Portfolio Allocation System

Arthur Guiot - EveryFinance

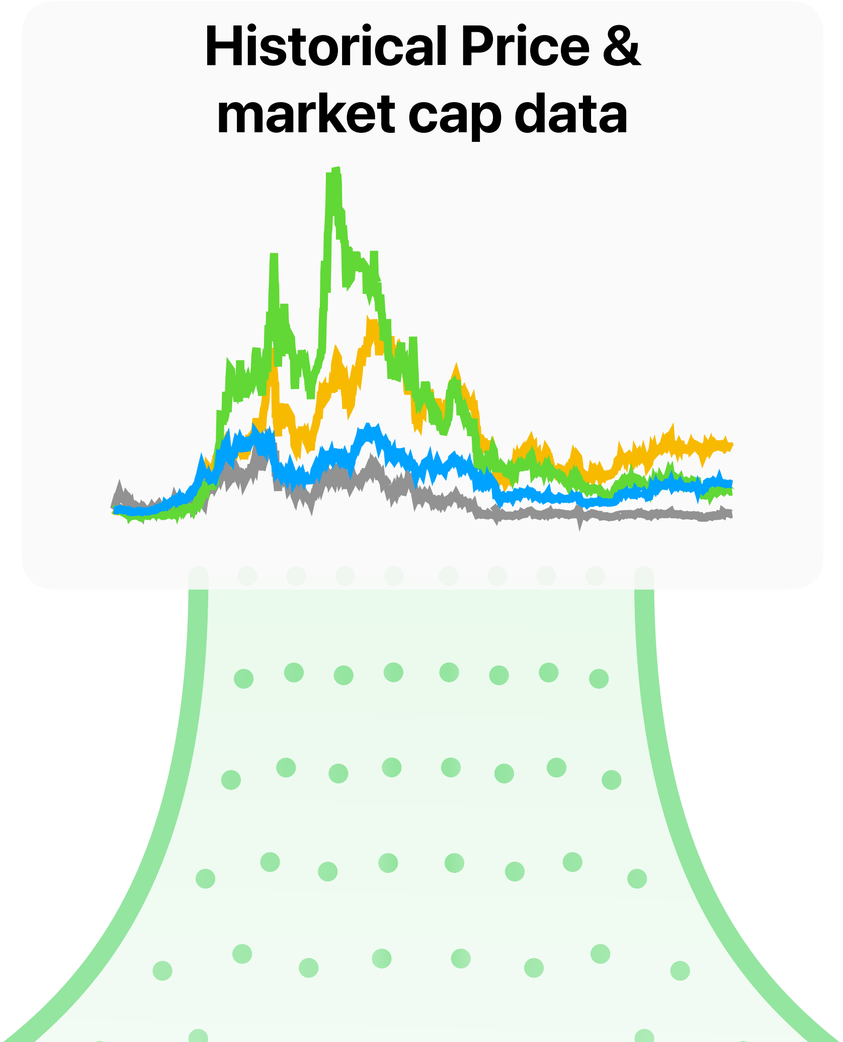
Backtesting

Portfolio
Risk Parity

Portfolio
Markowitz

Portfolio
HRP

Backtesting



Backtesting

Custom allocation

👉 **Parity Line**

Every Day/Week/Month

👉 **Rebalance**

Maximum Weights, yield, ...

👉 **Contraints**

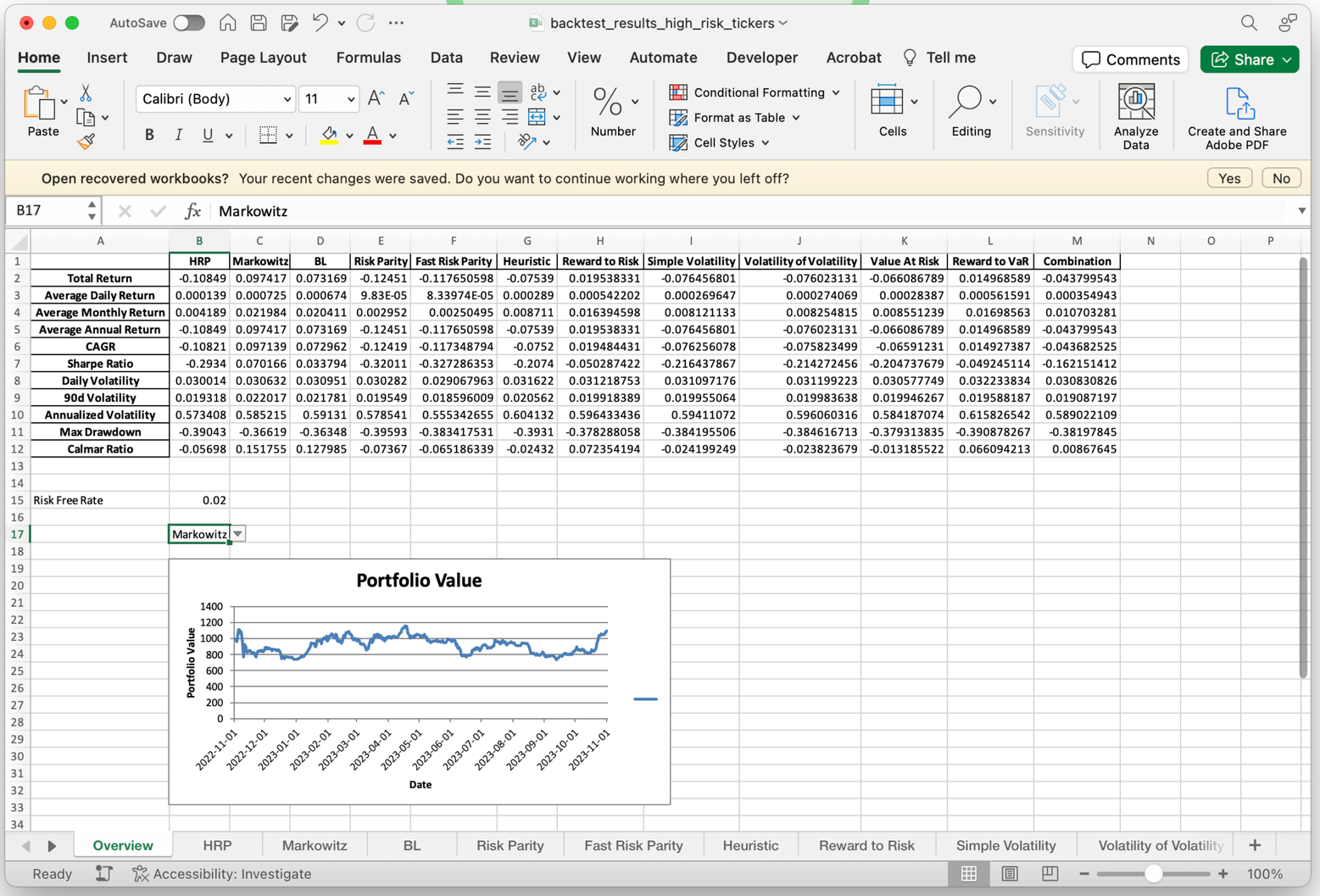
Portfolio

Risk Parity

Port

Marko

Results

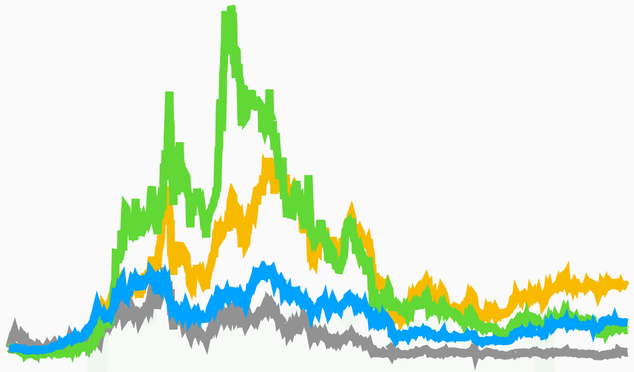


	HRP	Markowitz	BL	Risk Parity	Fast Risk Parity	Heuristic	Reward to Risk	Simple Volatility	Volatility of Volatility	Value At Risk	Reward to VaR	Combination
Total Return	-0.10849	0.097417	0.073169	-0.12451	-0.117650598	-0.07539	0.019538331	-0.076456801	-0.076023131	-0.066086789	0.014968589	-0.043799543
Average Daily Return	0.000139	0.000725	0.000674	9.83E-05	8.33974E-05	0.000289	0.000542202	0.000269647	0.000274069	0.00028387	0.000561591	0.000354943
Average Monthly Return	0.004189	0.021984	0.020411	0.002952	0.00250495	0.008711	0.016394598	0.008121133	0.008254815	0.008551239	0.01698563	0.010703281
Average Annual Return	-0.10849	0.097417	0.073169	-0.12451	-0.117650598	-0.07539	0.019538331	-0.076456801	-0.076023131	-0.066086789	0.014968589	-0.043799543
CAGR	-0.10821	0.097139	0.072962	-0.12419	-0.117348794	-0.0752	0.019484431	-0.076256078	-0.075823499	-0.06591231	0.014927387	-0.043682525
Sharpe Ratio	-0.2934	0.070166	0.033794	-0.32011	-0.327286353	-0.2074	-0.050287422	-0.216437867	-0.214272456	-0.204737679	-0.049245114	-0.162151412
Daily Volatility	0.030014	0.030632	0.030951	0.030282	0.029067963	0.031622	0.031218753	0.031097176	0.031199223	0.030577749	0.032233834	0.030830826
90d Volatility	0.019318	0.022017	0.021781	0.019549	0.018596009	0.020562	0.019918389	0.019955064	0.019983638	0.019946267	0.019588187	0.019087197
Annualized Volatility	0.573408	0.585215	0.59131	0.578541	0.555342655	0.604132	0.596433436	0.59411072	0.596060316	0.584187074	0.615826542	0.589022109
Max Drawdown	-0.39043	-0.36619	-0.36348	-0.39593	-0.383417531	-0.3931	-0.378288058	-0.384195506	-0.384616713	-0.379313835	-0.390878267	-0.38197845
Calmar Ratio	-0.05698	0.151755	0.127985	-0.07367	-0.065186339	-0.02432	0.072354194	-0.024199249	-0.023823679	-0.013185522	0.066094213	0.00867645

Running a simulation

Using the portfolio optimisation package

Historical Price & market cap data



```python
# Choose the asset class and the assets
asset_list = get_tickers()
asset_class = "high_risk_tickers"

_df = get_historical_prices_for_assets(
    asset_list[asset_class],
    time_range=timedelta(days=365 * 1 + 120),  # 1 years
    interested_columns=["ReferenceRate", "CapMrktEstUSD"],
)

# Filter out all columns containing `_` in their name
df = _df.loc[:, ~_df.columns.str.contains("_")]
```

Running a simulation

Using the portfolio optimisation package

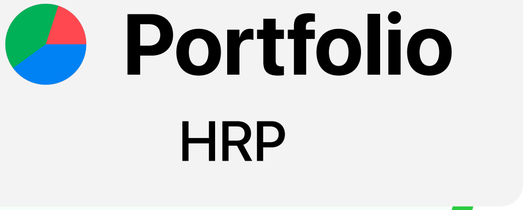
rtfolio

owitz

Portfolio

HRP

```
portfolio_hrp = Portfolio(
    base_value=initial_bid,
    initial_prices=df.loc[:start_date_portfolio],
    optimiser=HRPOptimization,
    max_weight=max_weight,
    min_weight=min_weight,
    weight_threshold=weight_threshold,
)
```




```python
portfolio_hrp = Portfolio(
    base_value=initial_bid,
    initial_prices=df.loc[:start_date_portfolio],
    optimiser=HRPOptimization,
    max_weight=max_weight,
    min_weight=min_weight,
    weight_threshold=weight_threshold,
)
```

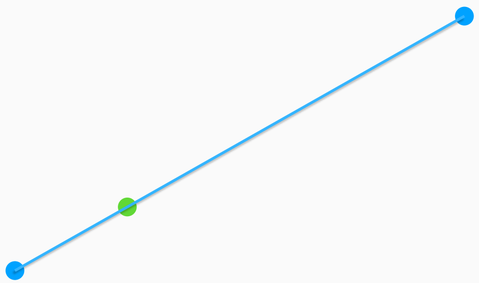
+

```python
class OptRebalancingPortfolioDelegate(PortfolioDelegate):
    def rebalance(
        self, holdings: pd.Series, prices: pd.Series, target_weights: pd.Series
    ) -> pd.Series:
        diff = optimize_trades(
            holdings=holdings,
            new_target_weights=target_weights,
            prices=prices,
            min_W=0.0,
            max_W=1.0,
            external_movement=0,
        )

        new_holdings = pd.Series(diff, index=holdings.index) + holdings
        return new_holdings

portfolio_hrp.delegate = OptRebalancingPortfolioDelegate()
```

Running a simulation

Using the portfolio optimisation package

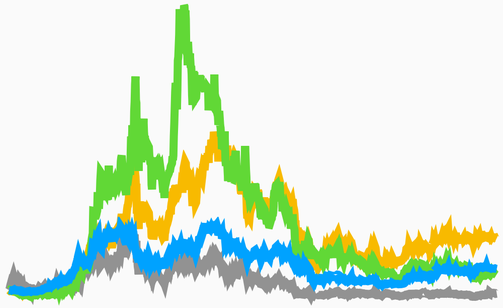
Parity Line



```
parity_processor = ParityBacktestingProcessor(
    perfs_alpha,
    perfs_beta,
    perfs_gamme,
)
parity_perf = parity_processor.backtest(initial_cash=1000)
```

Big improvements from v1

Historical Price & market cap data



User friendly APIs

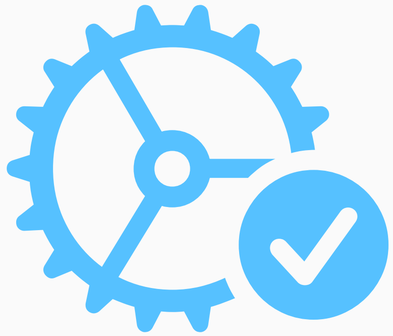
```
portfolio_hrp = Portfolio(
    base_value=initial_bid,
    initial_prices=df.loc[:start_date_portfolio],
    optimiser=HRPOptimization,
    max_weight=max_weight,
    min_weight=min_weight,
    weight_threshold=weight_threshold,
)
```

Fast

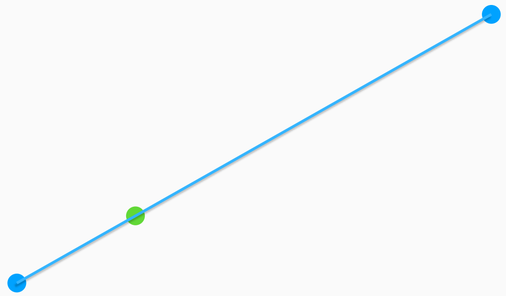
(Multi-threading & C++ optimisers)

Customizable

(parameters & delegates)



Parity Line Rebalancing



Excel Reports