 43(2):355–380
Specht K, Winker P (2008) Portfolio optimization under VaR constraints based on dynamic estimates of the variance–covariance matrix. In: Kontoghiorghes E, Rustem B, Winker P (eds) Computational methods in financial engineering. Springer, Berlin pp 73–94
Storn R, Price K (1997) Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces. J Global Optim 11:341–359
Winker P (2001) Optimization heuristics in econometrics: applications of threshold accepting. Wiley, New York
Winker P, Fang K-T (1997) Application of threshold accepting to the evaluation of the discrepancy of a set of points. SIAM J Numer Anal 34(5):2028–2042
Winker P, Maringer D (2007a) The hidden risks of optimizing bond portfolios under VaR. J Risk 9(4):1–19
Winker P, Maringer D (2007b) The threshold accepting optimisation algorithm in economics and statistics. In: Erricos Kontoghiorghes J, Gatu C (eds) Optimisation, econometric and financial analysis. Advances in computational management science, vol 9. Springer, Berlin, pp 107–125
Yang Z, Tian Z, Yuan Z (2007) GSA-based maximum likelihood estimation for threshold vector error correction model. Comput Stat Data Anal 52(1):109–120
Zaman A, Rousseeuw PJ, Orhan M (2001) Econometric applications of high-breakdown robust regression techniques. Econ Lett 71:1–8