

Risk valuation using ProActive workflows



Value At Risk (VaR)

One-day 95% VaR of \$1 million \Leftrightarrow 5% probability that the portfolio worst-case loss will exceed \$1 million over a one-day period

Monte Carlo (MC) simulations

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{n \rightarrow +\infty} \mathbb{E} X$$

The Law of Large Numbers states for large n , the empirical average is very close to the expected value

Estimating the VaR using MC simulations

$$\mathbb{P}(L_t > x) = \frac{1}{nb_VaR} \sum_{i=1}^{nb_VaR} \mathbb{1}_{L_t^i > x} = 1 - \alpha$$

x the portfolio VaR

L_t the portfolio loss at t

α the VaR probability

nb_VaR the number of MC simulations

Estimates the Monte Carlo Value at Risk (MC VaR) of a portfolio. We use the geometric Brownian motion (GBM) method to simulate stock price paths, but more advanced assets can be integrated thanks to the Quantlib C++ lib.

nbMC

10000

nbTimeSteps

255

horizon

1

confidenceRate

0.99

nbBins

50

assets_params_file_path_from_datasp...

assets.txt

assets_correlations_file_path_from_dat...

assets_correlations.txt

nbTasks

5

width_chart

1000

height_chart

500

number of portfolio simulations. The number of tasks does not necessary divide the total number of simulations

number of time steps of the portfolio simulations paths, up to the horizon

horizon. Here time steps are set to 1/255

VaR confidence rate. Here 99%

number of bins to plot the frequencies bar chart

file to describe assets params, a line per asset: start price, drift rate (yearly return), volatility rate (yearly return), weight

correlations matrix of the assets

number of replicated tasks to parallelize the MC simulations

width of the generated chart

height of the generated chart

20.16,0.2312,0.2116,0.3

40,0.115,0.254,0.3

110,0.6,0.4,0.4

1 0 0

0 1 0

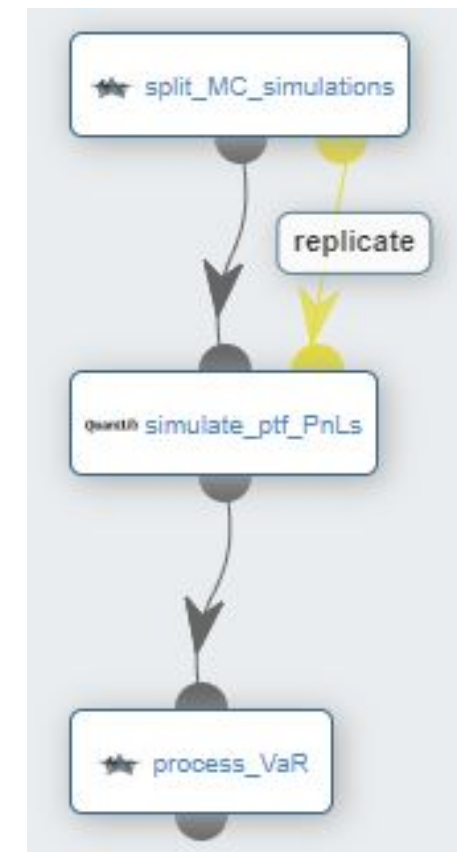
0 0 1



Estimate the number of MC simulations per replicated task. The tasks number does not necessary divide the total number of simulations

Using Quanlib, each replicated task proceses a subset of the MC simulations and deduces the PnL (profit and loss) of each simulated path (value at horizon of a simulation - value at start). On the task side, PnLs are saved into a dedicated file

This task gathers all the PnLs into a single array, sorts them, and retrieves the VaR at the VaR_index corresponding to $(1 - \text{confidenceRate}) * \text{nbMC}$. Finally, it generates the corresponding frequencies bar chart and exposes it (view/download)



▼ Details

Tasks

Visualization

Users Sessions

Statistics

Usage

Tag

Filter

Auto-refresh

	Id	Status	Name	Tag	Duration	Nodes	Executions	N
▶	0	Finished	Split_MC_simulations		1s 668ms	1	0 / 2	
▶	1	Finished	simulate_ptf_PnLs		4s 800ms	1	0 / 2	
▶	2	Finished	process_VaR		3s 66ms	1	0 / 2	
▶	3	Finished	simulate_ptf_PnLs*1	REPLICATE-Split_M...	4s 562ms	1	0 / 2	
▶	4	Finished	simulate_ptf_PnLs*2	REPLICATE-Split_M...	4s 246ms	1	0 / 2	
▶	5	Finished	simulate_ptf_PnLs*3	REPLICATE-Split_M...	4s 720ms	1	0 / 2	
▶	6	Finished	simulate_ptf_PnLs*4	REPLICATE-Split_M...	3s 6ms	1	0 / 2	

<< First < Previous Page 1 of 1 Tasks 1 - 7 Next > Last >>

Job Info

Task Info

Output

Server Logs

Preview

Remote Visualization

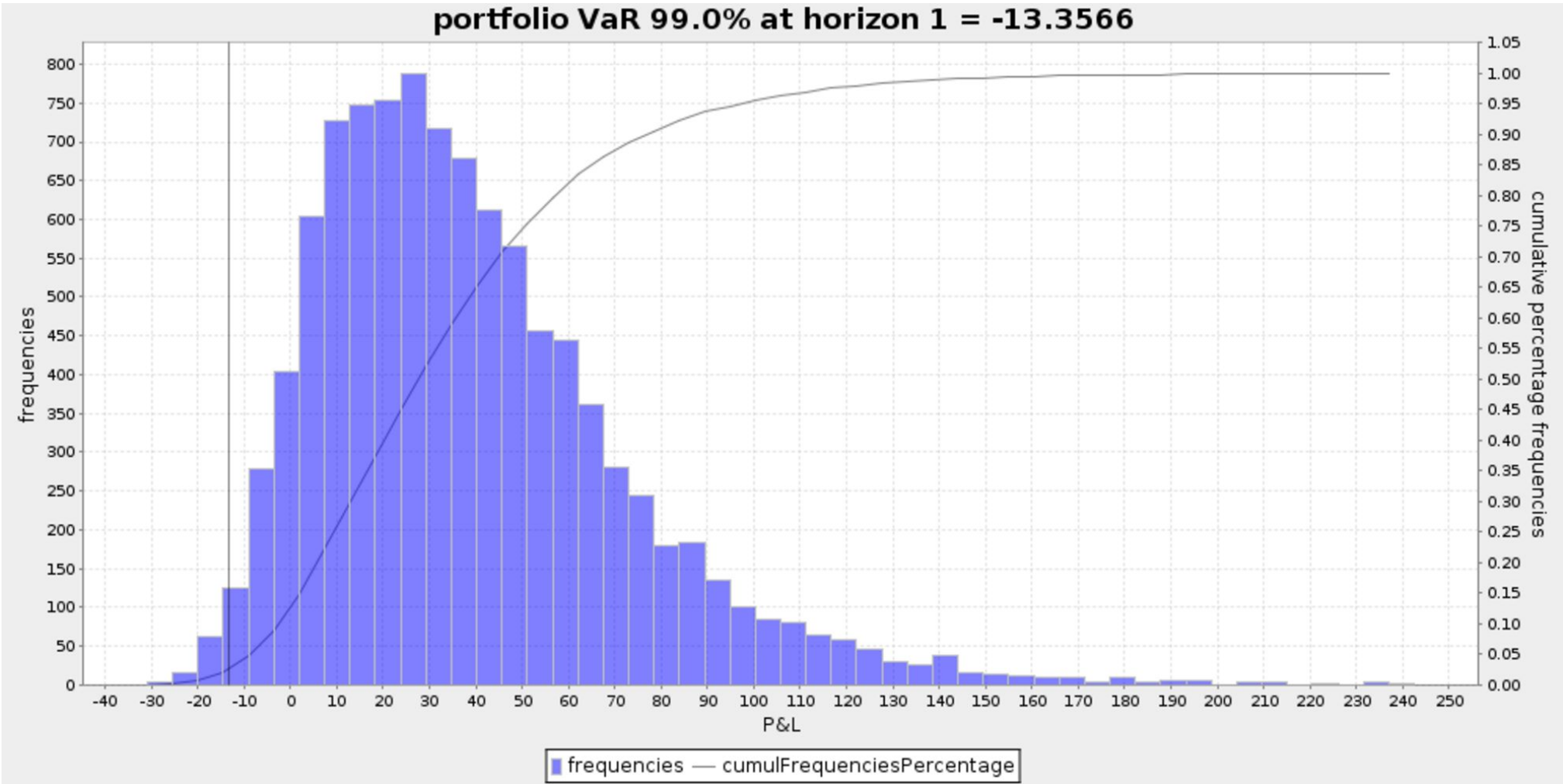
Remote visualization is disabled. Please toggle streaming in output view for a job in order to enable the remote visualization.

Task Result

Task process_VaR (id: 2) from job MCVaR (id: 1)

Open in browser

Save as file





Estimates the incremental VaR (iVaR) for each asset of the portfolio. iVaR quantifies the risk a position (or sub-portfolio) is adding to a portfolio. For instance, the iVaR related to an asset Y, is the difference between the portfolio VaR with and without Y.

nbMC	10000
nbTimeSteps	255
horizon	1
confidenceRate	0.99
nbBins	50
assets_params_file_path_from_datasp...	assets.txt
assets_correlations_file_path_from_dat...	assets_correlations.txt
nbTasksPerVaR	4
width_chart	1000
height_chart	500

number of portfolio simulations. The number of tasks does not necessary divide the total number of simulations

number of time steps of the portfolio simulations paths, up to the horizon

horizon. Here time steps are set to 1/255

VaR confidence rate. Here 99%

number of bins to plot the frequencies bar chart

file to describe assets params, a line per asset: start price, drift rate (yearly return), volatility rate (yearly return), weight

correlations matrix of the assets

number of replicated tasks per VaR to parallelize the MC simulations

width of the generated chart

height of the generated chart



Estimate the number of MC simulations per replicated task for each VaR (right branch and left branch). The tasks number must divide the total number of simulations

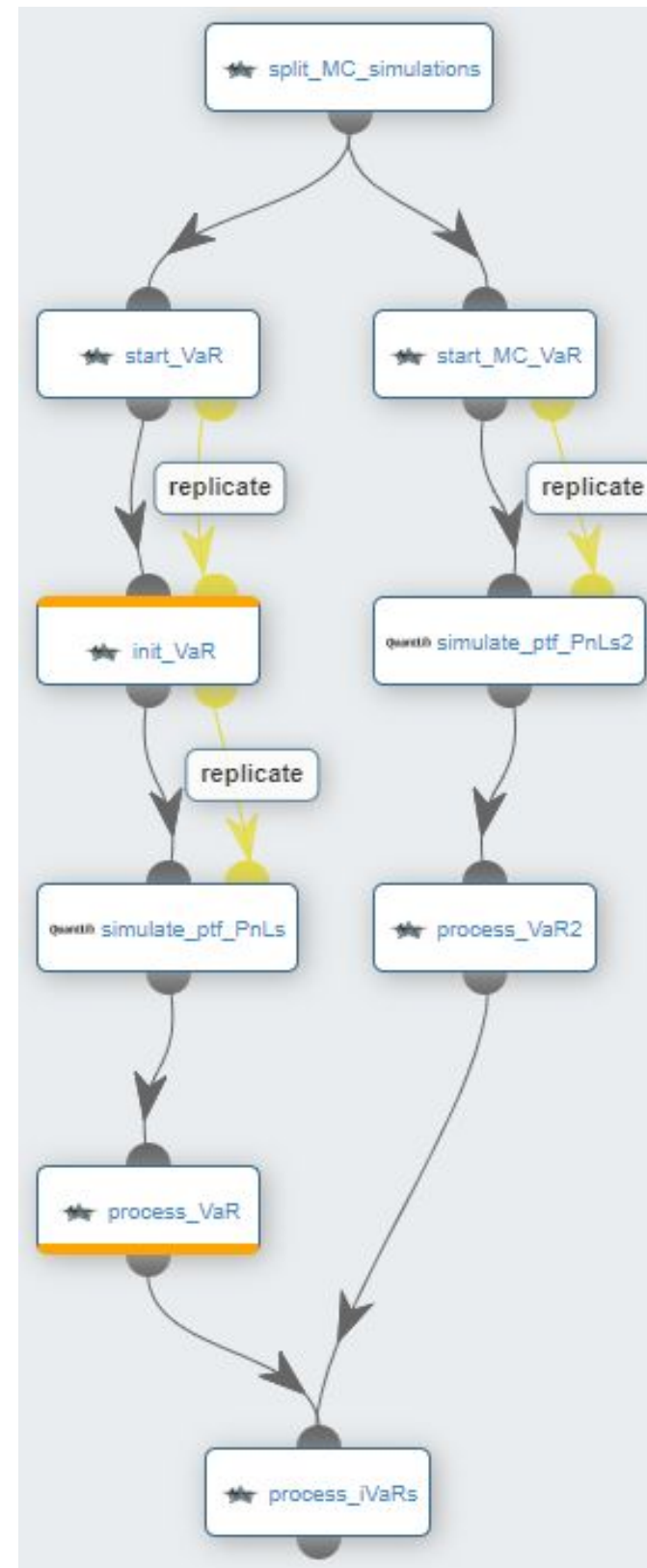
1st level of replicated tasks: a replicated task per asset (Y)

Each replicated task instantiates an assets params files, by setting to 0 its corresponding asset. By this way, the estimated VaR will not consider the asset

2nd level of replicated tasks: a replicated task per subset of MC simulations. Each replicated task processes a subset of the MC simulations according to its instantiated assets params file (i.e. instantiated by its father replicated task)

1st level of replicated tasks: a replicated task to merge the MC simulations processed by its sons task. Each task estimates the VaR related to Y, stores it, and generates the corresponding frequencies bar chart

Compute and printIn the iVaR related to each asset (portfolio VaR estimated by the right branch - portfolio VaR without the asset Y estimated by the left branch), merge all the frequencies bar charts into a single png file and exposes it



Perform the MC VaR like
Monte_Carlo_VaR_portfolio.xml



Details

Tasks
Visualization
Users Sessions
Statistics
Usage

Tag

Filter

Auto-refresh

	Id	Status	Name ^	Tag	Duration	Nodes	Executions	Node Failures	Visu
	1	Finished	Init_VaR		2s 565ms	1	0 / 2	0 / 2	
	9	Finished	Init_VaR*1	REPLICATE-Start_V...	1s 873ms	1	0 / 2	0 / 2	
	12	Finished	Init_VaR*2	REPLICATE-Start_V...	2s 52ms	1	0 / 2	0 / 2	
	4	Finished	process_iVaRs		2s 928ms	1	0 / 2	0 / 2	
	3	Finished	process_VaR		3s 721ms	1	0 / 2	0 / 2	
	11	Finished	process_VaR*1	REPLICATE-Start_V...	3s 14ms	1	0 / 2	0 / 2	
	14	Finished	process_VaR*2	REPLICATE-Start_V...	2s 890ms	1	0 / 2	0 / 2	
	7	Finished	process_VaR2		3s 361ms	1	0 / 2	0 / 2	
	2	Finished	simulate_ptf_PnLs		4s 503ms	1	0 / 2	0 / 2	
	10	Finished	simulate_ptf_PnLs*1	REPLICATE-Start_V...	3s 895ms	1	0 / 2	0 / 2	
	13	Finished	simulate_ptf_PnLs*2	REPLICATE-Start_V...	3s 622ms	1	0 / 2	0 / 2	
	18	Finished	simulate_ptf_PnLs*3	REPLICATE-Init_VaR-3	3s 549ms	1	0 / 2	0 / 2	

<< First < Previous

Page 1 of 2

Tasks 1 - 20

Next > Last >>

Job Info Task Info Output Server Logs Preview

Remote Visualization

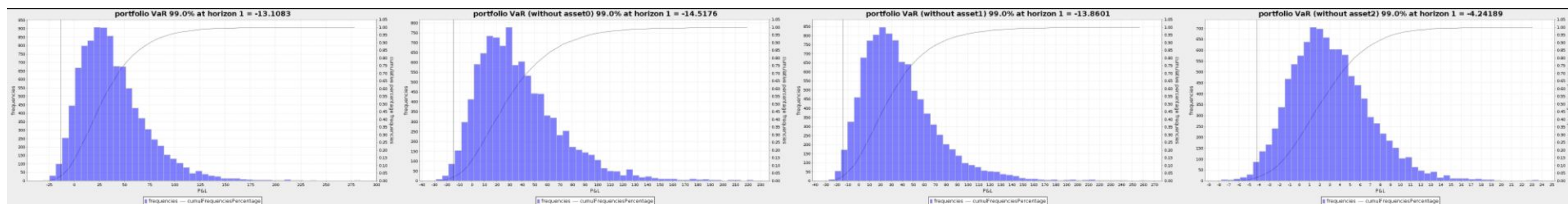
Remote visualization is disabled. Please toggle streaming in output view for a job in order to enable the remote visualization.

Task Result

Task process_iVaRs (id: 4) from job IncrementalVaRforEachPortfolioAsset-iVaR (id: 154)

Open in browser

Save as file



Estimates the portfolio PnL (Profit and Loss) over stressed volatilities and risk free rates.

DOCKER_ENABLED	<input checked="" type="radio"/> TRUE <input type="radio"/> FALSE	run the plotting task into a docker container for matplotlib support
nb_MC_per_ptf_value	<input type="text" value="10000"/>	number of portfolio simulations per PnL estimation
nb_time_steps	<input type="text" value="255"/>	number of time steps of the portfolio simulations paths, up to the horizon
stress_horizon	<input type="text" value="1"/>	horizon until which portfolio is stressed (here time steps are set to 1/255)
assets_params_file_path_from_datasp...	<input type="text" value="assets.txt"/>	file to describe assets params, a line per asset: start price, drift rate (yearly return), volatility rate (yearly return), weight
assets_correlations_file_path_from_dat...	<input type="text" value="assets_correlations.txt"/>	correlations matrix of the assets
stressed_risk_free_rate_min_max_in_p...	<input type="text" value="-10%,+10%,10"/>	range of the percentages of variation to apply to the risk free rate and number of steps to consider in this range
stressed_volatility_min_max_in_percent...	<input type="text" value="-5%,+5%,10"/>	range of the percentages of variation to apply to the portfolio asset volatilities and number of steps to consider in this range
nb_replicated_tasks	<input type="text" value="4"/>	number of replicated tasks which have in charge the stressed PnL simulations

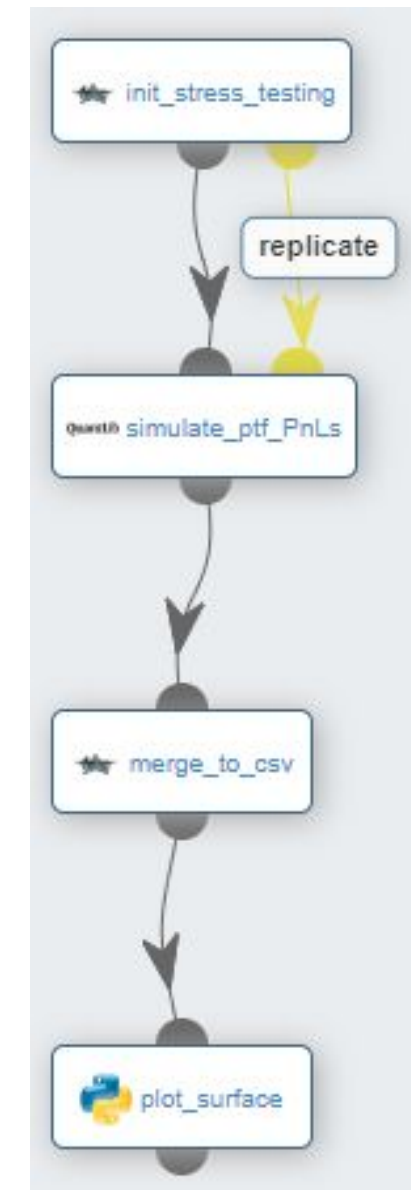


Estimate the number of MC simulations per replicated task. The tasks number must divide the total number of simulations

Using Quanlib, each replicated task processes a subset of the MC simulations and deduces the PnL (profit and loss) of each simulated path. An expected PnL is estimated per risk free rate and volatility stressed percentage. On the task side, PnLs are saved into a dedicated file

Merge all stressed PnLs into a single csv file

Plot a 3D representation of the stressed PnLs



Details

Tasks

Visualization

Users Sessions

Statistics

Usage

Tag

Filter

☒ Auto-refresh

	Id	Status	Name	Tag	Duration	Nodes	Executions	Node Failures	Visu
▶	0	Finished	plot_surface		5s 2ms	1	0 / 2	0 / 2	
▶	1	Finished	init_stress_testing		2s 43ms	1	0 / 2	0 / 2	
▶	2	Finished	simulate_ptf_PnLs		2m 43s 788ms	1	0 / 2	0 / 2	
▶	3	Finished	merge_to_csv		1s 931ms	1	0 / 2	0 / 2	
▶	4	Finished	simulate_ptf_PnLs*1	REPLICATE-init_stre...	2m 43s 606ms	1	0 / 2	0 / 2	
▶	5	Finished	simulate_ptf_PnLs*2	REPLICATE-init_stre...	2m 42s 81ms	1	0 / 2	0 / 2	
▶	6	Finished	simulate_ptf_PnLs*3	REPLICATE-init_stre...	2m 43s 284ms	1	0 / 2	0 / 2	

Job Info

Task Info

Output

Server Logs

Preview

Remote Visualization

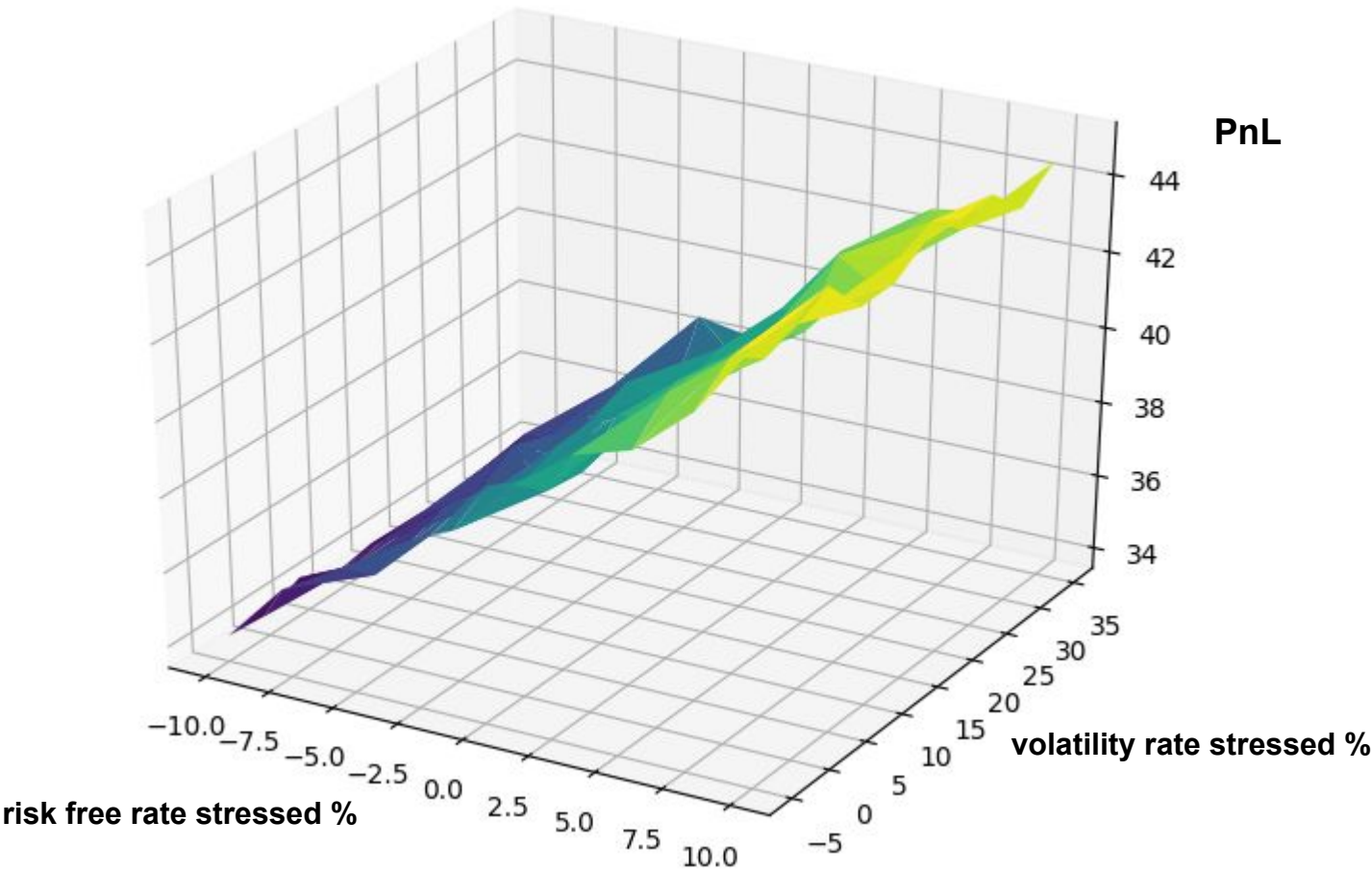
Remote visualization is disabled. Please toggle streaming in output view for a job in order to enable the remote visualization.

Task Result

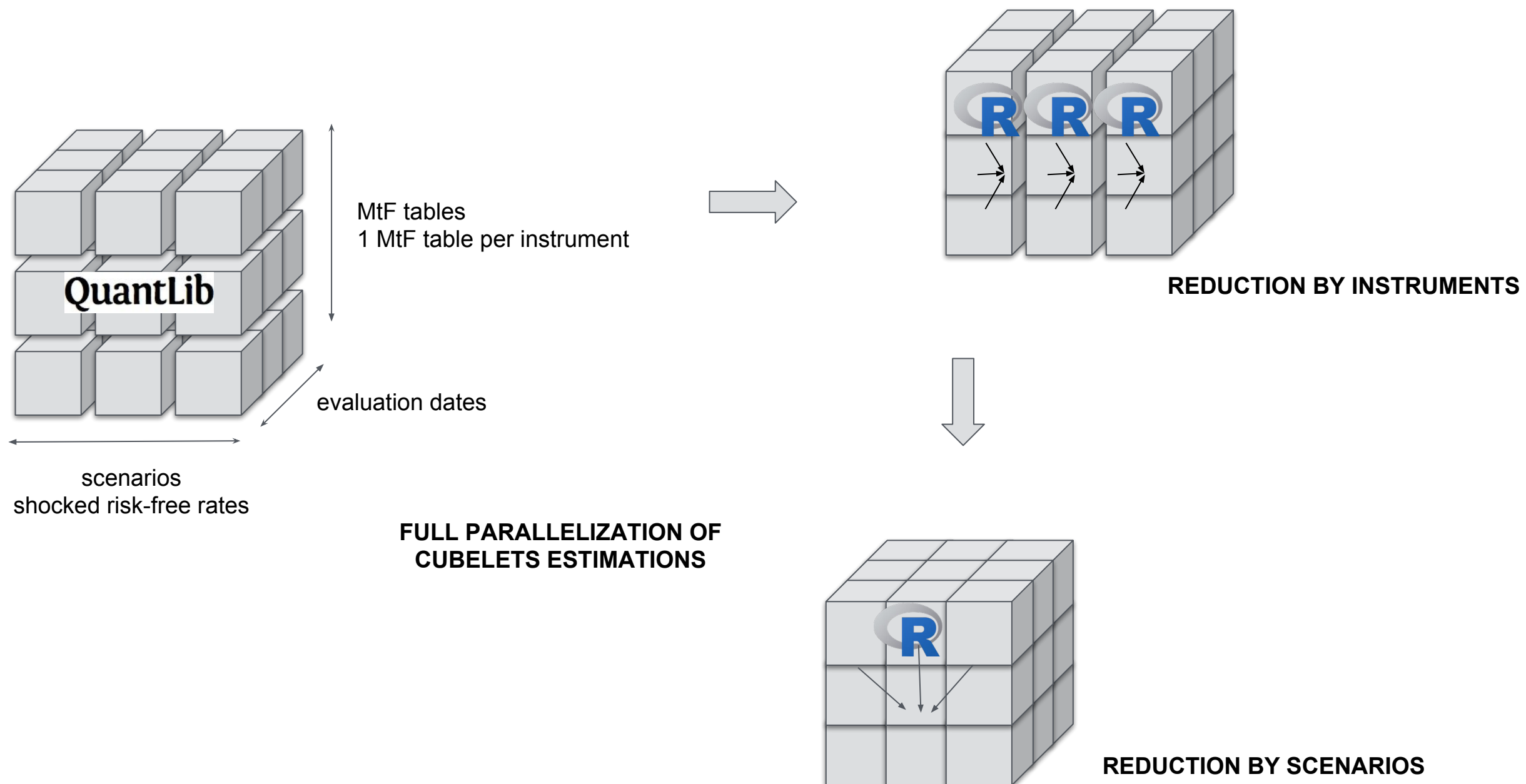
Task plot_surface (id: 0) from job stress_testing_Monte_Carlo_value_portfolio (id: 21)

Open in browser

Save as file



Estimates a Mark-to-Future (MtF) cube of a bond portfolio. Each cell of the cube integrates the valuation of a bond at a specific time given a specific scenario. This can be easily extended thanks to the high maintainability of the implementation (C++ Quantlib for the pricing engine, inputs split in Java/Groovy, R for the cube/cubelet stats,..).





evaluation_date_start	<input type="text" value="01/12/2018"/>	start date of the evaluations
evaluation_date_end	<input type="text" value="20/12/2018"/>	end date of the evaluations
evaluation_frequency	<input type="text" value="monthly"/>	evaluation frequency (monthly, biweekly)
shocked_yield_start	<input type="text" value="0"/>	shocked yield start
shocked_yield_end	<input type="text" value="0.1"/>	shocked yield end
shocked_yield_delta	<input type="text" value="0.005"/>	shocked yield delta
bonds_params_file_path_from_dataspacespace	<input type="text" value="bonds.txt"/>	file to describe bonds params, a line per bond: bond id, settlement days, face amount, coupon rate, redemption, issue date, maturity date, payment frequency
node_source_compute_memory_intens...	<input type="radio"/> TRUE <input checked="" type="radio"/> FALSE	boolean to either reserve compute and memory intensive nodes sources for compute and memory intensive tasks or not
node_source_compute_intensive	<input type="text" value="local"/>	compute intensive node source name
node_source_memory_intensive	<input type="text" value="local"/>	memory intensive node source name
data_dir_path	<input type="text" value="/tmp"/>	data dir path to store generated data before reductions (for large data)
nb_replicated_tasks	<input type="text" value="2"/>	number of replicated tasks to split scenarios
nb_replicated_tasks_per_cubelet	<input type="text" value="2"/>	number of replicated tasks to split bonds



Split the scenarios and bonds over the replicated tasks

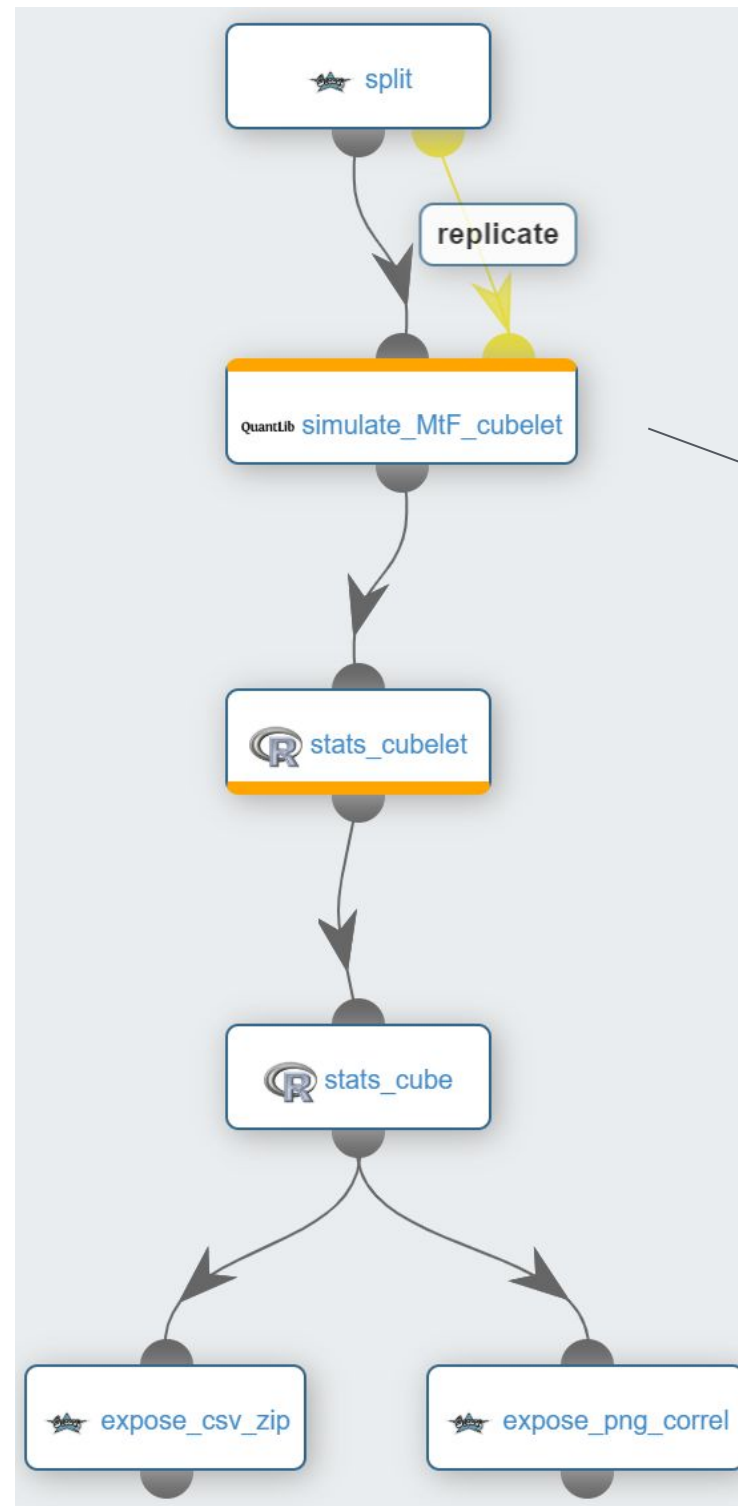
1st level of replicated tasks: a replicated task per subset of scenarios

Each replicated task pull from the catalog and submit the cubelet simulation workflow

Compute the portfolio clean prices per scenario and evaluation date (cubelets)

Merge the portfolio clean prices into a single csv file (cube). Compute portfolio prices time series for each scenario and create a csv file. Compute correlations over scenarios and create a csv file + heat map png file

Allow user to visualize/download the png/csv zip files

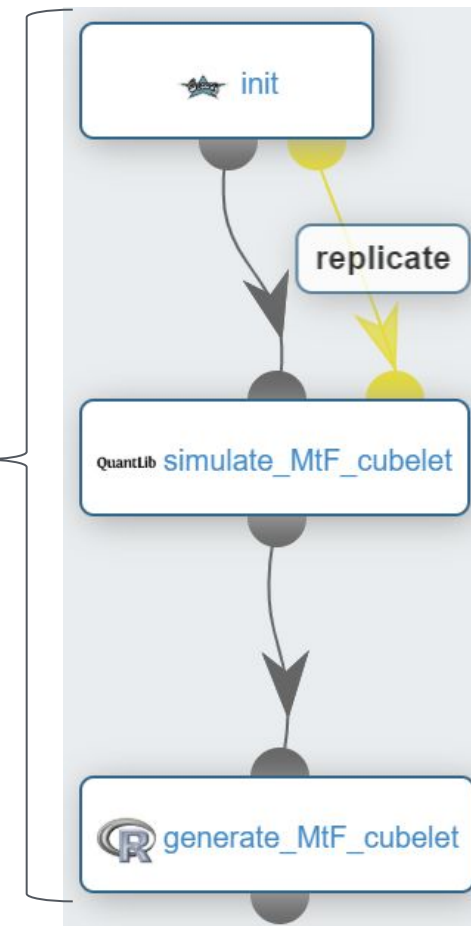


Init parameters

2nd level of replicated tasks: a replicated task per subset of bonds

Using Quanlib, each replicated task estimates a subset of the portfolio clean prices (cubelet)

Merge cubelets (bonds -> portfolio) and create a csv file





Details

Tasks Visualization Users Sessions Statistics Usage

Tag Filter ☒ Auto-refresh

	Id	Status	Name	Tag	Duration	Nodes	Executions
▶	0	Finished	split		2s 395ms	1	0 / 2
▶	1	Finished	simulate_MtF_cubelet		35s 107ms	1	1 / 2
▶	2	Finished	stats_cubelet		1s 304ms	1	0 / 2
▶	3	Finished	stats_cube		1s 842ms	1	0 / 2
▶	4	Finished	expose_csv_zip		2s 229ms	1	0 / 2
▶	5	Finished	expose_png_correl		1s 936ms	1	0 / 2
▶	6	Finished	simulate_MtF_cubelet...	REPLICATE-split-1	44s 471ms	1	1 / 2
▶	7	Finished	stats_cubelet*1	REPLICATE-split-1	1s 83ms	1	0 / 2

<< First < Previous Page 1 of 1 Tasks 1 - 8 Next > Last >>

Job Info Task Info Output Server Logs Preview

Remote Visualization

Remote visualization is disabled. Please toggle streaming in output view for a job in order to enable the remote visualization.

Task Result

Task expose_png_correl (id: 5) from job Mark_to_Future_cube_stats (id: 36)

Open in browser

Save as file

