

Current time: 2015-11-03 00:24:29

Application: Portfolio Dashboard

(c) Rmetrics 2011-2015

Choose Application

Access Data Base

Run Analytics

Plot Charts

Print Reports

Search Help

About

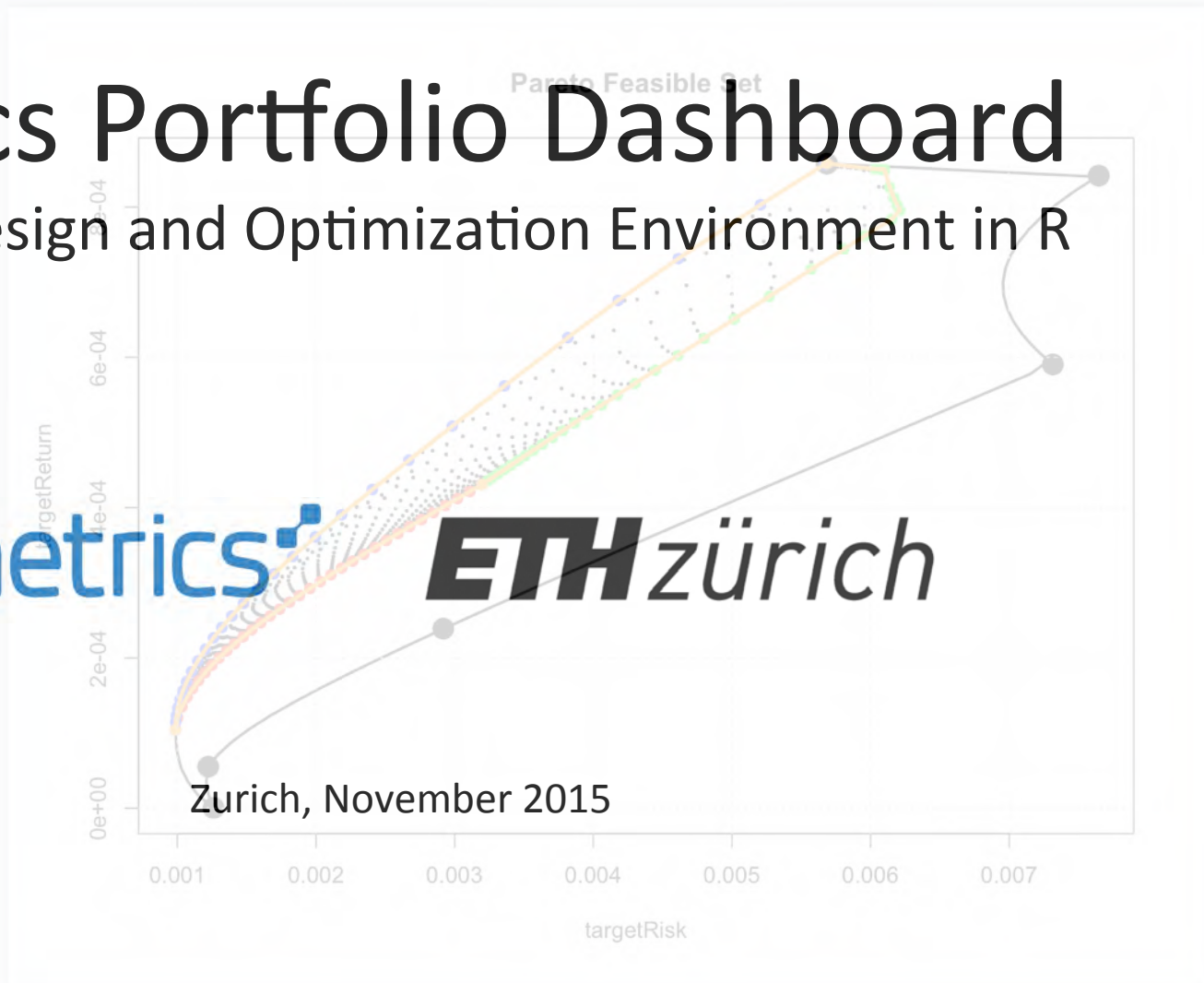
This is the Plot Charts Application

Choose Chart:

- Pareto Surface

# Rmetrics Portfolio Dashboard

## A Portfolio Design and Optimization Environment in R



**ETH** zürich

Zurich, November 2015

# Introduction

This presentation describes elements for a multi-objective portfolio design and optimization environment.

The environment is build on top of

- the statistical environment *R/Rmetrics*, *RStudio*, and *RShiny* using
- state-of-the-art mathematical programming solvers which are
- linked through the R/AMPL interface.

The specifications arise from previous dashboard developments and a pre-study on *Portfolio Goal Programming*. This study has investigated

- the technical feasibility of the *Portfolio Dashboard*
- the interplay of individual *Components and Modules*, and
- the practical *Realization* for such an environment.

# Introduction

The presentation is divided into two parts:

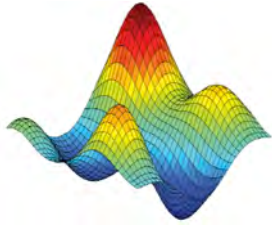
- the 1<sup>st</sup> Part describes the theoretical concepts behind the *Portfolio Dashboard*,
- the 2<sup>nd</sup> Part describes the design concepts and the technical specifications of the *Software Pilot*.

The *Software Pilot* represents the entry level of the Rmetrics Portfolio Dashboard. It will become ready for a complete mathematical and statistical portfolio design environment.

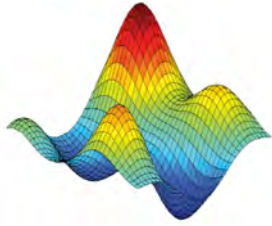
The *Portfolio Dashboard* is designed to outperform the traditional portfolio optimization tools provided by Matlab, Tomlab, and Mathworks.

In the field of *Multi-Objective Portfolio Optimization* the Portfolio Dashboard will set completely new standards.

# Part 1: Feasibility Portfolio Dashboard



# 1 Design Concepts



# Portfolio Designs

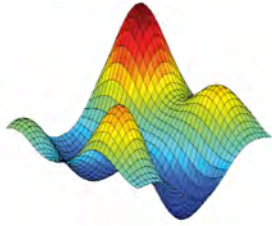
Classification:

In the Portfolio Dashboard we have implemented Rmetrics' traditional portfolio design tools, including

- 1 *Single Portfolio Designs* and their optimization
- 2 *portfolio Designs Along the Efficient Frontier* or any other kind of critical line, and
- 3 *portfolio Designs on a Grid* across feasible sets

However, the major Rmetrics' development is concerned with

- 4 *Multi-Objective Portfolio Designs* across the Pareto feasible set

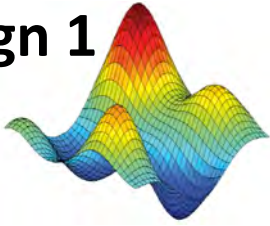


# Theoretical Concepts

Our solution proposes a mathematical programming approach in portfolio design allowing for multi-objective functions expressed by performance measures, risk measures, and/or any other kind of *diversification, parity budgeting, tail risk dependence, or stability* measures.

New tools and functionalities have been added including *non-convex* and *global* portfolio designs. The solvers in use, either *Open Source* or *Commercial Software*, cover a broad range from linear mathematical programming, via quadratic, and non-linear to non-convex, Genetic Algorithms and Simulated Annealing types of solvers.

## Design 1

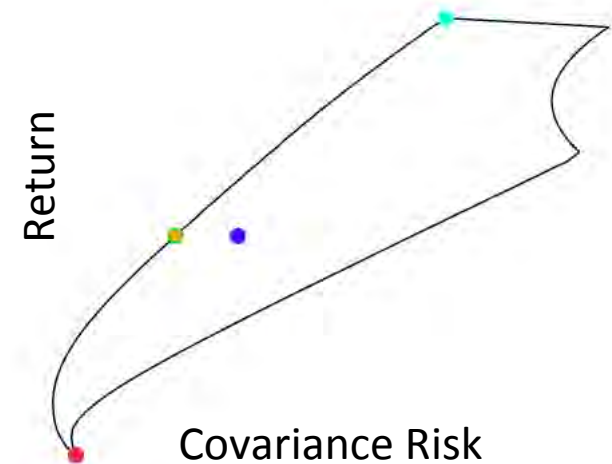


# Single Portfolio Designs

belong to the most simple portfolios at all.

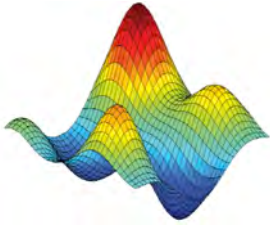
- Usually they allow to minimize the risk subject to a given target return (or vice versa).
- The risk part may be described
  - by the (robustified) covariance risk of Markowitz,
  - by the expected shortfall risk,
  - by a drawdown-at-risk measure,
  - by the Copula tail dependence risk,
  - by non-linear and/or non-quadratic more general risk measures.

Markowitz Single Portfolio  
Swiss Pension Fund Portfolio

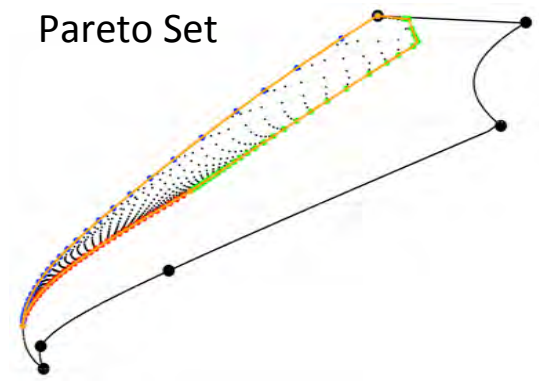
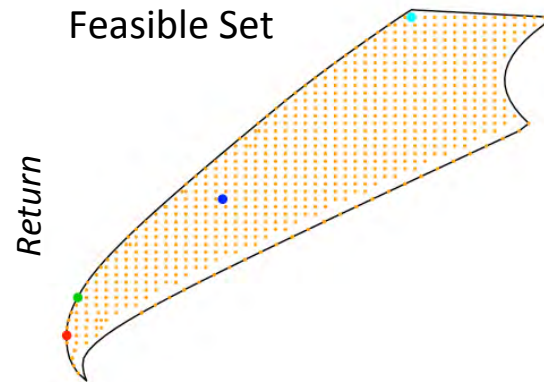
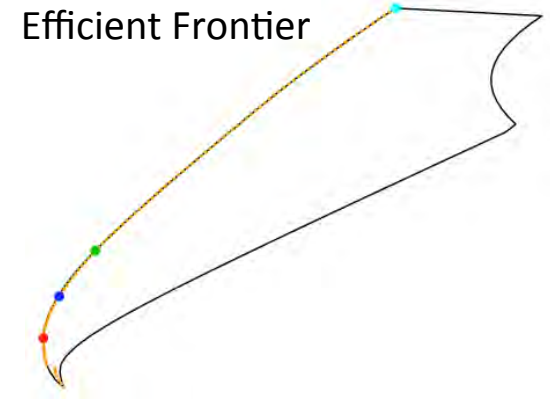
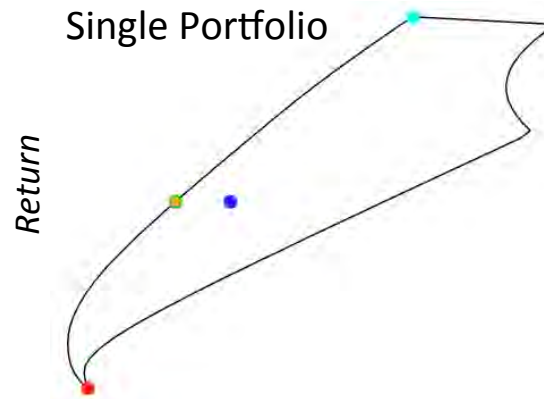


The optimal portfolio (orange) and three reference portfolio min/max Return and Equal Weights





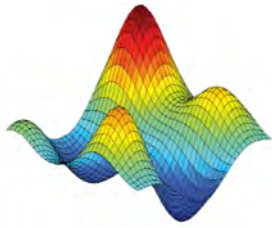
# Reference Designs



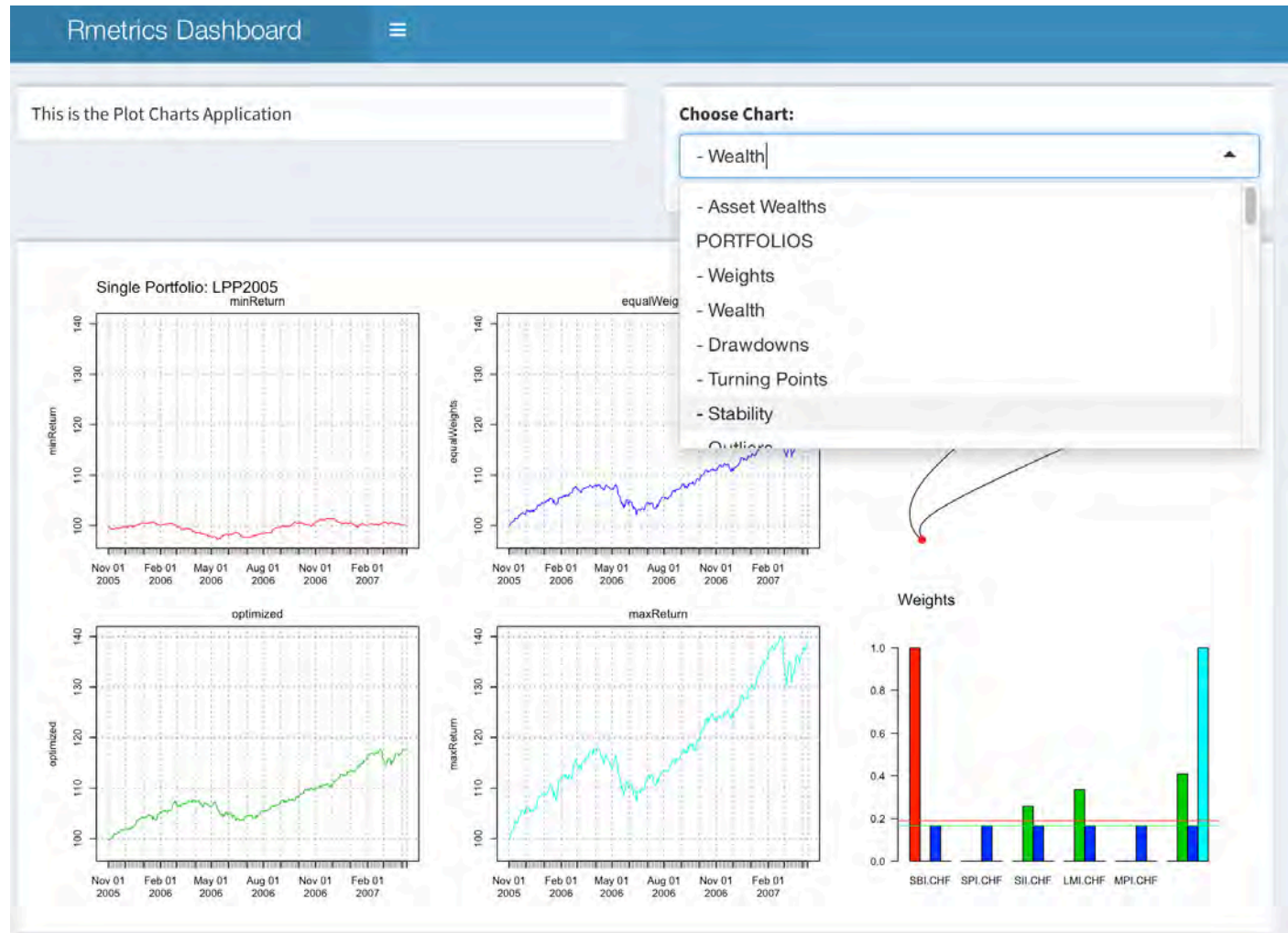
Each Single or Multiple Portfolio Design will be optimized in the context of the efficient frontier, the Hull, and the feasible sets.

*Covariance Risk*

*Covariance Risk*



# Single Portfolio Designs

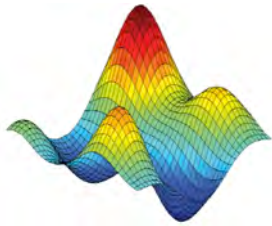


Basic Charts:

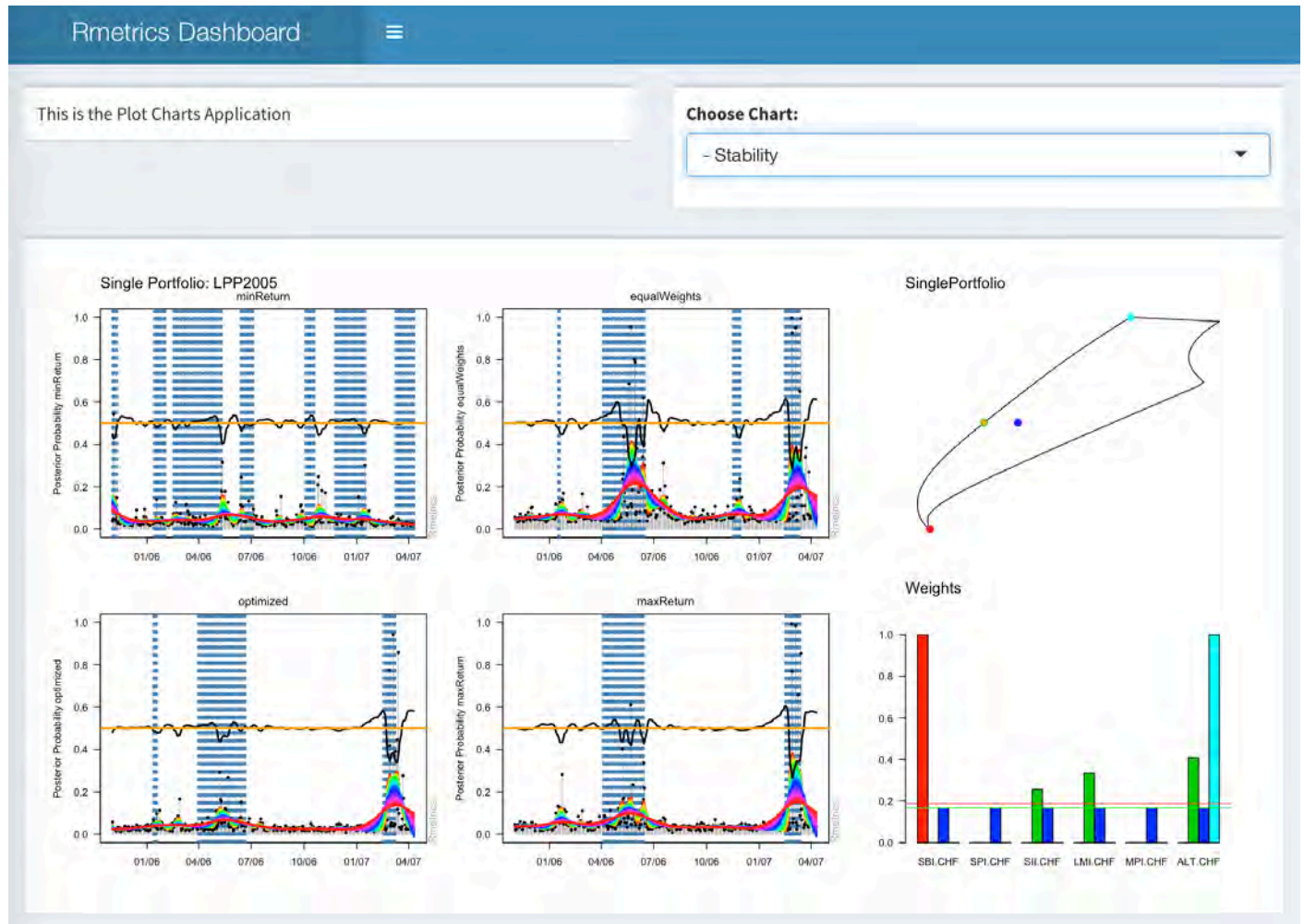
Weights

Wealth

Drawdowns

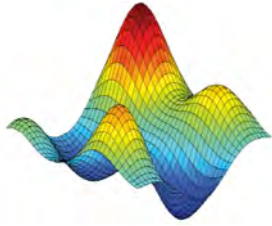


# Single Portfolio Designs



Analytics:

- Cycles
- Stability
- Outliers
- Spectrum



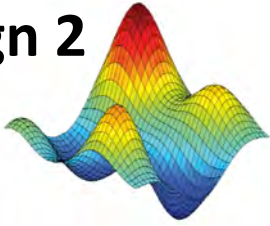
# Single Portfolio Designs

The type of the objective function together with the constraints define the *Complexity Class* of the design.

For a broad range of complexity classes our software pilot will formulate portfolio models and provide *Mathematical Programming Solvers* for their optimization.

Note, that each mathematical programming problem allows in its complexity class for additional *Constraints* including box, group, transaction costs, and integer constraints (like cardinality constraints and/or roundlot constraints).

## Design 2

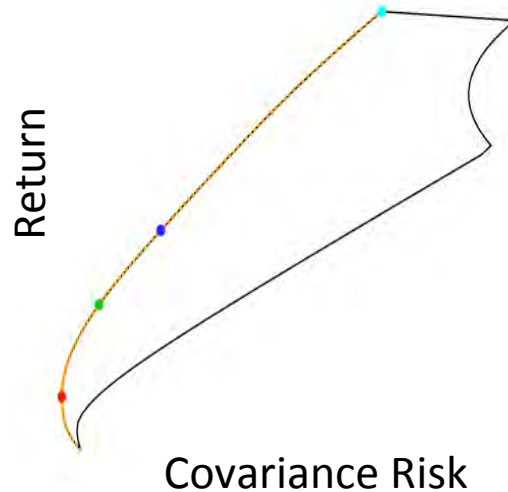


# Designs along Critical Lines

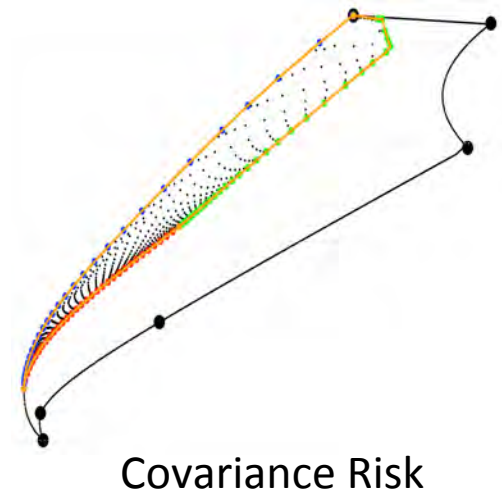
Efficient portfolios are formulated by loops over a performance, risk, or any other kind of portfolio objective.

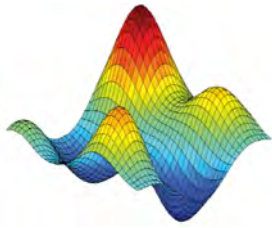
The portfolios are defined by a *Single Objective Criterion* subject to *Additional Constraints*.

Mean-Var Markowitz  
Efficient Frontier  
101 LPP Portfolios



Mean-Var Markowitz  
Pareto Hull  
502 LPP Portfolios

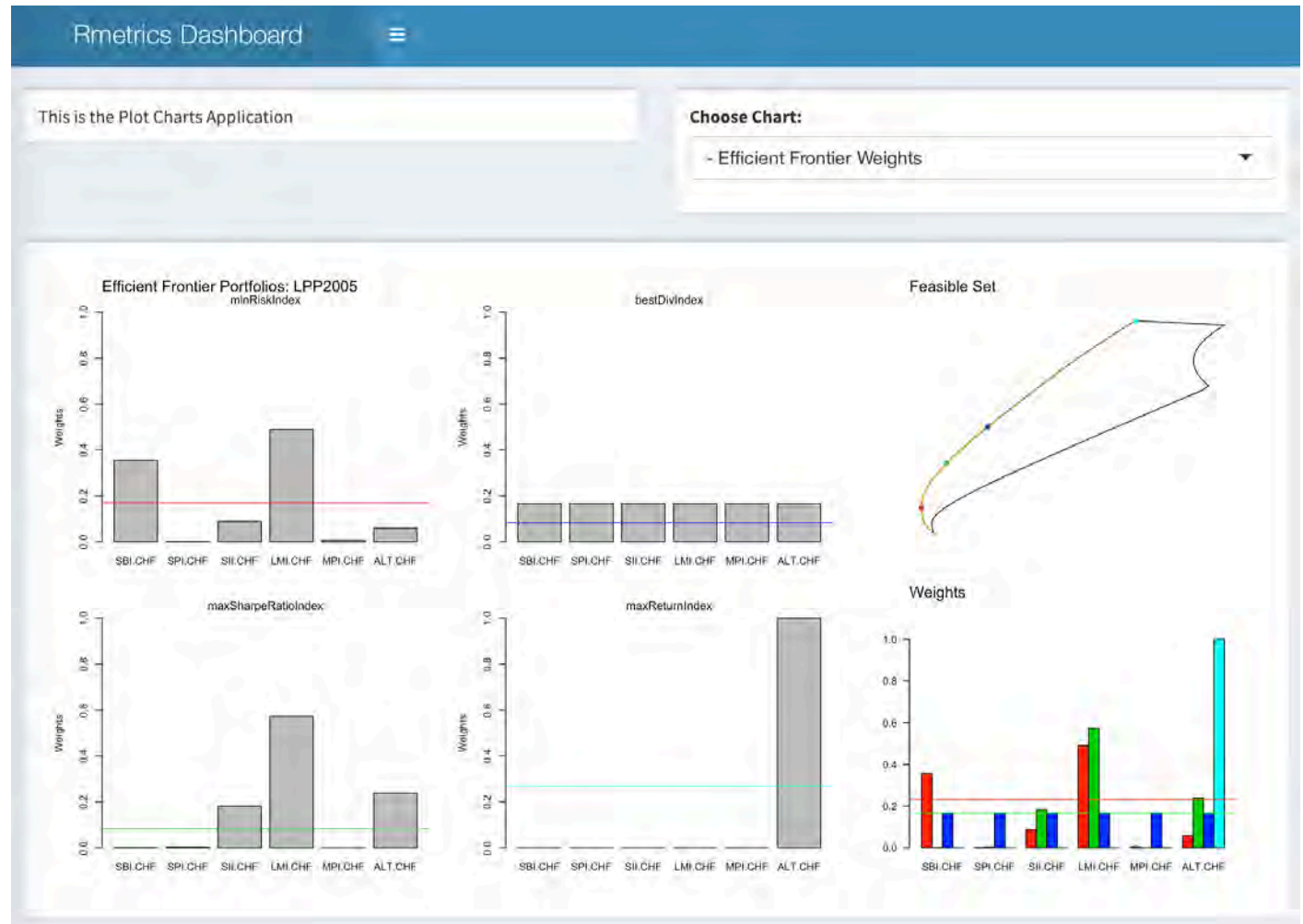




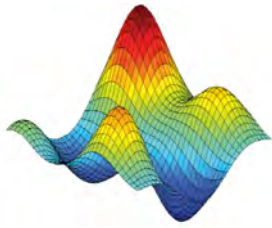
# Designs along Critical Lines

Optimal Weights  
along the Efficient  
Frontier:

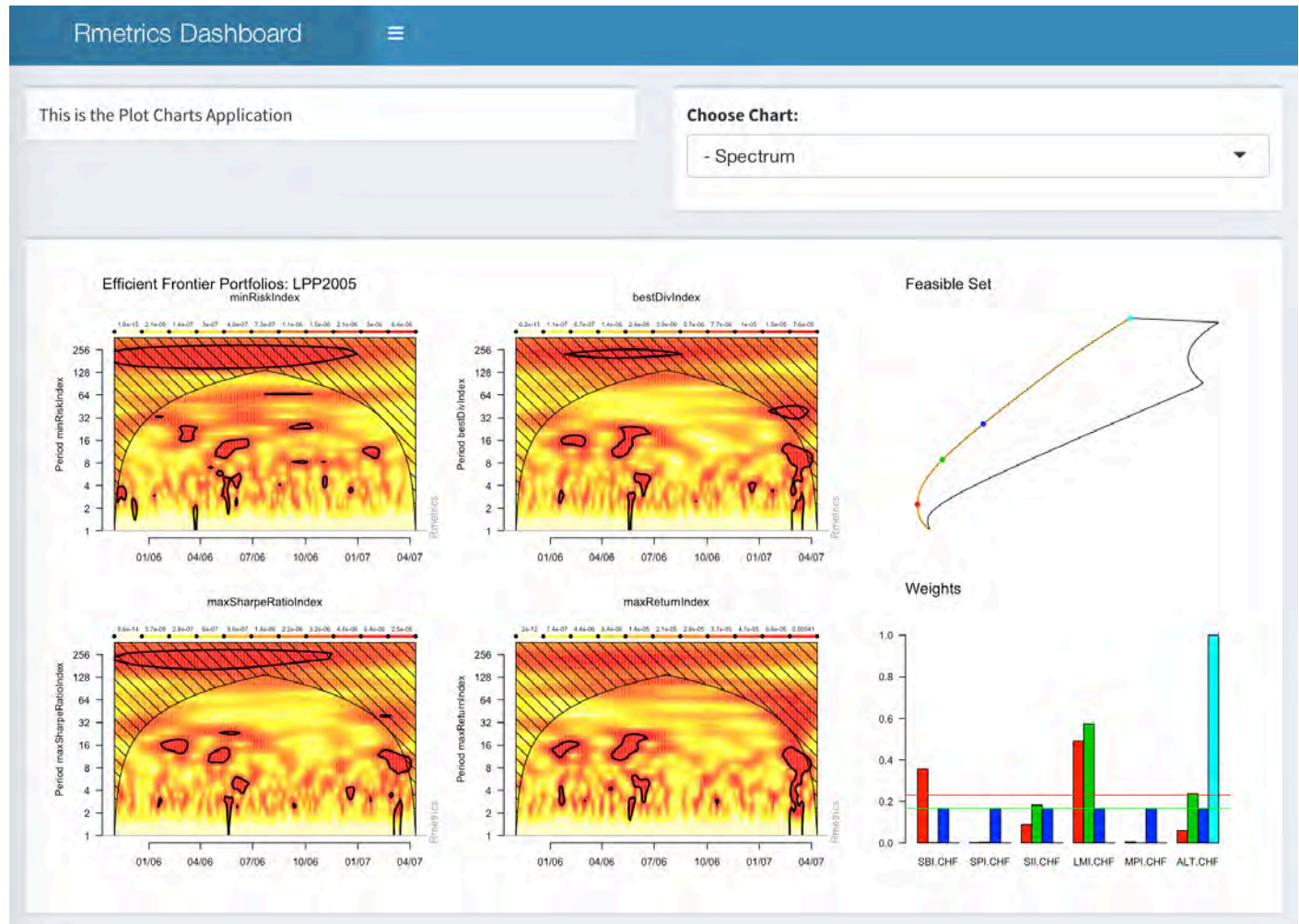
The optimal  
portfolio and three  
reference portfolios  
are displayed, all  
weights can be  
exported to CSV  
files.







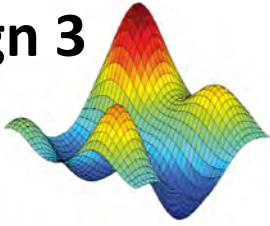
# Designs along Critical Lines



Power Spectrum:

The wavelet displays non-stationarities in the dynamics of portfolios along the efficient frontier.

## Design 3



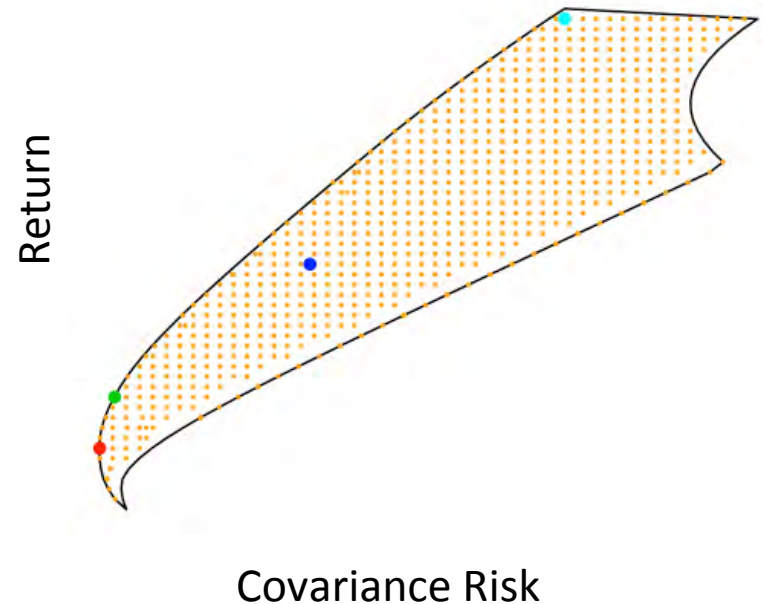
# Designs across Feasible Sets

These portfolios are formulated in the same way, but now by double loops over performance, risk, or any other kind of measure.

The design is defined with a single criterion subject to additional constraints.

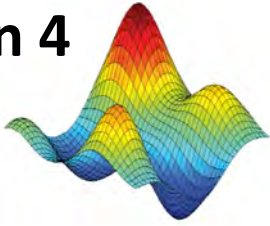
The loops are running across a square grid, where the first dimension represents the target risk and the second dimension the target return.

Mean-Var Markowitz Feasible Set  
502 Swiss Pension Fund Portfolios





## Design 4



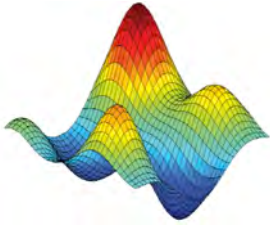
# Multi-Objective Designs

There exist different options for a multi-objective portfolio design.

The pilot will implement the *Scalarized Multi-Objective Optimization* approach.

- Scalarizing a multi-objective portfolio problem means to formulate a single-objective design in such a way that optimal solutions are *Pareto Optimal Solutions* to the multi-objective optimization problem.
- This allows to construct the complete feasible set of a portfolio since every Pareto optimal solution can be reached with a specific parameter set for the scalarization.

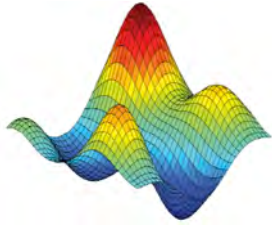
We follow this approach since it is straightforward to implement it and since this process is very general to construct the Pareto feasible set.



# Pareto Optimality

***Pareto optimality*** is a state of allocation of resources in which it is impossible to make any one component better off without making at least one component worse off.

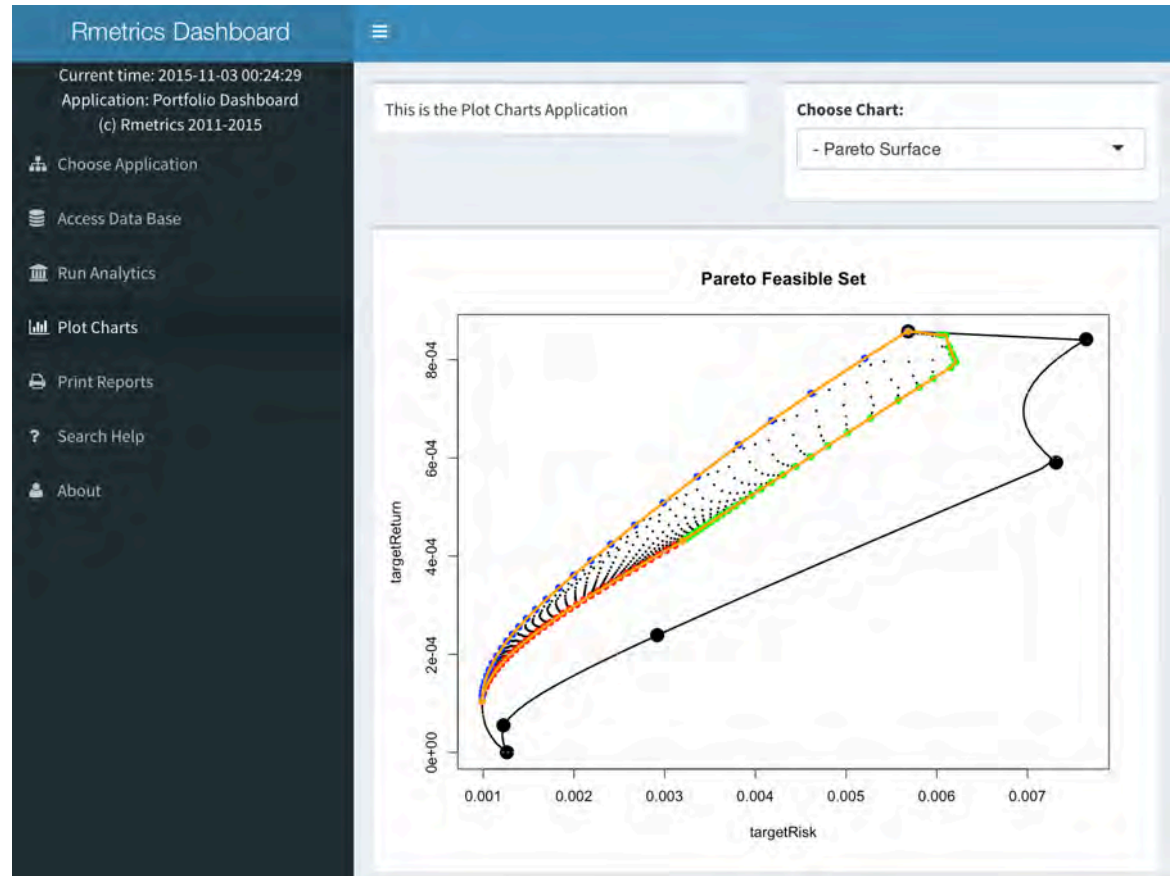
Given an initial allocation among a set of components, a change to a different allocation that makes at least one component better off without making any other individual worse off is called a ***Pareto Improvement***.



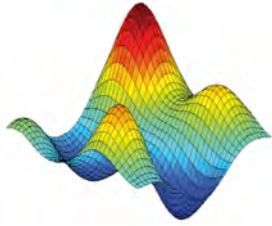
# Multi-Objective Designs

These portfolios involve more than one objective goals to be optimized simultaneously.

The most simple the bi-criterion optimization that can be described as an optimization along a critical line.



The next complex application is a tri-criterion portfolio optimization problem that delivers solutions with the so called Pareto feasible set.



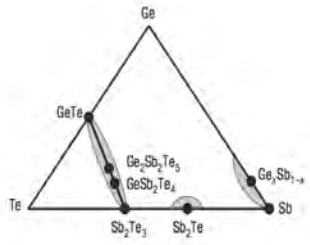
# Alternative Designs

The Software Pilot is ready to implement any other *Alternative Design* to portfolio optimization.

The user has just to formulate the portfolio *model* using a mathematical programming language, **AMPL**, to provide the required input *data*, and to select and *run* the appropriate mathematical programming solver.

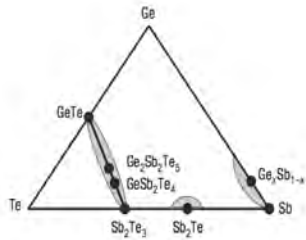
Note,

Rmetrics can help to add additional portfolio designs like tracking-error models, models that take care of transactions costs, or models that make use of the method of global criterion optimization, among others.



## 2 Ternary Maps

# Advanced Charts



Rmetrics Dashboard

Current time: 2015-11-03 00:28:48  
Application: Portfolio Dashboard  
(c) Rmetrics 2011-2015

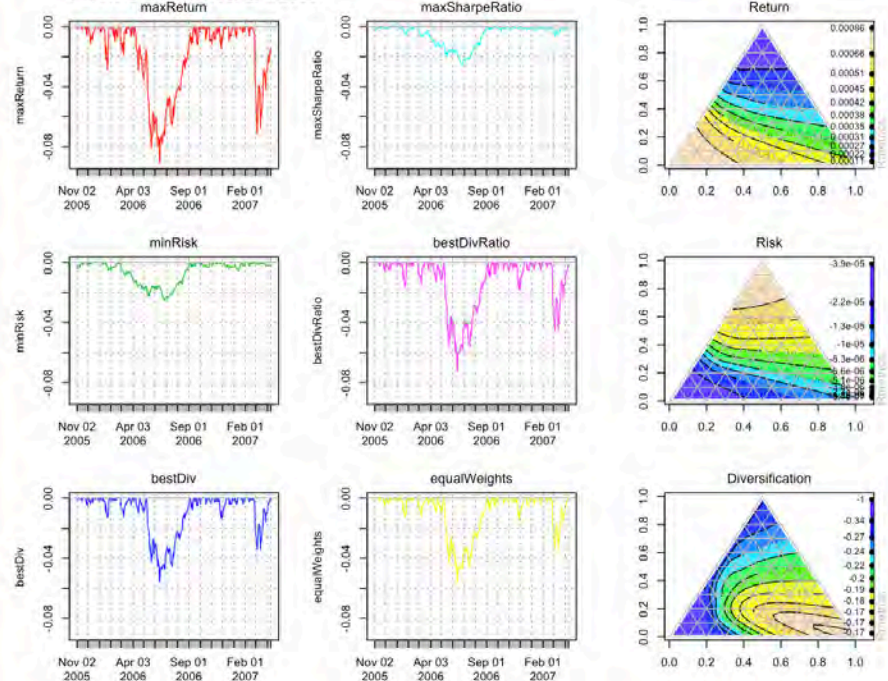
- Choose Application
- Access Data Base
- Run Analytics
- Plot Charts
- Print Reports
- Search Help
- About

This is the Plot Charts Application

Choose Chart:

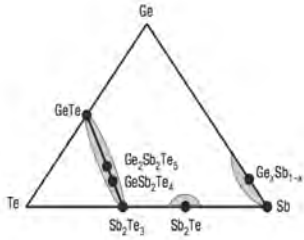
- Drawdowns

Pareto Set Portfolios: LPP2005



Drawdown Plots  
and  
Ternary Maps

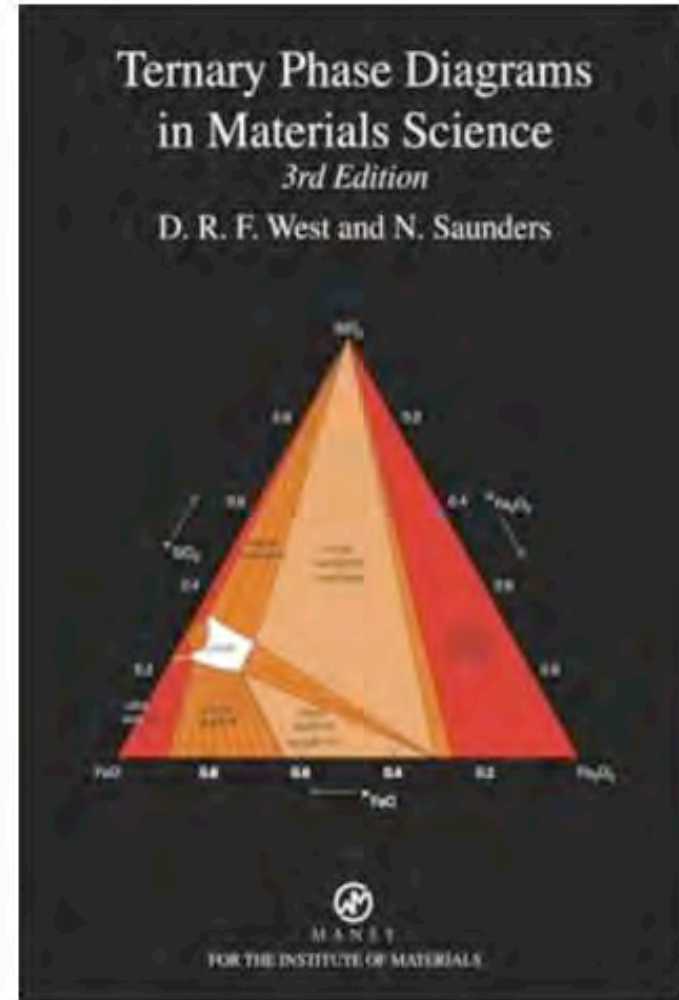
# Ternary Maps

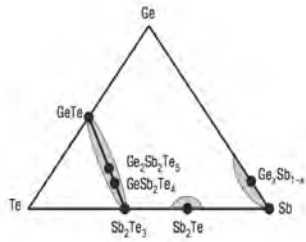


A Tri-Criterion Portfolio is composed of three goals, e.g.:  
*Return, Risk, and Diversification.*

“A bary-centric coordinate system is a coordinate system in which the location of a point is specified as the center of mass, or barycenter, of masses placed at the vertices of a simplex (a triangle, tetrahedron, etc).”  
*August Ferdinand Möbius (1827)*

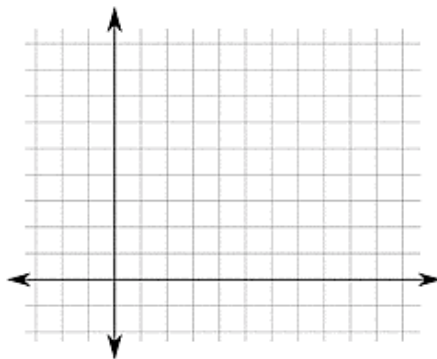
Ternary Plots are the most simple bary-centric plots.



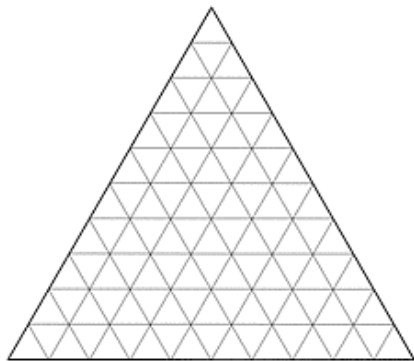


# How to Construct Ternary Maps

How to construct Ternary Maps from Bary-Centric Coordinates?



Cartesian Coordinate System



Ternary Diagram

Cartesian coordinates are useful for plotting points in the triangle. Consider an equilateral ternary plot where

$a = 100\%$  is placed at  $(x,y) = (0,0)$

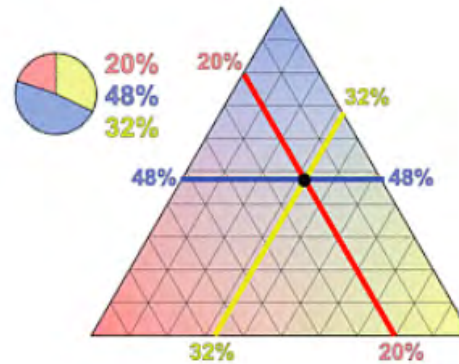
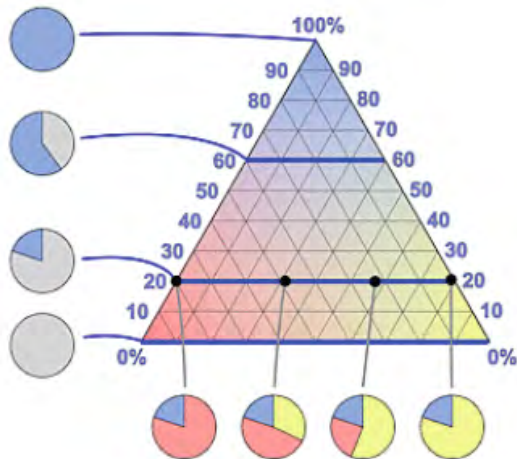
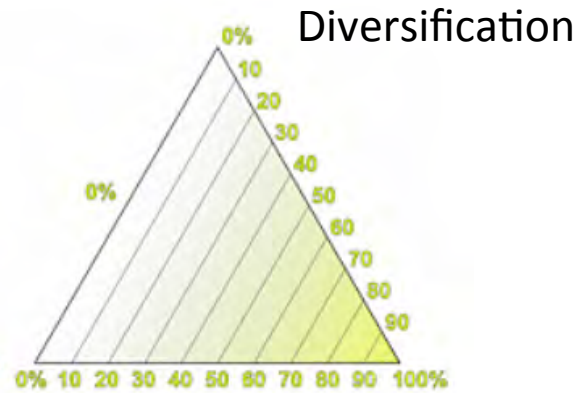
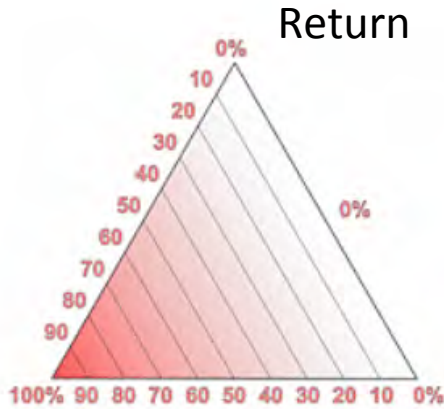
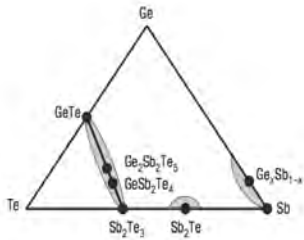
$b = 100\%$  at  $(1,0)$

$c = 100\%$  is  $\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$

the triple  $(a,b,c)$  is  $\left(\frac{1}{2} \frac{x + 2z}{x + y + z}, \frac{\sqrt{3}}{2} \frac{x}{x + y + z}\right)$



# How to Read Ternary Maps?



<http://www.quartzpage.de>



## 2 Model Library



# Portfolio Model Library

Rmetrics has started in 2010 to write a ***Portfolio Model Library*** that contains many *pre-formulated model files, data templates* and *run scripts*.

The library is written in the AMPL language and allows direct communication with the solvers and direction communication with R/Rmetrics through the ***R/AMPL*** interface, that is part of the Rmetrics ***fPortfolio*** package.

The ***AMPL Language*** supports the entire optimization modeling lifecycle, *Formulation, Testing, Deployment, and Maintenance* in an integrated way: Optimization models are described in the same ways that people think about them. (AMPL is a product from ***AT&T*** Bell Laboratories.)



# Markowitz & CVaR Models

The following portfolio models are provided for single portfolio designs, for portfolio designs along the efficient frontier, and for portfolio designs across the feasible set.

- Mean-Variance Markowitz-type designs
  - Min CovarianceRisk s.t. meanReturn
  - Max meanReturn s.t. CovarianceRisk
- Scenario based Mean-CVaR Uryasev/Rockafeller-type designs
  - Min CVaR s.t. meanReturn
  - Max meanReturn s.t. CVaR



# Multi-Objective Models

In the case of multi-objective portfolio designs the following tri-criterion models are implemented by default

- Mean Return - Total Covariance Risk - Quadratic Diversification
- Mean Return - Total CVaR Risk - Quadratic Diversification
- Tracking Error - Total Covariance Risk - Cardinality



# Mean-CVaR-Diversification

Example:

$$\min_{\bar{x} \in \mathcal{X}} -\lambda_1 \cdot \text{Return}(\bar{x}) + \lambda_2 \cdot \text{CVaR}(\bar{x}) - \lambda_3 \cdot \text{Diversification}(\bar{x})$$

*s.t.*

$$\lambda_1 + \lambda_2 + \lambda_3 = 1$$

$$f_1(\bar{x}) := -\bar{x}^T \bar{\mu}$$

Total Return

$$f_2(\bar{x}) := \text{CVaR}(\bar{x})$$

Conditional Value at Risk

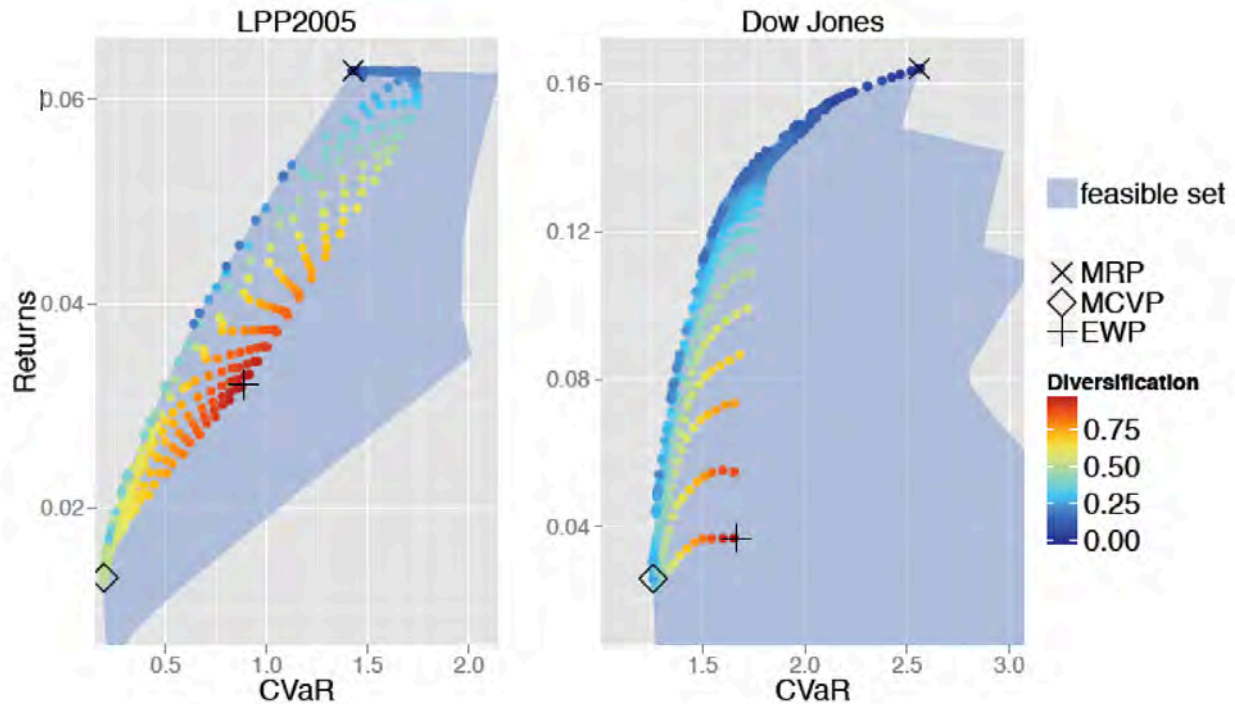
$$f_3(\bar{x}) := \bar{x}^T \mathbf{I} \bar{x}$$

Diversification (Herfindhal Index)



# Mean-CVaR-Diversification

Results:



Portfolio positions in the CVaR-Return plane when optimized with goal programming. The points are drawn from a grid-with distance  $1/30$  between the  $\lambda$ -values. The color of a point indicates the diversification of the portfolio.



# Mean-CVaR-Diversification

Computation Times:

	Numer of assets	5	25	50	100
Runtime [s]	Minos	11.5	12.2	16.1	19.5
	CPLEX	13.75	47.5	59.9	68.2
	Gurobi	17.3	20.6	46.5	39.5
	SNOPT	9.8	12.5	17.9	18.9

	$\lambda$ -grid	1/15	1/30	1/60	1/120
Runtime [s]	Minos	4.2	13.9	61.2	218.9

The calculations were done on a personal computer running Windows 7-OS with 8 GB memory and an Intel Core i7-2600 -processor with four 3.4 GHz cores.





# Multi-Objective Models

Mean Return - Total Covariance Risk - Quadratic Diversification  
AMPL Portfolio Model File

```
# MVGP – Goal Programming Diversified Mean–Variance Portfolio:
param N ;
param targetReturn ;
param targetRisk ;
param targetDiv ;
param lambda1 ;
param lambda2 ;
param mu{1..N} ;
param Sigma{1..N, 1..N} ;
param Div{1..N, 1..N} ;
var w{1..N} >= 0;
maximize Objective:
    (1 - lambda1 - lambda2 ) *sum{i in 1..N} mu[i] * w[i] / targetReturn -
    lambda1 * sum{i in 1..N} sum{j in 1..N} w[i] * Sigma[i,j] * w[j] / targetRisk -
    lambda2 * sum{i in 1..N} sum{j in 1..N} w[i] * Div[i,j] * w[j] / targetDiv ;
subject to Budget: sum{i in 1..N} w[i] = 1 ;
```



# Multi-Objective Models

## Example: Cardinality Constraints

```
# Cardinality Addon - Cardinality Constraints:
param lower{1..N} ;
param upper{1..N} ;
param kCard ;
var delta{1..N} binary ;
subject to BuyinLower{i in 1..N}:
    w[i] - lower[i] * delta[i] >= 0 ;
subject to BuyinUpper{i in 1..N}:
    w[i] - upper[i] * delta[i] <= 0 ;
subject to Cardinality: sum{i in 1..N} delta[i] <= kCard ;
```



# Investigated Aspects

The three implemented models can take several aspects into account:

- Covariance Robustification: More than a dozen of different robust covariance estimators are part of the Rmetrics *fAssets* package.
- The tri-criterion Markowitz problem with quadratic weights diversification was investigated in an ETH Bachelor Thesis project by **Tomas Tamfal** [2015].
- The tri-criterion CVaR problem with quadratic weights diversification was investigated in an ETH Master Thesis project by **Henry Claussen** [2015].
- The tri-criterion Markowitz portfolio problem where the third criterion measures bivariate Copula tail dependence risk is currently investigated in an ETH Bachelor Thesis project at by **Daniel Merten** [2015].
- The tri-portfolio problem, with objectives Tracking Error, Total Covariance Risk, and minimum Cardinality is investigated in an ETH Seminar thesis project by **Tobias Hatt** [2015]. Note, for this design the cardinality objective leads to a global, non-convex mixed integer optimization problem.

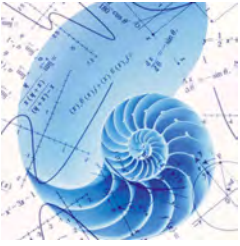


# Usage

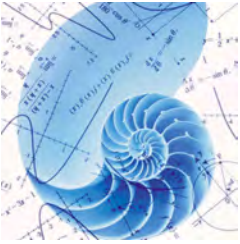
The Portfolio Model Library written in the AMPL Language is continuously growing and supported by Rmetrics:

- Rmetrics adds continuously *new models* to its library.
- *Maintenance* of the library from the software side by Rmetrics
- The library is *extendable* by the user with his own private and tailored models.

New models include designs with transaction costs, covariance, risk budgeting terms, and many other objectives.



# 3 Mathematical Programming

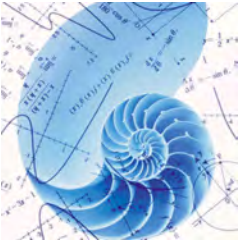


# Math Programming Solvers

The different types of portfolio models require different types of mathematical programming solvers depending on the *structure of the objective function* and the *structure of the constraints*.

State-of-the-art mathematical programming solvers that allow to solve all types of constrained portfolio models including

- linear, quadratic, non-linear, global, and mixed integer *objectives*
- box, group, quadratic, non-linear and integer types of *constraints*



# Math Programming Solvers

Three groups of solvers are implemented by default that cover the whole range of possible portfolio models:

- *Coin-Or Open Source Suite* of Mathematical Programming Solver (the Default Solvers)
- *AMPL Free Limited Version* of Selected Commercial Solvers (can be extended by commercial solvers)
- *NEOS Server* Environment and Infrastructure (allows to test Open Source and Commercial Solvers)



## 4 Post-Processing Tools





# Post-Processing: Charts & Reports

As an important feature of the Software Pilot, the pilot will not only calculate optimal portfolio solutions.

The pilot will also contain functions and tools

- to plot charts
- to print reports

from the results of the optimal portfolio design.



# List of Chart Tools

Charts include the

- display of the *optimal weights*, of *risk budgets*, *wealth*, *drawdowns* curves, and many other measures
- stability analysis based on *Bayesian change points*, *extreme and outlying points* detection, and on *wavelet analysis* visualizing non-stationarities.

The Charts are displayed using statistical visualization tools build

- on performance, risk and stability surfaces
- on heat maps
- on spectral images
- on ternary decision maps



# Multi-Objective Chart Tools

For multi-objective portfolio designs the following plots generate additional information:

- Pareto Performance, Risk and Diversification Surfaces
- Pareto Ternary Maps from Tree-Criterion Designs
- Pareto Optimality Views



# Report Tools

Reports are listing and summarizing the associated numbers obtained from optimization and the post-processing statistical analysis.

- They go hand in hand with the chart tools



# List of Report Tools

To be specific the following Print Reports can be generated:

- Portfolio Scenarios for all Components
- Portfolio Specifications from model, data, and run Files
- Portfolio Optimal Weights
- Portfolio Risk Budgets
- Portfolio Solver Information from Optimization
  
- Listing of all Available Portfolio Designs
- Listing of all Available Mathematical Programming Solvers



# Source of Chart & Report Tools

Chart, Report, and the Analytics Tools are linked from the Open Source Rmetrics *fBasics*, *fAssets*, and *fPortfolio* packages.

For the Garch volatility analysis of portfolios we use the functions from the *fGarch* package.

The lists for the charts and reports are user extendable. The user can write his own R functions for additional charts and reports and can add them to the list.

# Part 2: Software Pilot



# 1 R and Rmetrics





# Design of the Software Pilot

The Software Pilot is something like a draft version of the Portfolio Dashboard Environment. The software pilot will be build on the following pillars:

- RStudio GUI and RShiny Web Framework
- Ace Expert Edior for Portfolio Model Design, GUI (optional)
- Easy and Straightforward Portfolio Modeling with AMPL
- The R/AMPL Interface to access the Solvers
- COIN-OR Open Source Mathematical Programming Solvers
- AMPL's Commercial Solver Family Test Version
- The NEOS Solver Environment and Interface
- Data Base Access: CSV files, SQL RDBMs and/or Feeds (optional)
- Rmetrics Dashboard Configuration Files
- Command Line Batch Processing (optional)



# Good Reasons

We have chosen these pillars since we can rely on a broad experience working with these components:

- All our software development is done with R/RStudio
- All our applications are build in the RShiny application framework
- The R/AMPL solver interface was already developed in [2010]
- We are working with the AMPL Language since several years
- We are experienced with COIN-OR solvers and the NEOS Interface

We are sure that the combination of these software components and modules will create and guarantee a powerful environment for multi-objective portfolio optimization.



# Why to use it ?

- It's free! If you are a teacher or a student, the benefits are obvious.
- It runs on a variety of platforms including Windows, Unix and MacOS X.
- It provides an unparalleled platform for programming new statistical methods in an easy and straightforward manner.
- It contains advanced statistical routines not yet available in other packages.
- It has state-of-the-art graphics capabilities.



## Why to use it?

- Rmetrics is an state-of-the-art collection of advanced R functions for computational finance and financial engineering.
- From R's financial core packages 17 out of 22 have been contributed by Rmetrics.
- Rmetrics maintains more then 40 R packages contributed from more then 20 developers worldwide.
- Rmetrics has strong roots in training and teaching the R/Rmetrics environment, modern algorithms, and writing textbooks.
- Rmetrics is Open Source.



# Rmetrics Software Development

## Rmetrics Software Evaluation

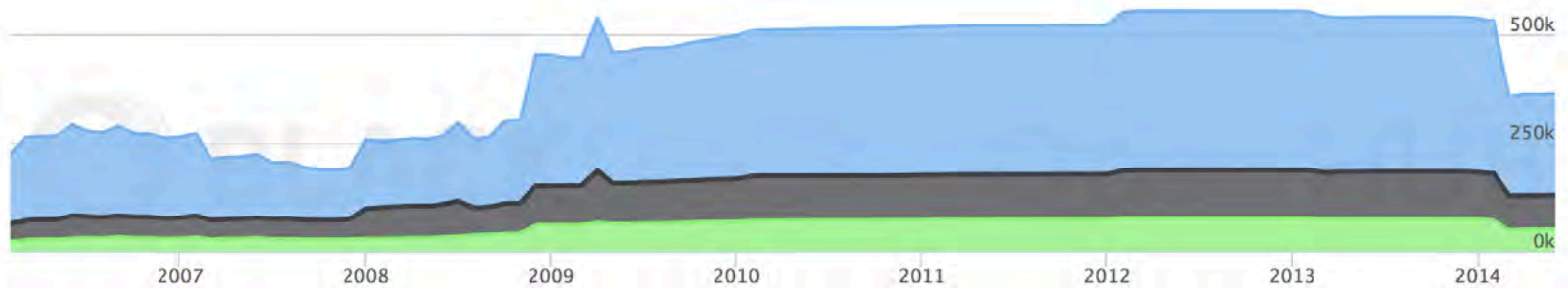
<https://www.openhub.net>

**Project Cost Calculator**

Include: All Code  
Average Salary (per year): \$ 80000 .00  
Codebase Size: 233,025 lines  
Estimated Effort: 60 person-years  
Estimated Cost: \$ 4,812,021 \*

\*Using the Basic COCOMO Model

Code, Comments and Blank Lines



Language Breakdown

Language	Code Lines	Comment Lines	Comment Ratio	Blank Lines	Total Lines	Total Percentage
R	164,607	62,539	27.5%	40,175	267,321	73.6%
C	31,685	13,473	29.8%	5,515	50,673	14.0%



# Rmetrics Community Channels

Meetings and Trainings  
that means Sustainability in Development



- Rmetrics Association organizes since 2007 every year **workshops, summer schools, master classes, trainings, tutorials**. An incomplete list of the events can be found here:

<https://www.rmetrics.org/LecturesCourses>



- In **more than 50 training courses** Rmetrics Association has educated more than 500 participants from all over the world. Most of the Courses were given in Zurich, Mumbai/Singapore (Rmetrics Asia). The most recent workshop was held at ETH Villa Hatt in Zurich:

<https://www.rmetrics.org/zurich2015>



# Bridging Academia and Industry

Rmetrics offers the **Rmetrics Juniors** Program.

Rmetrics Juniors was founded in 2014 to build a **Bridge** between theory and practical applications between academia and industry.

Rmetrics brings motivated students in **Internships** and projects closer to companies to conduct problems, to analyze applications and to develop creative solutions in R.





## 2 R Studio and Shiny





# RStudio

We are convinced that *RStudio* is the best environment for the development of the Software Pilot and its future continuous extension.

The RStudio IDE is the most powerful and most productive user interface to work with R. RStudio is free and open source, and works under the *Windows*, *Mac*, and *Linux* operating systems.

Therefore we have decided to develop the software pilot with RStudio.



# RStudio

The screenshot displays the RStudio environment with three main panes:

- Source Editor:** Contains an R script named `runShiny.R`. The script includes comments and code for setting a barcapId and tailored settings like `chartWidth` and `chartHeight`.
- Console:** Shows the execution output, including function calls like `amplDataFile`, `amplDataAdd`, `amplRunFile`, and `system`, along with a command `ampl myPortfolio.run > myPortfolio.out` and a message "Read 4566 items".
- Help Pane:** Displays the documentation for the `write` function, titled "Write Data to a File". It includes a description, usage examples, and arguments.

```
write(x, file = "data",
      ncolumns = if(is.character(x)) 1 else 5,
      append = FALSE, sep = " ")
```

**Arguments**

- `x`: the data to be written out, usually an atomic vector.
- `file`: A connection, or a character string naming the file to write to. If "", print to the standard output connection. If it is "|cmd", the output is piped to the command given by 'cmd'.
- `ncolumns`: the number of columns to write the data in.
- `append`: if TRUE the data x are appended to the connection.
- `sep`: a string used to separate columns. Using `sep = "\t"` gives tab delimited output; default is " ".

**References**

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.



# RShiny

We have also decided to integrate the software pilot into the RShiny application framework.

- ***RShiny*** is an extremely elegant and powerful web framework for integrating the portfolio engine, as well as building tools for the interactive visualizations and reports using R.
- The strength of RShiny is in its *Reactive Programming Model*. The RShiny dashboard technology is a powerful approach that allows to design and implement dashboards in a straightforward way.
- We have implemented as part of the Software Pilot the Open Source *Lightweight Public Applications*. The commercial version also supports *authentication* and allows *multiuser* access (optional).

For an an easy programming of the dashboard we use an *Intermediate Software Layer* provided by the contributed R package ***shinydashboard***.




# RShiny


Contributed R Packages: shiny + shinydashboard + shinyace


The screenshot shows the Rmetrics Dashboard interface. The top navigation bar is blue and contains the text "Rmetrics Dashboard" and a hamburger menu icon. Below the navigation bar, the main content area is white. On the left side, there is a dark grey sidebar with a list of menu items, each preceded by a small icon: "Choose Application" (person icon), "Access Data Base" (database icon), "Run Analytics" (building icon), "Plot Charts" (chart icon), "Print Reports" (printer icon), "Search Help" (question mark icon), and "About" (person icon). The main content area on the right contains the text "Current time: 2015-11-03 00:44:07", "Application: Portfolio Dashboard", and "(c) Rmetrics 2011-2015" in the top left corner. In the top right corner, there are two logos: "Rmetrics" (a blue logo with a stylized 'R') and "ETH zürich" (the logo of the Swiss Federal Institute of Technology in Zurich).





# 3 Data Base Management


 Choose Application


 Access Data Base

 Run Analytics

 Plot Charts

 Print Reports

 Search Help

 About



# Access Data Base


The default data base access is provided through CSV files.


- The date is listed in the first column in ISO-8601 format.
  - The CSV files are read in from Rmetrics as chronological S4 objects.
  - The data handling is done using R functions from the Rmetrics packages `timeDate` and `timeSeries`.
- In the following columns the financial returns are stored.


As alternatives Rmetrics offers in addition an SQL data base interconnection and access to commercial data feeds (Bloomberg API and IBrokers API) and the Internet (Rmetrics Web API). This is optional.





## 4 Run Analytics: Optimize


 Choose Application


 Access Data Base

 Run Analytics

 Plot Charts

 Print Reports

 Search Help

 About



# Expert Editor

The Open Source Java based Ace expert editor is in the heart of the dashboard. The editor is easy to use.

- Although a GUI may look more attractive and convenient, we have decided to implement the expert editor interface from Ace since it is much more flexible in its usage.
- This software design offers the huge advantage to add new and alternative portfolio designs from the user's side just on the fly. *There exists no portfolio model that cannot be designed with the Rmetrics Dashboard!*

Ace is provided by the contributed R package ***shinyAce***.

Note: Rmetrics can develop a graphical user interface that extends the Ace expert editor (optional).





# Run Analytics: Ace Expert Editor

Rmetrics Dashboard

Current time: 2015-11-02 11:00:39  
Application: Portfolio Dashboard  
(c) Rmetrics 2011-2015

- Choose Application
- Access Data Base
- Run Analytics
- Plot Charts
- Print Reports
- Search Help
- About

This is the Run Analytics Application

Click into textarea field to update from last entry:

```
1
2 # PROJECT NAME:
3 project <- 'myPortfolio'
4
5 # SELECT MODEL FROM LIBRARY OR DEFINE YOUR OWN MODEL
6 model <- modelMV1FS
7
8 # GENERATE AUTOMATED DATA USING dataAUTO() OR GENERATE MANUALLY:
9 Scenarios <- loadScenarios()
10 # General Settings:
11 N <- ncol(Scenarios)
12 targetReturn <- mean(Scenarios)
13 targetRisk <- mean(cov(Scenarios))
14 mu <- colMeans(Scenarios)
15 Sigma <- cov(Scenarios)
16 data <- dataAUTO(model)
17
18 # GENERATE AUTOMATED RUN USING runAUTO() OR GENERATE MANUALLY:
19 run <- runAUTO(model, project, solver='cplex')
20
21 # OPTIMIZE:
22 AMPL <- list(project = project, model = model, data = data, run = run)
23 solvePortfolio(Scenarios, AMPL, trace=TRUE)
```

Update Portfolio Dashboard:

Optimizing a Mean-Variance QP1 Markowitz Portfolio



# Reasons why to use AMPL

We use the AMPL language to describe and setting up the portfolio models.

- AMPL integrates a modeling language for describing input parameters, optimization variables, objective functions, and constraints
- AMPL allows for manipulating data, and for implementing iterative optimization schemes.
- All models to be optimized use the same concepts and syntax for streamlined application-building independent of the mathematical programming solver in use.
- AMPL offers valuable modeling support features. An interactive command environment with batch processing options is provided, that can be used from R/Rmetrics.



# Reasons why to use AMPL

## Example: AMPL Language: Mean-CVaR Single Portfolio Model

```
# CVAR1 – Single Min-Risk Mean-CVaR Portfolio:
param alpha ;
param N ;
param S ;
param mu{1..N} ;
param targetReturn ;
param Scenarios{1..S,1..N} ;
var w{1..N} >= 0 ;
var VaR ;
var x{1..S} >= 0 ;
maximize Objective: VaR - ( sum{s in 1..S} x[s] ) / ( alpha*S ) ;
subject to Budget: sum{i in 1..N} w[i] = 1 ;
subject to Return:
    sum{i in 1..N} mu[i] * w[i] = targetReturn ;
subject to CVaR{s in 1..S}:
    sum{i in 1..N} Scenarios[s,i] * w[i] - VaR + x[s] >= 0 ;
```



# Reasons why to use AMPL

## Example: AMPL Language: Cardinality Constraints

```
# Cardinality Addon - Cardinality Constraints:
param lower{1..N} ;
param upper{1..N} ;
param kCard ;
var delta{1..N} binary ;
subject to BuyinLower{i in 1..N}:
    w[i] - lower[i] * delta[i] >= 0 ;
subject to BuyinUpper{i in 1..N}:
    w[i] - upper[i] * delta[i] <= 0 ;
subject to Cardinality: sum{i in 1..N} delta[i] <= kCard ;
```



# R/AMPL Interface

To interface a portfolio problem from R/Rmetrics to AMPL we make use of the already existing R/AMPL Interface from the Rmetrics ***fPortfolio*** package (A concept similar to that one used by Matlab/TOMLAB).

The interface translates portfolio designs from R into the language of AMPL and creates the AMPL *model*, *data* and *run* files.

- the model file that keeps all the relevant model information
- the data file which stores the required parameters and data
- the run file which provides the execution batch commands

Optimization is done in the background calling the AMPL files.

The interface returns the optimization results from AMPL, i.e. solver information and optimal weights. The final results can be visualized in charts, printed to reports, and archived to files.



# 5 Math Programming Solvers



# Math Programming

We have evaluated a whole bundle of open source and commercially available solvers that work together with AMPL.

The solvers we have investigated are those from the open source project Coin-OR ([www.projects.coin-or.org](http://www.projects.coin-or.org)) and

selected (size-limited) solvers from commercial software providers available from the AMPL website ([www.ampl.com](http://www.ampl.com)).



# Solver Family: Open Source

The *Computational Infrastructure for Operations Research* project, COIN-OR is an initiative to spur the development of open-source software for operations research. COIN-OR solvers are in no issue behind commercial solutions.

The size of the COIN-OR projects is enormous. COIN-OR offers developer tools, excellent documentation, interfaces, and modeling systems. The solver applications range from

- convex non-differentiable problems,
- deterministic linear continuous problems,
- deterministic linear discrete problems,
- deterministic non-linear problems,
- deterministic non-linear discrete problems,
- deterministic semi-definite continuous problems,
- non-differentiable problems, to
- stochastic problems.





# Solver Family: Open Source

Rmetrics has a long experience with the use of COIN-OR solvers. We have implemented them in the Software Pilot: The implementation includes

- CBC – A Branch & Cut Mixed Integer Programming Solver
- IPOPT – An Interior Point OPTimizer for large-scale non-linear optimization
- BONMIN – for general (MINLP) Mixed Integer Non-Linear Programming
- COUENNE – Convex Over and Under ENvelopes for Nonlinear Estimation,  
A Branch & Bound solver for Mixed-Integer Non-Linear Programming

Note:

CBC, BONMIN and COUENNE solvers allow for Mixed Integer Programming.  
COUENNE allows to solve global, non-convex portfolio objectives.



# Size-Limited Versions: AMPL

From the commercial solvers we have implemented three *free size-limited versions*, that are available directly from AMPL\*. These are

- **CPLEX** – from IBM is the most widely used large-scale solver. The problem types supported include linear and quadratic optimization in continuous and integer variables. Support is provided for both convex and nonconvex quadratic objectives, for convex quadratic constraints also of elliptic and conic types.
- **SNOPT** – from Stanford Systems Optimization Laboratory is a widely used optimizer for linear, quadratic, and smooth nonlinear objectives and constraints in continuous variables..
- **MINOS** – from Stanford Systems Optimization Laboratory is an established choice for both linear and nonlinear optimization problems.

\* <http://ampl.com/try-ampl/download-a-demo-version>



# Resources in the Cloud

The NEOS server is a free internet-based service for testing and solving numerical optimization problems. It supports all relevant Open Source and Commercial solvers by AMPL and can be accessed through a R/NEOS interface.

- The NEOS Server provides access to more than 60 state-of-the-art solvers in more than a dozen optimization categories.
- Solvers run on distributed high-performance machines at Argonne National Laboratory, Arizona State University.
- The NEOS Guide web site provides showcase optimization studies, presenting optimization information and resources on the NEOS Server.
- The R/NEOS interface in the R package *rneos* allows to access the NEOS Server from the Portfolio Dashboard.



# Solvers @ NEOS

## 1. NEOS Solvers for Linear Programming Problems:

- bpmpd – [www.neos-server.org/neos/solvers/lp:bpmpd/AMPL.html](http://www.neos-server.org/neos/solvers/lp:bpmpd/AMPL.html)
- CPLEX – [www.neos-server.org/neos/solvers/lp:CPLEX/AMPL.html](http://www.neos-server.org/neos/solvers/lp:CPLEX/AMPL.html)
- Gurobi – [www.neos-server.org/neos/solvers/lp:Gurobi/AMPL.html](http://www.neos-server.org/neos/solvers/lp:Gurobi/AMPL.html)
- MOSEK – [www.neos-server.org/neos/solvers/lp:MOSEK/AMPL.html](http://www.neos-server.org/neos/solvers/lp:MOSEK/AMPL.html)
- OOQP – [www.neos-server.org/neos/solvers/lp:OOQP/AMPL.html](http://www.neos-server.org/neos/solvers/lp:OOQP/AMPL.html)
- XpressMP – [www.neos-server.org/neos/solvers/lp:XpressMP/AMPL.html](http://www.neos-server.org/neos/solvers/lp:XpressMP/AMPL.html)

e.g.

The mean-CVaR problem has a linear objective and linear constraints.



# Solvers @ NEOS

## 2. NEOS Solvers for Mixed Integer Linear Programming Problems:

- Cbc – [www.neos-server.org/neos/solvers/milp:Cbc/AMPL.html](http://www.neos-server.org/neos/solvers/milp:Cbc/AMPL.html)
- CPLEX – [www.neos-server.org/neos/solvers/milp:CPLEX/AMPL.html](http://www.neos-server.org/neos/solvers/milp:CPLEX/AMPL.html)
- feaspump – [www.neos-server.org/neos/solvers/milp:feaspump/AMPL.html](http://www.neos-server.org/neos/solvers/milp:feaspump/AMPL.html)
- Gurobi – [www.neos-server.org/neos/solvers/milp:Gurobi/AMPL.html](http://www.neos-server.org/neos/solvers/milp:Gurobi/AMPL.html)
- MINTO – [www.neos-server.org/neos/solvers/milp:MINTO/AMPL.html](http://www.neos-server.org/neos/solvers/milp:MINTO/AMPL.html)
- MOSEK – [www.neos-server.org/neos/solvers/milp:MOSEK/AMPL.html](http://www.neos-server.org/neos/solvers/milp:MOSEK/AMPL.html)
- qsopt\_ex – [www.neos-server.org/neos/solvers/milp:qsopt\\_ex/AMPL.html](http://www.neos-server.org/neos/solvers/milp:qsopt_ex/AMPL.html)
- scip – [www.neos-server.org/neos/solvers/milp:scip/AMPL.html](http://www.neos-server.org/neos/solvers/milp:scip/AMPL.html)
- XpressMP – [www.neos-server.org/neos/solvers/milp:XpressMP/AMPL.html](http://www.neos-server.org/neos/solvers/milp:XpressMP/AMPL.html)

e.g. Roundlot and Cardinality constrained problems may fall in this solver class.



# Solvers @ NEOS

3 Solvers for Non-Linearly Constrained Optimization Problems:

- CONOPT – [www.neos-server.org/neos/solvers/nco:CONOPT/AMPL.html](http://www.neos-server.org/neos/solvers/nco:CONOPT/AMPL.html)
- filter – [www.neos-server.org/neos/solvers/nco:filter/AMPL.html](http://www.neos-server.org/neos/solvers/nco:filter/AMPL.html)
- Ipopt – [www.neos-server.org/neos/solvers/nco:Ipopt/AMPL.html](http://www.neos-server.org/neos/solvers/nco:Ipopt/AMPL.html)
- KNITRO – [www.neos-server.org/neos/solvers/nco:KNITRO/AMPL.html](http://www.neos-server.org/neos/solvers/nco:KNITRO/AMPL.html)
- LANCELOT – [www.neos-server.org/neos/solvers/nco:LANCELOT/AMPL.html](http://www.neos-server.org/neos/solvers/nco:LANCELOT/AMPL.html)
- LOQO – [www.neos-server.org/neos/solvers/nco:LOQO/AMPL.html](http://www.neos-server.org/neos/solvers/nco:LOQO/AMPL.html)
- MINOS – [www.neos-server.org/neos/solvers/nco:MINOS/AMPL.html](http://www.neos-server.org/neos/solvers/nco:MINOS/AMPL.html)
- MOSEK – [www.neos-server.org/neos/solvers/nco:MOSEK/AMPL.html](http://www.neos-server.org/neos/solvers/nco:MOSEK/AMPL.html)
- SNOPT – [www.neos-server.org/neos/solvers/nco:SNOPT/AMPL.html](http://www.neos-server.org/neos/solvers/nco:SNOPT/AMPL.html)



# Solvers @ NEOS

4 NEOS Solvers for Mixed Integer Nonlinearly Constrained Optimization Problems:

- Bonmin – [www.neos-server.org/neos/solvers/minco:Bonmin/AMPL.html](http://www.neos-server.org/neos/solvers/minco:Bonmin/AMPL.html)
- Couenne – [www.neos-server.org/neos/solvers/minco:Couenne/AMPL.html](http://www.neos-server.org/neos/solvers/minco:Couenne/AMPL.html)
- FilmINT – [www.neos-server.org/neos/solvers/minco:FilmINT/AMPL.html](http://www.neos-server.org/neos/solvers/minco:FilmINT/AMPL.html)
- KNITRO – [www.neos-server.org/neos/solvers/minco:KNITRO/AMPL.html](http://www.neos-server.org/neos/solvers/minco:KNITRO/AMPL.html)
- MINLP – [www.neos-server.org/neos/solvers/minco:MINLP/AMPL.html](http://www.neos-server.org/neos/solvers/minco:MINLP/AMPL.html)
- scip – [www.neos-server.org/neos/solvers/minco:scip/AMPL.html](http://www.neos-server.org/neos/solvers/minco:scip/AMPL.html)



# Solvers @ NEOS

5 NEOS Solvers for Global Optimization Problems:

- ASA – [www.neos-server.org/neos/solvers/go:ASA/AMPL.html](http://www.neos-server.org/neos/solvers/go:ASA/AMPL.html)
- Couenne – [www.neos-server.org/neos/solvers/go:Couenne/AMPL.html](http://www.neos-server.org/neos/solvers/go:Couenne/AMPL.html)
- icos – [www.neos-server.org/neos/solvers/go:icos/AMPL.html](http://www.neos-server.org/neos/solvers/go:icos/AMPL.html)
- PGAPack – [www.neos-server.org/neos/solvers/go:PGAPack/AMPL.html](http://www.neos-server.org/neos/solvers/go:PGAPack/AMPL.html)
- PSwarm – [www.neos-server.org/neos/solvers/go:PSwarm/AMPL.html](http://www.neos-server.org/neos/solvers/go:PSwarm/AMPL.html)
- scip – [www.neos-server.org/neos/solvers/go:scip/AMPL.html](http://www.neos-server.org/neos/solvers/go:scip/AMPL.html)


Non-convex Portfolio Optimization Problems can be handled by this group of solvers offering several different types of heuristics.







# MSCI Barra Open Optimizer

 **Built for Portfolio Management** — The Barra Open Optimizer interface is designed specifically for portfolio managers and its algorithms are tuned for portfolio management challenges. It incorporates proprietary solvers developed by MSCI's optimization research team and solvers created by leading optimization experts.

 **Flexible Integration** — An intuitive programming API, available in **C++, Java™, C#, and COM**, provides easy integration with most libraries within statistical tools such as **MATLAB™, R, and SAS™**. In addition, an **XML and Protobuf** interface allows flexible creation and management of optimization data and parameters regardless of programming language. Documentation and working sample code are included to help accelerate integration time.

 **Multiple solvers in a single engine** — The Barra Open Optimizer utilizes multiple optimization solvers from MSCI and 3rd parties and delivers a single interface to our clients. Some portfolio management problems and strategies require a specialized solver to deliver an economically meaningful solution. We eliminate the need for our clients to implement multiple solutions.

 **Create Investable Optimal Portfolios** — Barra Open Optimizer moves beyond mean-variance optimization through support for advanced mandates and alternative portfolio construction techniques. Constraint-aware roundlotting ensures the portfolio rules are satisfied while creating round lots and include threshold constraints. Other features include **risk parity portfolio construction** (also known as equal risk weighting) and **transaction cost control** through fixed costs, thresholds, and maximums on the number of names.

 **Transparency in Optimization** — By providing constraint shadow costs reports, solution introspection and frontier analysis, Barra Open Optimizer provides users with more transparency and intuition around optimization results.

» **Industry Acceptance** — The Barra Optimizer engine powers Barra Aegis, BarraOne, and Barra Portfolio Manager, which are used by a wide range of institutional investors, including the MSCI Global Minimum Volatility Indexes.

Note, the Barra Open Optimizer Library can also be linked to the Portfolio Dashboard

The Software Pilot will offer a large set of major Key Features from Barra's Open Optimizer.

Even more, the Rmetrics Dashboard offers a GUI, an Expert Editor, Visualization and Reporting Tools together with the full mathematical and statistical tool set from R and Rmetrics.

## Key Features

- » Modern portfolio construction controls
- » **Risk parity** portfolio construction
- » Minimize small trades or positions with precise control on **thresholds**
- » Set a **maximum on the number of assets** help at portfolio or at group level
- » Enforce trades to round lots either during or after optimization
- » Penalizes the residual alpha in optimization to **correct alpha and risk factor misalignment**
- » Limit the number of trades/longs/shorts/buys/sells
- » Apply fixed transaction costs per trade in addition to piecewise-linear and non-linear transaction costs, specific to each asset
- » Provides soft bounds and constraints on leverage, risk, roundlotting, and other settings
- » Create a more flexible constraint hierarchy with the ability to set priority levels for factor constraints
- » **Enhance long/short optimization** with new leverage constraints, roundlotting, additional paring constraints, and non-convex risk constraints
- » Form efficient frontiers over tracking error, turnover, transaction costs, or other constraints
- » Add multiple risk terms to the objective function
- » Prioritize soft bounds and constraints to **increase likelihood of feasibility**
- » Control the optimality tolerance
- » Optimize dual risk models and multiple benchmarks
- » **Maximize the information ratio** and/or Sharpe ratio
- » Fully parameterized mean-variance utility function
- » Control the maximum time limit for an optimization that have reached feasibility but not yet converged to optimality

## Index Tracking

- » Bounds on total or active risk at portfolio or sub-group level
- » Maximum limit on piecewise-linear transaction costs

- » Constraints on maximum and/or minimum number of assets, trades, and more

## Long-Short Hedging Basket Creation

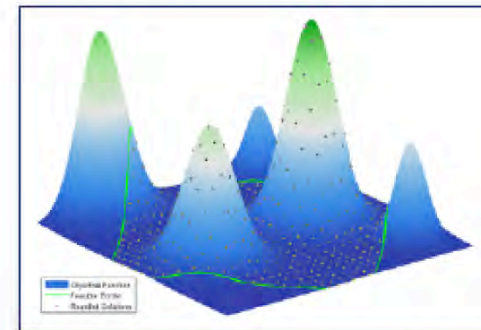
- » Constraints on leverage for longs, shorts, and turnover that can be defined independently by side
- » Limit on buy turnover or sell turnover
- » Paring constraints on holding or transaction levels on long or short side
- » Lower bounds on long or short groups even when the problem becomes non-convex
- » Targets and constraints on risk
- » Modeling of short costs

## Tax-Aware Optimization

- » Bounds on long-and short-term gross gains or losses for tax arbitrage
- » Tax lotting, with FIFO, LIFO, and FIFO trading rules
- » Multiple options for handling wash sales

## Asset Allocation

- » Risk parity portfolio construction
- » Utilizes an asset-by-asset covariance matrix
- » Includes futures, ETFs, currencies, and other alternative assets





Barra Open Optimizer can form portfolios under complex considerations. Here is a visualization of a potential optimization problem with risk aversion, maximum on the number of names, asset and factor constraints, and the requirement to trade round lots. The peaks represent local solutions, and the green lines demarcate the feasible region. Diamonds represent portfolios formed out of round lots.




## 5 Post-Processing





 Choose Application


 Access Data Base

 Run Analytics

 Plot Charts

 Print Reports


 Search Help

 About


# List of Plot Charts

We have implemented the following *Plot Charts*:


- Asset Wealth (of Portfolio Components)
- Portfolio Weights Pie, Bar and Line Plots
- Portfolio Wealth and Volatility Plots
- Portfolio Drawdowns and Recovery Times Plots
- Portfolio Turning Points and Cycles Analytics
- Portfolio Stability by Bayesian Change Points Analytics
- Portfolio Outlier Detection by Principal Component Analytics
- Portfolio Wavelet Spectrum to detect Nonstationarities
- Portfolio Efficient Frontier and Critical Lines Plots
- Portfolio Markowitz and Pareto Hull Plots
- Portfolio Feasible Set Plots


 Choose Application


 Access Data Base

 Run Analytics

 Plot Charts

 Print Reports


 Search Help

 About


# List of Print Reports


Print Reports allow to print results from the Portfolio Dashboard into PDF formatted Documents:


- Portfolio Scenarios for all Components
- Portfolio Specifications from model, data, and run Files
- Portfolio Weights
- Portfolio Risk Budgets
- Portfolio Solver Info from Optimization
  
- Listing of all Available Portfolio Designs
- Listing of all Available Math Programming Solvers


 Choose Application

 Access Data Base

 Run Analytics

 Plot Charts

 Print Reports

 Search Help

 About

# Search Help

We have implemented a Search for Help. With the help tool you can access several kinds of documentations

- General R and Rmetrics Help Pages and Vignettes
- AMPL Documentation
- Mathematical Programming Solvers Documentation



## 6 Some Additional Tools



# Dashboard Configuration Tools

The software pilot is stored in a directory named *RmetricsDashboard*. This directory contains the subdirectories

- DASHBOARD - with the dashboard configuration information
- ARCHIVE - with the portfolio data files
- MANUAL - with all help documents
- BATCH - comes with batch mode examples
- SRC - with the R/Rmetrics and third party software

The startup file named *startDashboard.R* is also saved in the *RmetricsDashboard* main directory.





# Getting Started

Once the Dashboard has been started, hidden files are created that keep the status of the dashboard. Whenever the Dashboard is re-started, it starts with the full information from the previous status.

Portfolio calculations are archived through a stack-like mechanism, so that previously optimized portfolios can be reloaded.

The Portfolio Dashboard is fully configurable. Configuration files can be modified and tailored by the user taking personal preferences into account. These files are

- the general *config* file with path and application information,
- the *symbols* file with the names of all portfolio data files,
- the *graphics* file with all R graphical functions names,
- the *tables* files with all R report function names, and
- the *runapp* file for the Ace editor configuration.

Through this new functionality can be very easily added.



# Command Line Batch Processing

The Portfolio Dashboard allows by default all portfolio designs through a Graphical User Interface.

- All portfolios can also be designed and processed from the R *Command Line Batch Mode*.
- The subdirectory BATCH of the Portfolio Dashboard Application holds several *Examples* for AMPL model files and batch model scripts.

The command line batch processing mode is optional.



# Command Line Batch Mode Processing

```
runShiny.R * batchPortfolio.R* * portfolio-Solver.R * portfolio-Library.R *
Source on Save Run Source
214 # BATCH MODE:
215
216 # Load Library:
217 require(appRmetricsDashboard)
218
219 # Load and Save Portfolio Scenarios:
220 series <- LPP2005REC[, 1:6]
221 seriesName <- "LPP2005"
222 .scenarios <- list(series=series, seriesName=seriesName)
223 save(.scenarios, file=".scenarios.RData")
224
225 # Optimize Feasible Set Portfolio:
226 .pfolio <- solveDefaultPortfolio(model=modelMV1FS, solver="Ipop")
227
228 # Plot Charts - Automated Portfolio Type Detection:
229 .optimizeAssetWealthsPlot()
230 .optimizeWeightsPlot()
231 .optimizeWealthPlot()
232 .optimizeDrawdownsPlot()
233 .optimizeTurningPointsPlot()
234 .optimizeStabilityPlot()
235 .optimizeOutliersPlot()
236 .optimizeSpectrumPlot()
237
238 # Print Reports - Show all Available Models and Solvers:
239 .optimizeAvailableModelsPrint()
240 .optimizeAvailableSolversPrint()
241
242 # Print Reports - Optimized Portfolio Info:
243 .optimizePortfolioScenariosPrint()
244 .optimizePortfolioSpecificationsPrint()
245 .optimizePortfolioWeightsPrint()
246 .optimizePortfolioSolverInfoPrint()
247
248
248:1 (Untitled) R Script
```

# Selected References

Rmetrics: [www.rmetrics.org](http://www.rmetrics.org)

This is the home of the Rmetrics Open Source Project R-in-Finance

R Cran: [www.r-project.org](http://www.r-project.org)

This is the home for the R Statistical Computing Project.

R Studio: [www.rstudio.com](http://www.rstudio.com)

This links to the R Studio GUI and R Shiny Web Application

Rmetrics Working Papers: [www.rmetrics.org/WorkingPapers](http://www.rmetrics.org/WorkingPapers)

This link lists Rmetrics' working papers and white papers and allows to download them

Rmetrics Workshop and Trainings: [www.rmetrics.org/LecturesCourses](http://www.rmetrics.org/LecturesCourses)

This link lists all Rmetrics Workshops, Lectures, Master Classes, Trainings and Tutorials

Rmetrics Villa Hatt: [www.rmetrics.org/zurich2015](http://www.rmetrics.org/zurich2015)

This is a link to the most recent Villa Hatt Meeting and Master Class 2015 Zurich

Rmetrics R-forge Repository: [r-forge.r-project.org/projects/rmetrics](http://r-forge.r-project.org/projects/rmetrics)

This is the link to the Rmetrics Repository at R-forge for package development



# Thanks

Research & Development  
Realization & Design  
Implementation

Econophysics Group ETH Zurich  
Rmetrics Association Zurich  
Rmetrics Juniors Zurich

Maintenance

Rmetrics Association Zurich

Team

Professor Diethelm Würtz  
PhD cand. Tobias Setz  
Tomas Tamfal (Bachelor Thesis Project)  
Henry Claussen (Master Thesis Project)  
Daniel Mertsen (Bachelor Thesis Project)  
Tobias Hatt (Semester Thesis Project)