Portfolio Optimization in **parma** (Version 1.5-0)

Alexios Ghalanos

October 19, 2013

Abstract

The **p**ortfolio **a**llocation and **r**isk **m**anagement **a**pplications (**parma**) package provides a set of models and methods for use in the allocation and management of capital in financial portfolios. It uniquely represents certain discontinuous problems using their smooth approximation counterparts and implements fractional based programming for the direct optimization of risk-to-reward ratios. This paper forms an introduction to the models and methods.

1 Introduction

Generally speaking, the portfolio management life-cycle is the process of allocating and managing investment capital with respect to a set of assumptions on the market dynamics and available universe of investable assets. The assumptions made in the model building stage will generally guide decisions on allocation, while trading and monitoring those allocations forms part of risk management. Figure 1 succinctly illustrates the decision making process of model builing, allocation and management (*MAM*).

The **p**ortfolio **a**llocation and **r**isk **m**anagement **a**pplications (**parma**) package provides a rich

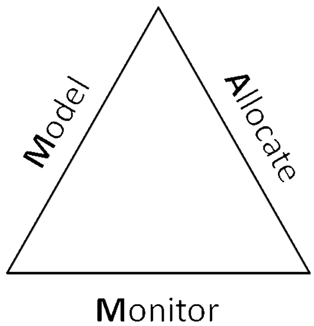

Figure 1: MAM's the word

subset of models and methods for use in the portfolio allocation process. The models are classified according to the problem\program class they belong to, namely Linear *LP*, Mixed Integer LP (*MILP*), Quadratic (*QP*), Mixed Integer QP (*MIQP*), Quadratically Constrained QP (*QCQP*), Non-Linear Convex (*NLP*), Mixed Integer NLP (*MINLP*) and Non-Linear Non-Convex (*GNLP*), the latter belonging to the Global Optimization (*GO*) type of problems. Membership in these general problem classes is determined by the intersection of the objective and constraint function space. Table 1 summarizes the types of problems currently supported in the **parma** package which is completely dependent on the availability of high quality solvers of the given types in R. A range of popular risk measures are implemented, with the ability to perform both risk

Table 1: Problems and Solvers in the **parma** package.

Problem Type **Solver**	**LP** **Rglpk**	**MILP** **Rglpk**	**QP** **quadprog**	**MIQP** **.**	**QCQP** **Rsocp**	**NLP** **nloptr**	**MINLP** **.**	**GNLP** **cmaes/crs**
Scenario [minrisk]	T	T	.	.	.	T	.	T
+ LP custom constraints	T	.	.	.	.	.	.	.
+ NLP custom constraints	.	.	.	.	.	T	.	T
+ cardinality constraints	.	T	.	.	.	.	TBI	T
Scenario optrisk	T	.	.	.	.	T	.	T
+ LP custom constraints	T	.	.	.	.	.	.	.
+ NLP custom constraints	.	.	.	.	.	T	.	T
+ cardinality constraints	.	.	.	.	.	.	.	T
Covariance minrisk	.	.	T	.	.	.	.	.
+ LP custom constraints	.	.	T	.	.	.	.	.
+ Q custom constraints	.	.	.	.	TBI	.	.	.
+ cardinality constraints	.	.	.	F	.	.	.	.
Covariance optrisk	.	.	T	.	.	.	.	.
+ LP custom constraints	.	.	T	.	.	.	.	.
+ Q custom constraints	.	.	.	.	.	.	.	.
+ cardinality constraints	.	.	.	.	.	.	.	.

Note: The Table presents the types of problems which the **parma** package can solve based on the available solvers, where *T* means that a problem can be solved by the particular solver/type, *F* that there is currently no available solver for this type of problem, and *TBI* that a solver will be implemented for that particular problem in due course. Note that for the scenario based optimization, all risk measures which can be represented (without additional restrictions) as LP can also be represented as NLP, with the exception of Conditional Drawdown at Risk for which only an LP formulation exists. The covariance based optimization refers to the EV model which may also be solved using a scenario model as NLP.

minimization or optimal risk-reward optimization using fractional programming methods. The package makes use of smooth approximations to discontinuous functions to represent risk measures such as CVaR and LPM, as well as leverage in long-short optimization, as proper convex and continuous NLP problems, making use of analytic gradients for consistent and confident solutions. For problems which cannot be represented as convex NLPs, experimental support for global optimization is provided using derivative free penalty functions but it is up to the user to decide how optimal such an approach is. Serious users of GNLP problems should consider plugging in their own high quality global optimization solvers the majority of which are either non GPL or reside in another language[1].
Finally, a separate set of functions is also included for utility maximization based on a Taylor Series expansion, taking as inputs moment and co-moment matrices.
This document is organized as follows: Section 2 briefly discusses the scenario approach to decision making in the context of stochastic programming. Section 3 presents and discusses the risk measures implemented in the package, and some recent topics in the definitions and properties these measures. Section 4 discusses fractional programming and the derivation of the optimal risk-reward portfolios. Section 5 presents details of the implementation of the models in the parma package, with a particular focus on the smooth approximations used, and Section 6 a small FAQ section. Examples and demos may be found online at:
`http://www.unstarched.net/r-examples/parma/`.

The package is provided AS IS, without any implied warranty as to its accuracy or suitability. A lot of time and effort has gone into the development of this package, and it is offered under the GPL-3 license in the spirit of open knowledge sharing and dissemination. If you do use the model in published work DO remember to citet the package and author (type **citation(”parma”)** for the appropriate BibTeX entry).

2 Uncertainty and Scenario Based Allocation

Randomness in the underlying environment leads to uncertainty, which can be characterized, albeit approximately, by a model with a probability distribution. The uncertainty is by no means resolved, but simply structured under a set of assumptions for enabling decision making, by assigning some probability to some ’unknowns’ so that they become ’known unknowns’. The purpose of stochastic programming (SP) is to incorporate such uncertainty into the objective or constraint functions with a view to obtaining an optimal set of decisions. This is done by constructing a scenario, or set of scenarios, representing the possible future path or paths of the underlying process (as a discrete time approximation to the continuous case) incorporating the uncertainty with respect to the model and future, and from which decisions can be based. These types of models were originally proposed and analyzed among others by Dantzig (1956, 1992), Beale [1955], Dantzig and Infanger [1993], Madansky [1962] and Charnes and Cooper [1959]. An excellent exposition of SP models in asset and liability management can be found in Kouwenberg and Zenios [2006]. Generally speaking, stochastic programming covers a spectrum of uncertainty from that of complete information (distribution model) to no information (anticipative model), with partial information allowing for adaptation and multistage models with recourse. In the **parma** package, only single stage anticipative models are considered at present taking as inputs 1-period ahead scenarios of the simulated discrete approximation to the multivariate conditional

[1] As a first attempt at providing a reasonable Global solver for such problems, a ported (from Matlab) version of the cmaes solver of Hansen [2006] is made available in this package with a rather comprehensive set of control options.

density. For this purpose, the *fscenario* function in the **rmgarch** package provides for a set of parametric data generating processes from which it is possible to generate such scenarios quite easily, as illustrated below:

```
> require(rmgarch)
> args(fscenario)

function (Data, sim=1000, roll=0,
    model=c("gogarch", "dcc","cgarch", "var", "mdist"), spec=NULL,
    var.model=list(lag=1,lag.max=NULL,
        lag.criterion=c("AIC", "HQ", "SC", "FPE"), robust=FALSE,
        robust.control=list(gamma=0.25, delta=0.01, nc=10, ns=500)),
    mdist.model=list(distribution=c("mvn","mvt", "manig"), AR=TRUE, lag=1),
    spd.control=list(lower=0.1,upper=0.9, type="pwm", kernel="epanech"),
    cov.method=c("ML", "LW", "EWMA", "MVE", "MCD", "MVT", "BS"),
    cov.options=list(shrinkage=-1, lambda=0.96), solver="solnp",
    solver.control=list(), fit.control=list(eval.se=FALSE),
    cluster=NULL, save.output=FALSE, save.dir=getwd(),
    save.name=paste("S", sample(1:1000,1), sep=""), rseed=NULL, ...)
```

The models currently supported, are the 3 multivariate GARCH models GO-GARCH, DCC and GARCH-Copula (DCC or non dynamic based) and the Vector AR model with optional choice the type of covariance matrix used (*cov.method*) for the generation of the conditional multivariate random Normal errors in the scenario. The choice of non dynamic multivariate distribution (*mdist*) is not yet implemented. The option for parallel processing is provided by passing a pre-created cluster object from the parallel package, and the replication of results by passing seeding values (*rseed*) to the random number generator (either a single integer or a vector of integers of length roll+1). In addition, the scenarios which may span several thousand rows and include rolling simulated forecasts of the conditional multivariate density (useful in backtesting) can optionally be saved to file instead of being returned to the workspace, where the function *goload* reconstitutes previously saved data with an object from such an operation. In addition to the scenario based mechanism, the **fmoments** function generates conditional moment based forecasts from the GO-GARCH and DCC models for use in the quadratic mean-variance (EV) model as well as a separate function which maximizes utility based on a Taylor series approximation of CARA (*parmautility*), for which it is possible to use, beyond the mean and covariance, higher co-moment tensors generated from the GO-GARCH with maNIG (or the more general maGH) distribution.

```
> args(fmoments)

function (spec, Data, n.ahead=1, roll=0, solver="solnp",
    solver.control=list(), fit.control=list(eval.se=FALSE),
    cluster=NULL, save.output=FALSE, save.dir=getwd(),
    save.name=paste("M", sample(1:1000, 1), sep=""), ...)
```

Scenarios or moments, whether they are derived from these auxiliary **rmgarch** wrapper functions or a user's own programs, then form part of the **parmaspec** function which defines the type of problem to be optimized:

```
> args(parmaspec)

function (scenario=NULL, S=NULL, benchmark=NULL, benchmarkS=NULL,
    forecast=NULL, target=NULL, targetType=c("inequality", "equality"),
    risk=c("MAD", "MiniMax", "CVaR", "CDaR", "EV","LPM", "LPMUPM"),
    riskType=c("minrisk", "optimal"),
    options=list(alpha=0.05, threshold=999, moment=1), LB=NULL, UB=NULL,
    budget=1, leverage=NULL, ineqfun=NULL, ineqgrad=NULL, eqfun=NULL,
    eqgrad=NULL, uservars=list(), ineq.mat=NULL, ineq.LB=NULL, ineq.UB=NULL,
    eq.mat=NULL, eqB=NULL, max.pos=NULL, asset.names=NULL, ...)
```

The function consists of the data inputs: either a scenario or covariance matrix and related optional benchmark details (for benchmark relative optimization), the forecast and optionally portfolio target and whether this should be a hard equality or inequality, the risk measure and optimization method, and remaining arguments relating to the constraints. Note that NLP based constraints in the **parma** package need to conform to a certain form, an example of which is available in the *parma.tests* folder, and in order for the problems to remain convex the inequality must be convex and the equality affine. Departures from these guidelines means that you are not guaranteed a global optimum. More details can be found in the documentation and the online examples, whilst the types of risk/deviation measures supported and their implementation are described in the next section.

3 Risk and Deviation Measures

In portfolio and resource allocation, characterization of the future uncertainty by a scenario of possible outcomes does not in itself provide value to the decision maker unless he is able to rank, choose and allocate among competing alternatives based on a set of preferences. Historically, theories of such preferences have been normative, describing a certain set of principles or axioms for rational behavior. The expected utility theory, first proposed by Bernoulli [1954] as a solution to the St.Petersburg Paradox[2], and formalized by Von Neumann and Morgenstern [1944] into 4 key axioms (Completeness, Transitivity, Independence, Continuity), provides the most popular approach[3] to rational decision making. Risk attitudes in expected utility theory are usually measured by the Arrow-Pratt (see Arrow [1963]) definitions of absolute and relative risk aversion (ARA and RRA respectively) which are standardized measures of the degree of curvature in the utility functions[4] Utility functions of the form $U(W) = -\exp(-\lambda W)$, for instance, have constant absolute risk aversion, which the *paramutility* function implements based on a 2 and 4 moment Taylor series approximation.
In an attempt to depart from the utility framework altogether and to make use of criteria based on more objective and concrete concepts, a parallel strand of research has focused mainly on the concept of loss aversion. A first attempt at quantifying risk as the loss beyond a certain threshold was the Safety-First criterion of Roy [1952] which aimed at minimizing the probability of being below an investor's minimum acceptable return (MAR). Later concepts have looked at improving on this measure by penalizing losses below the threshold at different rates representing different preferences. Irrespective of the type of measure, the more general reward-risk approach has proved very popular both academically and in practise since it enables preferences to be

[2]Where the distinction between expected utility and expected return was made.

[3]Though by no means the only approach. See for example Savage [1962] for subjective expected utility, Quiggin [1982] and Schmeidler [1989] for rank dependent utility and Zadeh [1965] for Fuzzy Logic.

[4] Formally, $ARA(W) = -\frac{U''(W)}{U'(W)}$ and $RRA(W) = -\frac{WU''(W)}{U'(W)}$.

summarized in a few scalar parameters such as the mean and variance. However, it was not until recently that formal qualifications of the properties of such measures where defined in seminal papers by Artzner et al. [1999] and Acerbi [2002] on risk and Rockafellar et al. [2006] on deviation, with the latter establishing the connection between the two and briefly described here. Consider the probability space $\{\Omega, \mathcal{A}, P\}$ where P is the probability on the $\mathcal{A}$ measurable subsets of Ω. Rockafellar et al. [2006] defined a set of axioms which functions in the linear space $\mathcal{L}^2$ (which include the mean and variance) should satisfy. Formally, the deviation measure functionals $\mathcal{D} : \mathcal{L}^2(\Omega) \to [0, \infty]$ should satisfy the following axioms:

- (D1) $\mathcal{D}(C) = 0 \ \forall$ constants C,
- (D2) $\mathcal{D}(\lambda X) = \lambda \mathcal{D}(X) \ \forall$ X and $\lambda > 0$,
- (D3) $\mathcal{D}(X + X') \leq \mathcal{D}(X) + \mathcal{D}(X') \ \forall$ X and X',
- (D4) $\mathcal{D}(X) \geq 0 \ \forall$ X and $\mathcal{D}(X) > 0 \ \forall$ nonconstant X,

where (D1) is the translation invariance property under the special condition given for constants (i.e. insensitivity to constant shifts), (D2) represents the positive homogeneity property, (D3) the subadditivity property , while (D4) is the lower bound implied by the domain of $\mathcal{D}$. Artzner et al. [1999] provides the equivalent 'coherent' risk measure functionals $\mathcal{R} : \mathcal{L}^2(\Omega) \to (-\infty, \infty]$ which should satisfy the following axioms:

- (R1) $\mathcal{R}(C) = -C \ \forall$ constants C,
- (R2) $\mathcal{R}(\lambda X) = \lambda \mathcal{R}(X) \ \forall$ X and $\lambda > 0$,
- (R3) $\mathcal{R}(X + X') \leq \mathcal{R}(X) + \mathcal{R}(X') \ \forall$ X and X',
- (R4) $\mathcal{R}(X) \leq \mathcal{R}(X')$ whenever $X \geq X'$,

where (R1) is the translation invariance property, (R2) is positive homogeneity, (R3) subadditivity property and (R4) the monotonicity property. More plainly, (R1) implies that adding a constant to a set of losses does not change the risk, (R2) that holdings and risk scale by the same linear factor, (R3) that portfolio risk cannot be more than the combined risks of the individual positions, and (R4) that larger losses equate to larger risks. Acerbi [2002] defined the family of spectral risk measures as those with weighted[5] quantiles, possessing the properties of coherent risk measures and additionally:

- (R5) If $\mathcal{F}(X) = \mathcal{F}(Y)$, then $\mathcal{R}(X) = \mathcal{R}(Y)$,

which essentially implies that portfolios with equal cumulative distribution functions ($\mathcal{F}$) should have the same risk. Rockafellar et al. [2006] defined a one-to-one relationship between deviation and risk measures[6] which satisfy properties (R1),(R2),(R3) and are strictly expectation bounded so that $\mathcal{R}(X) > E[-X]$ as:

- $\mathcal{D}(X) = \mathcal{R}(X - E[X])$,
- $\mathcal{R}(X) = E[-X] + \mathcal{D}(X)$.

[5] With positive weights which are normalized to sum to 1.

[6] In the rest of this paper, I will refer to 'risk' to mean both risk and deviation measures.

In the following subsections, I consider the properties and representations of 5 interesting and popular measures which are implemented in the **parma** package . The first 3 loosely belong to the general L^p function space[7] and include the Absolute Deviation, Variance and Minimax measures, while the other 2 are the threshold based measures of Lower Partial Moments (LPM) and Conditional Value at Risk ($CVaR$).

3.1 Mean Variance (EV)

Markowitz [1952] ushered in the era of modern portfolio management with the introduction of the Mean-Variance model of risk-return. Variance is a valid measure of risk for ranking preferences if either the investor exhibits quadratic utility (in which case it does not matter whether the underlying data is multivariate normal), or the underlying data is multivariate normal (in which case the utility of the investor is irrelevant since variance is the optimal choice). The optimization problem may be posed as the following NLP problem:

$$
\begin{aligned}
&\min_{w} \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m} w_j\left(r_{i,j}-\mu_j\right)\right)^2\\
&s.t.\\
&\sum_{j=1}^{m} w_j\mu_j = C\\
&\sum_{j=1}^{m} w_j = 1\\
&w_j \geq 0, \forall j \in \{1,\ldots,m\}
\end{aligned}
\tag{2}
$$

where w represent the weights of the $j = 1,\ldots,m$ assets, $i = 1,\ldots,n$ are the number of periods or scenario points for the returns r and μ_j the forecast return. The problem effectively minimizes portfolio variance subject to the portfolio forecast return being equal to C, a full investment constraint and positivity constraints on the weights. While it is simple to express the problem in its quadratic form such that variance is equal to $w'\Sigma w$, I leave the problem in its more general NLP form which admits nonlinear constraints which would for example include long-short optimization with a leverage constraint.[8] Criticisms of variance as a valid method for ranking portfolios is mainly aimed at the quadratic utility assumption which seems nothing more than a mathematical convenience rather than a reflection of reality, leading to the strange case of investors desiring less to more after a certain point on the utility curve, whilst the multivariate normality assumption is not usually borne out by the data. The symmetric nature of variance, penalizing both up and down deviations at the same rate was criticized by quite early on by Hanoch and Levy [1969][9], while its lack of consistency with stochastic dominance relations should have effectively buried it as a method for portfolio allocation. However, its

[7] The L^p function space is defined as:

$$
\|e\|_p = \left(\sum_{j=1}^{m} |e_j|^p\right)^{1/p} \tag{1}
$$

with $p = 1$ representing the absolute (or Manhattan distance) measure, $p = 2$ the standard deviation (or Euclidean distance) where we can make use of variance instead because of the monotone transformation property, and $p = \infty$ represents the largest absolute value where we can represent the losses for Minimax optimization.

[8] In the case of quadratic based constraints, the problem can also be posed as a second order cone ($SOCP$) problem which will be implemented in a future update to the package.

[9] The criticism was in fact also aimed at any symmetric dispersion measure, not just variance.

tractability and ease of use has made it a very popular choice, particularly for the modelling of monthly returns, with numerous extensions to provide for robustness and uncertainty mainly in the derivation of the covariance matrix. For example James and Stein [1956] provide for a shrinkage estimator, Black and Litterman [1992] a semi-Bayesian approach while Michaud [1989] a general criticism of the approach with a patented alternative based on resampling methods. The Vector AR model in the package allows for a choice of covariance estimators to be chosen for the simulation of the random scenario matrix.

3.2 Mean Absolute Deviation (MAD)

In the early days of computer programming, large scale quadratic problems were computationally more demanding to solve than linear problems. In light of this, Konno and Yamazaki [1991] introduced a piece-wise linear formulation of the absolute deviation function as an alternative to the Markowitz [1952] method. The standard NLP objective function may be formulated as:

$$\frac{1}{n}\sum_{i=1}^{n}\left|\sum_{j=1}^{m} w_j\left(r_{i,j}-\mu_j\right)\right| \tag{3}$$

which Konno and Yamazaki [1991] reduced to the following piece-wise linear problem:

$$\begin{aligned}
&\min_{w,d}\frac{1}{n}\sum_{i=1}^{n} d_i\\
&s.t.\\
&\sum_{j=1}^{m}\left(r_{i,j}-\mu_j\right)w_j \leq y_i, \forall i\in\{1,\ldots,n\}\\
&\sum_{j=1}^{m}\left(r_{i,j}-\mu_j\right)w_j \geq -y_i, \forall i\in\{1,\ldots,n\}\\
&\sum_{j=1}^{m} w_j\mu_j = C\\
&\sum_{j=1}^{m} w_j = 1\\
&w_j \geq 0, \forall j\in\{1,\ldots,m\}
\end{aligned} \tag{4}$$

where d represent the absolute deviations of the portfolio from its forecast mean, forming a vector of variables of size n (length of the scenario) to be optimized. However, the constraints imposed to create the piece-wise linear function for the absolute deviation requires two $n \times n$ diagonal matrices stacked together[10] which may lead to computer memory problems for very large scenarios. This is in direct contrast to the EV model which only depends on the number of assets. Furthermore, while in the EV model deviations from the mean are penalized at an increasing rate arising from the square function, in the MAD model deviations are penalized at a linear rate which may not realistically represent the average investor. However, by not giving undue weight to the extreme observations, the MAD model may be more robust to possible misspecification in the dynamics from which the scenario was generated. Extensions to the model have included the addition of skewness in Konno et al. [1993], and semi-absolute

[10] Feinstein and Thapa [1993] provide for a reformulated representation with only one n diagonal matrix.

deviation first suggested by Speranza [1993] who showed that the mean semi-deviation is a half of the mean absolute deviation from the mean. Similar to the EV model, the MAD model lacks consistency with stochastic dominance relations.

3.3 Minimizing Regret (MiniMax)

The MiniMax model of Young [1998], aims to minimize the maximum loss, $\max\left(\sum_{j=1}^{m} -r_{i,j}w_j, \forall i = 1, \ldots, n\right)$ and as such is a very conservative criterion. It has a very simple LP formulation:

$$
\begin{aligned}
&\min_{M_p,w} M_p \\
&s.t. \\
&M_p - \sum_{j=1}^{m} w_j r_{i,j} \leq 0, \forall i = \{1, \ldots, n\} \\
&\sum_{j=1}^{m} w_j \mu_j = C \\
&\sum_{j=1}^{m} w_j = 1 \\
&w_j \geq 0, \forall j \in \{1, \ldots, m\}
\end{aligned}
\tag{5}
$$

where M_p is the objective minimization value representing the maximum loss of the portfolio and guaranteed to be bounded from above by the maximum portfolio loss as a result of the first constraint. While Young [1998] only considered the problem in light of historical scenarios, there is no reason why r in the formulation may not represent a future simulated forecast scenario. Contrary to the MAD model, it only requires 1 additional variable and $n \times 1$ additional constraint vector in the LP formulation, and as such does not pose any computational challenges even for very large problems. The Minimax principle is also consistent with expected utility theory at the limit based on a very risk averse decision maker, and a good approximation to the EV model when returns are multivariate Normal. Interestingly, the model is also a limiting case of the Conditional Value at Risk spectral risk measure described in the Section 3.5.

3.4 Lower Partial Moments (LPM)

The concept of penalizing deviations below a certain threshold at a different rate is at the heart of modern risk management and was already hinted at by Markowitz [1952] in a reference to semi-standard deviation. This was later formalized into a very general class of measures by Stone [1973], and the Lower Partial Moment (LPM) framework of Fishburn [1977] which, in the continuous case, may be defined as:

$$LPM_{a,\tau}(f) = \int_{-\infty}^{\tau} (\tau - x)^a f(x)\, dx \tag{6}$$

where a is some positive number which represents the rate at which deviations below the threshold τ are penalized and f some density function. In the discrete case, the function may be represented as:

$$LPM_{a,\tau}(x) = E\left[\max(\tau - x, 0)^a\right]. \tag{7}$$

Upper Partial Moments (UPM) are defined similarly. Usually, in the portfolio optimization context, the measure is standardized by raising it to the power of $\frac{1}{a}$. Fishburn [1977] derived a utility representation for this measure consistent with the von Neumann-Morgenstern axioms, and represented as:

$$U(x) = \begin{array}{cc} x - k(\tau - x)^{a_l} & x < \tau \\ x & x \geq \tau \end{array} \tag{8}$$

where k is a positive constant. Harlow and Rao [1989] describe an asset pricing model in the mean-lower partial moment framework (MLPM) and show that an MLPM framework is consistent with a very general set of utility functions. For example, the hyperbolic absolute risk aversion ($HARA$) class of utility functions is consistent with 1^{st}-degree LPM, whereas any risk averse utility function displaying skewness preference with positive first and third derivatives and negative second derivatives are consistent with 2^{nd}-degree LPM. In addition to this strong link with expected utility theory, Bawa and Lindenberg [1977], Bawa [1978] and Fishburn [1977] showed that stochastic dominance is equivalent to all degrees of n-degree LPM.
The portfolio optimization problem can be posed as follows:

$$\begin{aligned}
&\min \left(\frac{1}{n} \sum_{i=1}^{n} \max \left(0, \tau - \left(\sum_{j=1}^{m} w_j r_{j,i} \right) \right)^a \right)^{1/a} \\
&s.t. \\
&\sum_{j=1}^{m} w_j \mu_j = C \\
&\sum_{j=1}^{m} w_j = 1 \\
&w_j \geq 0, \forall j \in \{1, \ldots, m\}
\end{aligned} \tag{9}$$

Special cases are $a = 0$ representing the shortfall probability or Safety-First model of Roy [1952], $a = 1$ the below target shortfall and $a = 2$ the shortfall variance which is equivalent to the central semi-variance when $\tau = E(x)$. When $a = 1$, an LP formulation exists and is given by:

$$\begin{aligned}
&\min_{w} \frac{1}{n} \sum_{i=1}^{n} d_i \\
&s.t. \\
&\tau - \sum_{j=1}^{m} w_j r_{j,i} \leq d_i, \forall i \in \{1, \ldots, n\} \\
&\sum_{j=1}^{m} w_j \mu_j = C \\
&\sum_{j=1}^{m} w_j = 1 \\
&w_j \geq 0, \forall j \in \{1, \ldots, m\} \\
&d_i \geq 0, \forall i \in \{1 \ldots, n\}
\end{aligned} \tag{10}$$

The **parma** package implements the general LPM as NLP, but also includes the special case when $a = 1$ as an LP formulation. For positive values of a other than 1, the discontinuous

max function appears to pose some problems in the optimization strategy. Nawrocki and Staples [1989] devised a heuristic measure which approximates the function using only quadratic programming methods. Instead, I replace the max function with a smoothed approximation for which derivatives exist and discussed further in Section 5. With regards to the choice of threshold variable τ, the choice may be motivated by the investor's minimum acceptable return, some benchmark rate[11] or any other reasonable choice. A simple choice which makes use of the properties of this deviation measure is to use the mean of the portfolio (this is implemented by passing threshold=999 in the options to the **parmaspec** function) which is equivalently equal to using a threshold of zero and passing a demeaned scenario matrix.[12]
Because the linear reward function may be too restrictive in practise, Holthausen [1981] extended the LPM model to include a non-linear reward measure so that for $x \geq \tau$ in (8) $U(x) = x + (x - \tau)^{a_u}$, where a_u is the power exponent for the upper partial moment, thus effectively capturing a range of linear and nonlinear utility curves (such as S-shaped and inverse S-shaped) with reference to gains and losses as illustrated in the example in Figure 2. Unfortunately, this measure is non-convex and pretty hard to optimize with confidence. At present the **parma** package experimentally supports Global Optimization (GO) using a derivative free penalty function approach and this model may be optimized under this setup (the *LPMUPM* measure). However, without making use of advanced state of the art commercial global optimization solvers, it is quite hard to gauge the optimality of this solution and the user should not expect too much as things currently stand.

3.5 Conditional Value at Risk (CVaR)

Since the report by the Group of Thirty G30 [1993], the use of Value at Risk (VaR) is almost universal among banks, trading desks and other financial entities as a key measure for measuring and managing risk. Despite its popularity, it has come under growing pressure as a non coherent (it lacks the subadditivity property) and inadequate measure, or an imperfect measure which which has been incorrectly used, overused, abused and over-relied upon. An alternative measure, based on the average loss conditional on the VaR being violated is called Conditional Value at Risk (CVaR)[13] which is a coherent and convex risk measure belonging to the class of spectral risk measures of Acerbi and Tasche [2002]. Formally, a spectral risk measure M_ψ is a weighted average of the the loss distribution quantile q evaluated at p, such that:

$$M_\psi = \int_0^1 \psi(p)\, q_p dp \tag{13}$$

[11] However, Brogan and Stidham (2005, 2008) have shown that for the linear separation property to hold, which assumes convexity of the mean-LPM space, the threshold must either be equal to the risk free rate or the mean of the portfolio.

[12] This is because the following general relationship holds for LPM measures:

$$LPM_{\tau,a}(X) = LPM_{t+C,a}(X + C). \tag{11}$$

which is equivalent to property (D1) presented previously, when accounting for the threshold parameter's shift by the constant C. Additionally, and with important implications in fractional programming, the LPM measure also has the scaling property so that:

$$LPM_{\tau,a}(X) = \frac{1}{b} LPM_{bt,a}(bX), \tag{12}$$

where it is understood that for the non-standardized version of the measure, i.e. when not raised to the power of $\frac{1}{a}$, the measure is multiplied not by $\frac{1}{b}$ but $\frac{1}{b^a}$ instead.

[13] Also called Expected Tail Loss with distinctions in the names sometimes denoting differences for the continuous and sample cases, with the latter requiring a specialized representation in order to be deemed convex according to Rockafellar and Uryasev [2000].

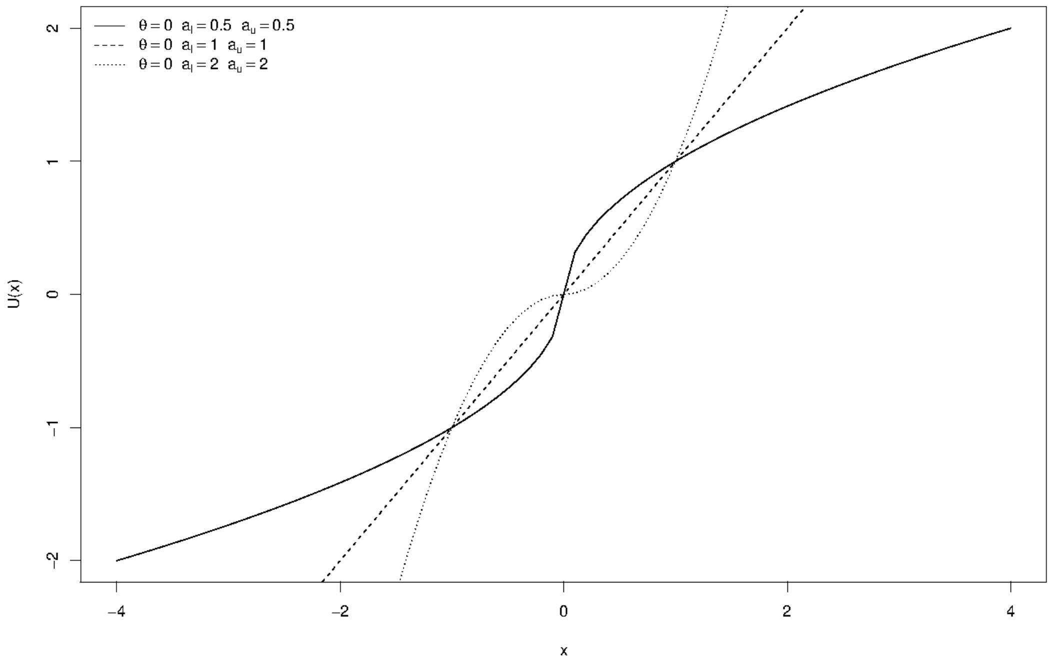

Figure 2: Upper to Lower Partial Moment Utility

where $\psi(p)$ is a weighting function defined over the full range of probabilities $p \in [0,1]$ and restricted to be non-negative, normalized to sum to 1, and increasing or constant in p (such that higher losses have equal or higher weights to lower loses). VaR is clearly a spectral risk measure with weighting the dirac delta function which is degenerate, while CVaR is based on a step function (constant weight for losses greater than VaR). Cotter and Dowd [2006] investigated alternative weighting functions to account for truly risk averse behavior by considering strictly increasing weights functions in an application for establishing futures clearinghouse margin requirements. While they found that such weighting schemes were superior to the standard CVaR, Dowd et al. [2008] also found some problems in their implementation both in the choice of functions as well as the mixing properties of these measures with respect to nonlinear weighting functions. In a different direction Rockafellar et al. [2006] considered the so called mixed-CVaR problem whereby it is possible to mix together CVaR at different coverage rates using a weighting function, and established the relationship between this and the spectral risk representation. In terms of the general optimization problem, CVaR many be represented as an NLP minimization problem with objective function given by:

$$\min_{w,v} \frac{1}{na} \sum_{i=1}^{n} \left[\max \left(0, v - \sum_{j=1}^{m} w_j r_{i,j} \right) \right] - v \tag{14}$$

where v is the a-quantile of the distribution. For a discrete scenario, this can be represented using auxiliary variables as the following LP problem (due to Rockafellar and Uryasev [2000]):

$$\begin{aligned}
&\min_{w,d,v} \frac{1}{na}\sum_{i=1}^{n} d_i + v \\
&s.t. \\
&\sum_{j=1}^{m} w_j r_{i,j} + v \geq -d_i, \forall i \in \{1, ..., n\} \\
&\sum_{j=1}^{m} w_j \mu_j = C \\
&\sum_{j=1}^{m} w_j = 1 \\
&w_j \geq 0, \forall j \in \{1, \ldots, m\} \\
&d_i \geq 0, \forall i \in \{1, \ldots, n\}
\end{aligned} \tag{15}$$

where v represents the VaR at the a-coverage rate and d_i the deviations below the VaR. The formulation presented here is in such a way as to represent the asset returns scenario matrix rather than the more typical loss representation in the literature.
Direct extensions have followed in the same vein as the LPM measures with Biglova et al. [2004] proposing the Rachev Ratio as the upper to lower CVaR for which they provide a mixed integer representation in Stoyanov et al. [2007] (modified by Konno et al. [2011] for cases when the returns are completely distributed on the positive side), and also proposed in the same paper the Generalized Rachev Ratio which is the Rachev Ratio but with the numerator and denominator raised to different powers representing different penalization to gains and losses beyond some upper and lower quantiles. Unfortunately, this generalization, like the upper to lower LPM has both a convex numerator and denominator[14] making it non quasi-convex and hence necessitating a GO approach which will be supported in due course.[15]

3.6 Conditional Drawdown at Risk (CDaR)

Drawdown is an interesting concept in the literature on optimization since this is a strongly path dependent measure. With the exception of Brownian motion with zero drift discussed in Douady et al. [1999], there is no closed form solution for the distribution of this measure. While some interesting solutions have been proposed, see for instance Magdon-Ismail et al. [2004], the measure considered here and its optimization is based on Chekhlov et al. [2005]. The problem

[14] Technically, both risk and reward CVaR functions are convex for values of the power ≥ 1.

[15] The mixed-integer approach for the Rachev Ratio is just as difficult to estimate as it is limited by the size of the scenario which determines the number of binary variables required.

may be posed as the following LP:

$$
\begin{aligned}
& \min_{w,u,v,z} v + \frac{1}{na}\sum_{i=1}^{n} z_i \\
& s.t. \\
& z_i - u_i + v \geqslant 0, \forall i \in \{1,\ldots,n\} \\
& \sum_{j=1}^{m} w_j r_{i,j} + u_i - u_{i-1} \geqslant 0, u_0 = 0, \forall i \in \{1,\ldots,n\} \\
& z_i \geqslant 0, u_i \geqslant 0, \forall i \in \{1,\ldots,n\} \\
& \sum_{j=1}^{m} w_j \mu_j = C \\
& \sum_{j=1}^{m} w_j = 1 \\
& w_j \geqslant 0, \forall j \in \{1,\ldots,m\}
\end{aligned} \tag{16}
$$

where z is an auxiliary vector of variables of the conditional drawdowns, u the auxiliary vector of variables to model the cumulative returns and v represents the Drawdown at Risk at the quantile level a. In the **parma** package, this problem is represented only in LP form as of the current release, allowing for both minimization subject to a minimum return constraint or as an optimal risk-return fractional LP problem. In addition, providing the type of scenario which makes sense in this context (i.e. multi-period ahead) is completely left to the users' own devices, as it remains a mystery to this author how to fully optimize multi-period uncertainty in a single scenario and single-period SP setup.

3.7 Comparison of Measures

The *riskfun* function in the **parma** package provides a helpful utility to investigate the properties of the risk/deviation measures discussed thus far. The following table, taken from the first test example in the *parma.tests* folder of the package presents some insights into the properties of the risk measures: The properties tested are defined as follows:

Table 2: Properties of Risk/Deviation Measures.

	MAD	V	SD	$MiniMax$	$CVaR$	$CDaR$	$LPM_{m,\tau=c}$	$LPM_{m,\tau=\mu}$
Scaling	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
Location^1	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE
Location^2	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE
Subadditivity	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE

Note: The Table presents some of the properties of the measures used in the **parma** package, where the definitions of the properties are defined in Section 3.7.

- Scaling $f(bX) = \frac{1}{b}f(X)$
- *Location*^{1} $f(a+X) = f(X)$
- *Location*^{2} $f(a+X) = f(X) + a$
- Subadditivity $f(X_1) + f(X_2) \geq f(X_1 + X_2)$

where f is some measure, b a positive scalar and a some constant $\in \mathbb{R}$. The scaling property is shared by all measures, being a feature of their underlying constituent functions, and a requirement for using fractional programming. The fact that MAD, V, and SD are location invariant ($Location^1$) is not surprising since they are deviation measures, which means that they are calculated after centering. It is no surprise either that variance (V) is not subadditive since the square function is known to be superadditive, whilst standard deviation (SD) is subadditive. This has certain implications for the fractional programming problem which leads to an optimal Sharpe ratio, even though it is the variance which is minimized. The next 3 measures, $CVaR$, $CDaR$, and LPM are not deviation measures and as such are not location invariant, but do have the location property ($Location^2$), with the exception of $CDaR$ which is path dependent. While the location and scaling of the LPM measure was discussed in Section 3.4, it is interesting to note that subadditivity is only present when the threshold is equal to the portfolio mean[16], something also discussed in Brogan and Stidham [2005].

4 General Problem Formulation

As was summarized in Table 1, the **parma** package supports a variety of solvers, depending on the type of objective and constraints. This section briefly outlines the general MILP, QP, NLP and GNLP formulations used.

4.1 MILP

The MILP problem may be very generally represented as:

$$\begin{aligned}
&\min_{\mathbf{w},\{\}} \mathbf{Sw} \\
&s.t. \\
&\ldots \\
&\mathbf{U} \leq \mathbf{Aw} \leq \mathbf{L} \\
&\mathbf{Cw} = \mathbf{b} \\
&\mathbf{w}'\mathbf{1} = budget \\
&\delta_i w_i^{lower} \leq w_i \leq \delta_i w_i^{upper} \\
&\delta'\mathbf{1} = \#assets \\
&w_i^{lower} \leq w_i \leq w_i^{upper}
\end{aligned} \tag{17}$$

given a vector of weights $\mathbf{w}$ of length m, where $\{\}$ denote any additional parameters passed to $(\ldots)$ additional problem specific constraints. The k inequality constraints are stacked in a $k \times m$ $\mathbf{A}$ matrix with lower and upper bounds of length k given by $\mathbf{L}$ and $\mathbf{U}$ respectively. The general equality constraints are stacked in the $l \times m$ $\mathbf{C}$ matrix with bounds of length l given by $\mathbf{b}$, where the budget constraint is represented separately for clarity. Custom inequality and equality matrices can be passed in the **parmaspec** function via the *ineq.mat* and *eq.mat* arguments with their corresponding bounds given by (*ineq.LB*, *ineq.UB*) and (*eqB*) respectively. Optionally, a cardinality constraint based on an 'in-between or out' formulation is represented by use of m binary variables δ. The use of cardinality constraints is only allowed when the *targetType* argument in the **parmaspec** is 'minrisk' since the use of the fractional programming 'optimal' option and the 'in-between or out' cardinality constraint formulation makes the problem no

[16] Equivalent to setting the threshold to 999 in the *parmaspec* options

longer LP (because of the fractional multiplier). Additionally, it is possible to pass a benchmark series which is subtracted from the weighted returns for benchmark relative optimization. In that case, it is usual to set the upper and lower bounds to some positive and negative values representing the allowable deviations from the benchmark weights, the budget to zero, so that the resulting weights represent active bets on the benchmark, and the forecast and target return as active values (i.e. in excess of the benchmark). Finally, and currently only supported by the MILP type problems[17], a vector of probabilities may also be passed (which must sum to 1), giving different weights to each row of the scenario.

4.2 NLP

The NLP problem may be very generally represented as:

$$
\begin{aligned}
&\min_{\mathbf{w}} f(\mathbf{w}, \ldots) \\
&s.t. \\
&g(\mathbf{w}, \ldots) \leq \mathbf{0} \\
&h(\mathbf{w}, \ldots) = \mathbf{0} \\
&\left|\mathbf{w}'\right| \mathbf{1} = c \\
&w_i^{lower} \leq w_i \leq w_i^{upper}
\end{aligned}
\tag{18}
$$

where $g(\mathbf{w}, \ldots)$ is the convex inequality function returning a vector of length equal to the actual inequalities evaluated, and $h(\mathbf{w}, \ldots)$ the affine equality function returning a vector of length l of the actual equalities evaluated. In the **parma** package, custom inequalities and equalities can be passed in the **parmaspec** function as list of functions (via the *ineqfun* and *eqfun* arguments) but the user must also pass their equivalent jacobian functions (via the *ineqgrad* and *eqgrad* arguments). The absolute sum of weights may be used to control leverage (c) when the weights are allowed to take on negative values, else this translates to a simply sum of weights when the weights are all positive. As in the MILP problem, benchmark relative optimization is possible (see the previous section for details).

4.3 QP

The quadratic formulation, for use in the EV type problem of Markowitz [1952] may be represented as:

$$
\begin{aligned}
&\min_{\mathbf{w}} \mathbf{w}'\mathbf{S}\mathbf{w} \\
&s.t. \\
&\mathbf{U} \leq \mathbf{A}\mathbf{w} \leq \mathbf{L} \\
&\mathbf{C}\mathbf{w} = \mathbf{b} \\
&w_i^{lower} \leq w_i \leq w_i^{upper}
\end{aligned}
\tag{19}
$$

where $\mathbf{S}$ is some positive definite covariance matrix, and constraints as in the LP case in Equation (17). Benchmark relative optimization is fully supported, and in this case a vector of the covariance between the benchmark and the portfolio must be passed (via the *benchmarkS* option in the *parmaspec* function), where the first value is the variance of the benchmark, so that the resulting joint covariance matrix constructed by the routine is:

$$
\Sigma_{b,r} = \begin{pmatrix} \sigma_b^2 & \sigma_{\{b,r\}} \\ \sigma_{\{b,r\}} & \Sigma_r \end{pmatrix}
\tag{20}
$$

[17] This may be extended in future to the NLP formulation.

and the portfolio relative risk is then given by $(-1, w)'\Sigma_{b,r}(-1, w)$. The rest of the values should be passed as explained in the MILP section. The interested reader should consult for example Stoyanov et al. [2007] pp.412–414 for details of benchmark relative optimization in a QP setup.

4.3.1 Optimal Portfolio

The optimal portfolios admit one of 2 equivalent representations, depending on whether we are maximizing reward to risk:

$$\begin{aligned}
&\max_{\mathbf{x},t}\ \mathbf{x}'f_r \\
&\text{s.t. } (-t,\mathbf{x})'\Sigma_{b,r}(-t,\mathbf{x}) \leqslant 1 \\
&\mathbf{x}'\mathbf{1} = t \\
&t\mathbf{L} \leqslant \mathbf{A}\mathbf{x} \leqslant t\mathbf{U} \\
&t \geqslant 0
\end{aligned} \tag{21}$$

or minimizing risk to reward:

$$\begin{aligned}
&\min_{\mathbf{x},t}\ (-t,\mathbf{x})'\Sigma_{b,r}(-t,\mathbf{x}) \\
&\text{s.t. } \mathbf{x}'f_r = 1 \\
&\mathbf{x}'\mathbf{1} = t \\
&t\mathbf{L} \leqslant \mathbf{A}\mathbf{x} \leqslant t\mathbf{U} \\
&t \geqslant 0
\end{aligned} \tag{22}$$

where it is understood that f_r is the returns forecast, and in the case of benchmark relative optimization, the returns forecast in excess of the benchmark forecast.

4.4 SOCP

A Second Order Cone Programming (*SOCP*) problem has the following form:

$$\begin{aligned}
&\text{minimize } c'x \\
&\text{subject to } \|A_i x + b_i\| \leq c'_i x + d_i,\ \text{i} = 1,\ldots,\text{L}
\end{aligned} \tag{23}$$

where $\|...\|$ denotes the euclidean norm so that $\|x\| = \sqrt{x'x}$. Special cases include linear, quadratic and quadratically constrained quadratic problems, as well as a number of nonlinear and possible non-differentiable problems. While SOCP is a special case of Semi-Definite programming (*SDP*) it is always more efficient to solve these problems using special purpose solvers rather than more general ones.

A special set of constraints based on p-norms can be represented using SOCP. Consider the following p-norm inequality constraint:

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p} \leq t \tag{24}$$

which involves the p-norm of a vector $\mathrm{x} \in \Re^n$. We can re-write p as l/m, where $l \geq m$, so that:

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^{l/m}\right)^{m/l} \leq t \tag{25}$$

and is equivalent to the following set of constraints:

$$\begin{aligned}
&-x_i + t^{\frac{l-m}{l}} s_i^{\frac{m}{l}} \geq 0 \\
&x_i + t^{\frac{l-m}{l}} s_i^{\frac{m}{l}} \geq 0 \\
&s_i \geq 0, i = 1, ..., n \\
&\left(\sum_{i=1}^{n} s_i \right) \leq t, t \geq 0
\end{aligned} \tag{26}$$

where s is an auxilliary vector of variables of same size as the original decision vector x. Obviously, the inequality can be turned into an equality using one extra constraint. However, care should be taken to allow for some slack between the upper and lower bounds of the 2 inequality constraints used to represent one equality else numerical difficulties may be encountered by the solver. It thus follows from the above type of exposition that it is quite easy to include a leverage constraint in a long/short portfolio optimization setting by setting $l = m = 1$ which represents the manhattan norm.

In the parma package, problems which include the covariance matrix instead of a scenario may be solved either by the QP solver else the SOCP solver. In the latter case, there is a greater deal of flexibility since QCQP problems are easily solved (see the option for a list of matrices Q in the documentation on **parmaspec**), as is the case of long/short optimization with a leverage (gross sum) constraint and the optimal risk to reward problem using the fractional approach of Section 4.3.1

4.5 GNLP

The GNLP problem in the **parma** package is represented by use of derivative free penalty functions as:

$$\begin{aligned}
&\min_{\mathbf{w}} f\left(\mathbf{w}, \ldots\right) + p \sum_{i=1}^{k} \max\left(g_i\left(\mathbf{w}, \ldots\right), 0\right)^2 & (27) \\
&\qquad + p \sum_{i=1}^{l} h_i(\mathbf{w}, \ldots)^2 + p \max\left(0, \sum_{i=1}^{m} I_{|w_i| > 0.001} - \#assets\right)^2 \\
&\qquad s.t. & (28) \\
&\qquad w_i^{lower} \leq w_i \leq w_i^{upper}
\end{aligned}$$

where p is a penalty parameter. No gradients are used in this case which means that any user specified equality and inequality functions can be passed (without Jacobians). All NLP problems can be solved by GNLP, but mileage will vary with regards to the optimality of the solution.

5 Optimal Portfolios

Consider the general nonlinear problem of minimizing a risk to reward problem represented as a fraction:

$$\begin{aligned}
&\min_{\mathbf{w}} \frac{\rho_{risk}\left(\mathbf{Rw}\right)}{\rho_{reward}\left(\mathbf{Rw}\right)} \\
&\mathbf{w}'\mathbf{1} = 1 \\
&\mathbf{L} \leq \mathbf{Aw} \leq \mathbf{U} \\
&\mathbf{w} \geq 0
\end{aligned} \tag{29}$$

where $\mathbf{w}$ is an $m \times 1$ vector of weights, $\mathbf{R}$ the $n \times m$ Scenario matrix of returns so that the risk (ρ_{risk}) and reward (ρ_{reward}) functions are applied on the weighted scenario returns, $\mathbf{1}$ an $m \times 1$ vector ones and $\mathbf{A}$ a $q \times m$ matrix of linear constraints with lower and upper bounds given by $\mathbf{L}$ and $\mathbf{U}$ respectively. The key developments in the theory of fractional programming were provided in the linear case by Charnes and Cooper [1962], while for nonlinear cases the main contributions can be traced to Dinkelbach [1967] and Schaible(1976a, 1976b). More recently, Stoyanov et al. [2007] provided a more focused review of fractional programming with reference to financial portfolio optimization. Under the assumption that both numerator and denominator are positive homogeneous, the problem in (29) can be transformed into the following simpler nonlinear fractional programming (*NLFP*) problem:

$$
\begin{aligned}
&\min_{\hat{\mathbf{w}},\upsilon} \rho_{risk}\left(\mathbf{R}\hat{\mathbf{w}}\right) \\
&\rho_{reward}\left(\mathbf{R}\hat{\mathbf{w}}\right) \geq 1 \\
&\hat{\mathbf{w}}'\mathbf{1} = \upsilon \\
&\upsilon\mathbf{L} \leq \mathbf{A}\hat{\mathbf{w}} \leq \upsilon\mathbf{U} \\
&\upsilon > 0
\end{aligned}
\tag{30}
$$

where υ represents a scalar auxiliary scaling variable and $\hat{\mathbf{w}}$ the unconstrained optimal weight vector such that the optimal weight vector $\mathbf{w} = \frac{\hat{\mathbf{w}}}{\upsilon}$. In order for this problem to be convex, the reward function must be concave and the risk function convex, with strict positivity required for both functions.[18] Different relaxations of these basic conditions lead to different classes of problems in the literature, some with unique solutions and others requiring global search methods for solution. These simple conditions admit both convex risk and deviation measures as defined in Section 3. The **parma** package implements LP, QP and NLP based fractional optimization for all measures so far discussed, including benchmark relative problems, and in the case of the NLP formulation the analytical gradient of the functions and jacobian of the constraints have also been derived and used.[19]

6 Smooth Approximations to Non-Continuous Functions

While it is preferable to work with an LP formulation of a decision problem, there are certain situations where this poses certain challenges. First, for some LP problems, the dimension of the dataset and constraints may tax the limits of computer memory. Consider for example the MAD model presented in Section 3 which has a constraint matrix of size $2n \times m$ in order to create the piecewise linear representation for the absolute value, where for large scenarios (n) and assets (m) memory considerations become important. Second, in practise, many problems and/or constraints simply cannot be expressed in LP form necessitating the use of either QP or NLP based methods. In that case, it is always preferable to have analytic derivatives of the function and constraints, for speed and accuracy versus numerically evaluation methods. Interestingly, some problems, while convex are discontinuous because of the presence of such functions as make use of the *minimum* or *absolute* values. For these problems, an approximation may be obtained by considering smooth and continuous versions of these functions. Consider for example the CVaR and LPM measures, both of which depend on the max function, for which the following

[18] For the reward function the requirement is a little more relaxed in that there must be at least some combination of the weights and returns for which the reward is positive. Additionally, for a linear reward function the constraint becomes an equality.

[19] This excludes the case of cardinality constraints which make the problems non convex

smooth approximation, s_{max} may be used:

$$\max(x, 0) \approx s_{\max}(x, 0) = \frac{\left(\sqrt{x^2 + \varepsilon} + x\right)}{2} \tag{31}$$

where ε is some very small positive number controlling the degree of approximation error. The absolute value may also be approximated with the following function s_{abs}:

$$abs(x) \approx s_{abs}(x) = \sqrt{(x + \varepsilon)^2} \tag{32}$$

although alternatives also exist[20]. Apart from allowing the MAD problem to be represented in NLP form with a smooth function, it also allows for the use of short positions, replacing the full investment constraint with a leverage constraint (the absolute sum of positions)[21], without resorting to such methods as described in Jacobs et al. [2006] which double the size of the problem and require certain very specific assumptions about the 'trimability' of the portfolio. Finally, for the case of the Minimax problem, it is possible to make use of the generalized mean function $M_p(x_1, \ldots, x_n) = \left(\frac{1}{n}\sum_{i=1}^{n} x_i^p\right)^{1/p}$, which approximates the maximum of a set of positive values as $p \to \infty$. In order to obtain the maximum loss for use in the NLP minimax optimization function, this function is combined with the s_{max} function defined in (31) applied to the negative of the scenario returns:

$$\left(\frac{1}{n}\sum_{i=1}^{n} s_{\max}\left(-\mathbf{w}'\mathbf{r}_i, 0\right)^p\right)^{1/p}. \tag{33}$$

In practice, because the optimization problem needs to be calibrated for p, making this a very hard problem, the **parma** package instead represents the NLP objective in its LP form which leads to very high accuracy. Table 3 shows the relative accuracy of the NLP representation of the problems versus the exact LP formulation for a typical minimization problem, while Table 4 shows the relative accuracy in a fractional problem setting.

Table 3: LP vs NLP Smooth Approximations (minrisk)

	MAD	**MiniMax**	**CVaR**	**EV**	**LPM[1]**
MSE[weights]	4.18E-13	3.66E-31	4.53E-16	3.71E-18	2.10E-17
MAE[weights]	2.94E-07	2.81E-16	1.34E-08	1.03E-09	2.53E-09
MaxE[weights]	2.20E-06	1.78E-15	4.74E-08	5.63E-09	1.23E-08
AbsErr[risk]	1.37E-11	6.94E-17	2.04E-12	1.53E-08	1.85E-14

Note: The Table reports the mean squared error (MSE), mean absolute error (MAE) and maximum error of the weights optimized under the NLP smooth approximation representation versus the exact LP formulation. The absolute error (AbsErr) in the optimized risk is also shown. The problem formulation was based on parma.test2 in the parma.tests folder of the **parma** package, using the ETF dataset with the objective of minimizing risk given an equality for the target return.

7 Custom Constraints

Since versions 1.5-0 the package has exported a number of custom constraint functions and their analytic derivatives (jacobians) to be used with NLP formulations. At present, there are a set of functions for defining turnover constraints and a maximum portfolio variance given a user supplied covariance matrix. These functions can be passed to *parmaspec* quite easily and details are provided in the documentation. This section will instead provide for a brief description of the type of turnover constraints which are included.

[20] One such alternative is: $(2x/\pi)\left(\tan^{-1}(ox)\right)$, where o is some very large positive number.

[21] A common mistake is to keep the full investment constraint instead of replacing it with the leverage constraint, which makes no sense even when controlling for individual position limits.

Table 4: LP vs NLP Smooth Approximations (fractional)

	MAD	MiniMax	CVaR	EV	LPM[1]
MSE[weights]	5.35E-10	1.28E-35	1.34E-23	8.89E-19	1.25E-24
MAE[weights]	1.27E-05	9.51E-19	1.87E-12	4.57E-10	5.37E-13
MaxE[weights]	5.93E-05	1.39E-17	1.02E-11	2.71E-09	3.20E-12
Err[risk]	1.05E-07	6.94E-18	2.62E-12	1.91E-08	4.66E-15

Note: The Table reports the mean squared error (MSE), mean absolute error (MAE) and maximum error of the weights optimized under the NLP smooth approximation representation versus the exact LP formulation. The absolute error (AbsErr) in the optimized risk is also shown. The problem formulation was based on parma.test3 in the parma.tests folder of the **parma** package, using the ETF dataset with the fractional objective of minimizing risk/reward.

7.1 Simple Turnover

The simple turnover (T) constraint, given the set of optimal decision weights ($\mathbf{w}$) versus the existing set of weights ($\mathbf{w}^{old}$) can be represented as:

$$\sum_{i=1}^{m} \left| w_i - w_i^{old} \right| \leqslant T, T \in \mathbb{R}^+ \tag{34}$$

Because of the absolute value function, this problem is most easily represented in an NLP setup with the use of the smooth absolute value function presented in Section 6. For LP problems, this may also be formulated by use of auxiliary variables and this may be included in the package at a future time.

7.2 Buy and Sell Turnover

A more flexible turnover constraint limits the buy (T^+) and sell (T^-) turnover separately, and can be represented as:

$$\begin{aligned} &\sum_{i=1}^{m} \max\left(0, w_i - w_i^{old}\right) \leqslant T^+, T^+ \in \mathbb{R}^+ \\ &\sum_{i=1}^{m} \max\left(0, w_i^{old} - w_i\right) \leqslant T^-, T^- \in \mathbb{R}^+ \end{aligned} \tag{35}$$

where again the smooth approximation to the maximum value function presented in Section 6 is used. **When using this constraint in a fractional programming setup, care should be taken that the combination of bounds, turnover limits and the forecast return vector do not result in a negative expected return in which case the problem is not solvable**.

8 FAQ's and Misc Notes

The following additional notes may be of interest:

- QCQP (Quadratically Constrained Quadratic Problems) are not yet implemented but may be in due course.
- Custom equality constraints in NLP problems MUST be affine in order to guarantee convexity of the problem.
- Custom inequality constraints in NLP problems MUST be convex in order to guarantee convexity of the problem.

- No plots yet...may come in due course.
- The parma.tests folder (under the *inst* folder in the source) contains a large number of instructive examples.

If you have questions, use the R-SIG-FINANCE mailing list to ask them.

References

C. Acerbi. Spectral measures of risk: a coherent representation of subjective risk aversion. *Journal of Banking & Finance*, 26(7):1505–1518, 2002.

C. Acerbi and D. Tasche. Expected shortfall: A natural coherent alternative to value at risk. *Economic Notes*, 31(2):379–388, 2002.

K.J. Arrow. Uncertainty and the welfare economics of medical care. *The American economic review*, 53(5):941–973, 1963.

P. Artzner, F. Delbaen, J.M. Eber, and D. Heath. Coherent measures of risk. *Mathematical finance*, 9(3):203–228, 1999.

V.S. Bawa. Safety-first, stochastic dominance, and optimal portfolio choice. *Journal of Financial and Quantitative Analysis*, pages 255–271, 1978.

V.S. Bawa and E.B. Lindenberg. Capital market equilibrium in a mean-lower partial moment framework. *Journal of Financial Economics*, 5(2):189–200, 1977.

EML Beale. On minimizing a convex function subject to linear inequalities. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 173–184, 1955.

D. Bernoulli. Exposition of a new theory on the measurement of risk (originally published 1738 st. petersburg, russia). *Econometrica*, 22(1):23–36, 1954.

A. Biglova, S. Ortobelli, S.T. Rachev, and S. Stoyanov. Different approaches to risk estimation in portfolio theory. *The Journal of Portfolio Management*, 31(1):103–112, 2004.

F. Black and R. Litterman. Global portfolio optimization. *Financial Analysts Journal*, 48(5): 28–43, 1992.

A.J. Brogan and S. Stidham. A note on separation in mean-lower-partial-moment portfolio optimization with fixed and moving targets. *IIE Transactions*, 37(10):901–906, 2005.

A.J. Brogan and S. Stidham. Non-separation in the mean–lower-partial-moment portfolio optimization problem. *European Journal of Operational Research*, 184(2):701–710, 2008.

A. Charnes and W.W. Cooper. Chance-constrained programming. *Management science*, pages 73–79, 1959.

A. Charnes and W.W. Cooper. Programming with linear fractional functionals. *Naval Research logistics quarterly*, 9(3-4):181–186, 1962.

A. Chekhlov, S. Uryasev, and M. Zabarankin. Drawdown measure in portfolio optimization. *International Journal of Theoretical and Applied Finance*, 8(1):13–58, 2005.

J. Cotter and K. Dowd. Extreme spectral risk measures: an application to futures clearinghouse margin requirements. *Journal of Banking & Finance*, 30(12):3469–3485, 2006.

G.B. Dantzig. Recent advances in linear programming. *Management Science*, 2(2):131–144, 1956.

G.B. Dantzig and G. Infanger. Large-scale stochastic linear programs: Importance sampling and benders decomposition. In *IMACS'91 (July 22-26)*, volume 1991, page 111. North Holland, 1992.

G.B. Dantzig and G. Infanger. Multi-stage stochastic linear programs for portfolio optimization. *Annals of Operations Research*, 45(1):59–76, 1993.

W. Dinkelbach. On nonlinear fractional programming. *Management Science*, 13(7):492–498, 1967.

R. Douady, M. Yor, and A.N. Shiryaev. On probability characteristics of downfalls in a standard brownian motion. *Teoriya Veroyatnostei i ee Primeneniya*, 44(1):3–13, 1999.

K. Dowd, J. Cotter, and G. Sorwar. Spectral risk measures: properties and limitations. *Journal of Financial Services Research*, 34(1):61–75, 2008.

Charles D. Feinstein and Mukund N. Thapa. A reformulation of a mean-absolute deviation portfolio optimization model. *Management Science*, 39(12):pp. 1552–1553, 1993. ISSN 00251909. URL `http://0-www.jstor.org.wam.city.ac.uk/stable/2633071`.

P.C. Fishburn. Mean-risk analysis with risk associated with below-target returns. *The American Economic Review*, pages 116–126, 1977.

G30. *Derivatives: practices and principles*, volume 5. Group of Thirty, 1993. URL `http://www.group30.org/rpt_29.shtml`.

G. Hanoch and H. Levy. The efficiency analysis of choices involving risk. *The Review of Economic Studies*, 36(3):335–346, 1969.

Nikolaus Hansen. The cma evolution strategy: A comparing review. In Jose Lozano, Pedro Larranaga, Inake Inza, and Endika Bengoetxea, editors, *Towards a New Evolutionary Computation*, volume 192 of *Studies in Fuzziness and Soft Computing*, pages 75–102. Springer Berlin / Heidelberg, 2006. ISBN 978-3-540-29006-3. URL `http://dx.doi.org/10.1007/3-540-32494-1_4`.

W.V. Harlow and R.K. Rao. Asset pricing in a generalized mean-lower partial moment framework: Theory and evidence. *Journal of Financial and Quantitative Analysis*, pages 285–311, 1989.

D.M. Holthausen. A risk-return model with risk and return measured as deviations from a target return. *The American Economic Review*, 71(1):182–188, 1981.

Bruce I. Jacobs, Kenneth N. Levy, and Harry M. Markowitz. Trimability and fast optimization of long-short portfolios. *Financial Analysts Journal*, 62(2):36–46, 2006. ISSN 0015198X. URL `http://0-www.jstor.org.wam.city.ac.uk/stable/4480743`.

W. James and C. Stein. Estimation with quadratic loss. In *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, volume 4, page 361. University of California Press, 1956.

H. Konno and H. Yamazaki. Mean-absolute deviation portfolio optimization model and its applications to tokyo stock market. *Management science*, pages 519–531, 1991.

H. Konno, H. Shirakawa, and H. Yamazaki. A mean-absolute deviation-skewness portfolio optimization model. *Annals of Operations Research*, 45(1):205–220, 1993.

H. Konno, K. Tanaka, and R. Yamamoto. Construction of a portfolio with shorter downside tail and longer upside tail. *Computational Optimization and Applications*, 48(2):199–212, 2011.

R. Kouwenberg and S.A. Zenios. Stochastic programming models for asset liability management. *Handbook of asset and liability management*, 1:253–303, 2006.

A. Madansky. Methods of solution of linear programs under uncertainty. *Operations Research*, pages 463–471, 1962.

M. Magdon-Ismail, A.F. Atiya, A. Pratap, and Y.S. Abu-Mostafa. On the maximum drawdown of a brownian motion. *Journal of applied probability*, 41(1):147–161, 2004.

H. Markowitz. Portfolio selection. *Journal of finance*, pages 77–91, 1952.

R.O. Michaud. The markowitz optimization enigma: Is optimized optimal? *Financial Analysts Journal*, 45(1):31–42, 1989.

D. Nawrocki and K. Staples. A customized lpm risk measure for portfolio analysis. *Applied economics*, 21(2):205–218, 1989.

J. Quiggin. A theory of anticipated utility. *Journal of Economic Behavior & Organization*, 3 (4):323–343, 1982.

R.T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of risk*, 2: 21–42, 2000.

R.T. Rockafellar, S. Uryasev, and M. Zabarankin. Generalized deviations in risk analysis. *Finance and Stochastics*, 10(1):51–74, 2006.

A.D. Roy. Safety first and the holding of assets. *Econometrica: Journal of the Econometric Society*, 20:431–449, 1952.

L.J. Savage. Subjective probability and statistical practice. *The foundations of statistical inference*, pages 9–35, 1962.

S. Schaible. Fractional programming. i, duality. *Management Science*, 22(8):858–867, 1976a.

S. Schaible. Fractional programming. ii, on dinkelbach's algorithm. *Management Science*, 22(8): 868–873, 1976b.

D. Schmeidler. Subjective probability and expected utility without additivity. *Econometrica*, 57 (3):571–87, 1989.

M.G. Speranza. Linear programming models for portfolio optimization. *Finance*, 14(1):107–123, 1993.

B.K. Stone. A general class of three-parameter risk measures. *The Journal of Finance*, 28(3): 675–685, 1973.

S.V. Stoyanov, S.T. Rachev, and F.J. Fabozzi. Optimal financial portfolios. *Applied Mathematical Finance*, 14(5):401–436, 2007.

J. Von Neumann and O. Morgenstern. *Theory of games and economic behavior.* Princeton University Press, 1944.

M.R. Young. A minimax portfolio selection rule with linear programming solution. *Management science*, pages 673–683, 1998.

L.A. Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.