Bachelor in Data Science and Engineering - 2021/2022

University Carlos III Madrid

Data Science Project - Group 96

Robo Advisor algorithm design for drawdown-based optimization of investment portfolios.

Collaborating with **IronIA Fintech**

Bernardo Bouzas García
NIA: **100406634**
David Méndez Encinas
NIA: **100406667**
Claudio Sotillos Peceroso
NIA: **10040940**
Laura María Torregrosa Gómez-Meana
NIA: **100406691**

Index

Introduction

Introduction to the partner

IronIA Fintech has collaborated in the realization of this project: IronIA is a financial asset management platform founded and based in Spain. Most traditional banking networks do not work with open architecture, they only sell their own products, but IronIA is committed to making available to customers more than **18,000 investment funds** in which they can invest their savings.

IronIA stands out with a differential value proposition in a financial sector where restrictions and obstacles make the novice investor's experience a complicated process that sometimes leads to poor investment decisions and loss of capital. By drastically reducing commissions and offering unlimited changes at no cost in its portfolio allocations, IronIA aims to reinvent the way to **invest with freedom**, at a very low cost in the form of a monthly subscription fee.

As it is a large number of funds, they have designed their own ranking to help the client find the best ones. They rate their funds with IronIA points -from 1 to 5, the same as the stars used to evaluate an application-. They have designed a model that makes it possible to discriminate between what is important and what is not in each fund in order to compare them. In this way, the client can see how a certain fund stands out.

IronIA uses **AI models** to rate their funds and then rank them based on their ironIA points. In addition, they use a kind of recommendation system to suggest funds that may be of interest to the clients. In this way, the clients can compare and choose which fund suits them best. This raises the level of the firm's activity from simply offering funds to being able to make **data-driven investment recommendations**, which is a considerable advantage for the small investor who needs additional tools to evaluate the market.

Introduction to the project

To introduce the approach to our work, we must make clear the central concept on which we build our model: funds, a type of financial product. **Funds** are **collective investment vehicles** managed by a professional fund manager. The fund can invest in a wide range of assets: bonds, equities, derivatives, currencies... They can also invest in any geographical area, as long as they adhere to the estimated risk profile determined for each fund.

The profitability of the investment is given by the performance of the fund. Each investment fund has its own risk exposure, and the investment can be adapted based on the risk that each participant is willing to assume, as well as their interests.

Just as there are shareholders of a company, there are unit holders of an investment fund, who are those investors who become unit holders of the fund in the proportion of the contributions they have made. Each of the unit-holders of a fund may leave the fund, obtaining the reimbursement of the investment, at any time. The units are negotiable securities, not normally traded on any stock market, but sold and repurchased by the management company. Note that IronIA is not the entity that manages the composition of the funds, they offer clients the possibility to invest in them.

Our task is to **explore different methods of portfolio optimization**, that is, to achieve an algorithm that indicates in which funds it is more convenient to invest, and the percentage of capital that should be allocated to that investment, based on the expected return and an exhaustive analysis that we will perform around different risk measures. In this way, we want to improve or complement the currently proposed model, relying on both **classical** portfolio optimization theory and more **modern** and disruptive approaches that we will study.

Initial thoughts

Historical background

To design a strategy consistent with the state-of-the-art in the industry, we must understand the historical basis of portfolio optimization, explore the techniques that have evolved since that point, and grasp the economic theory behind them.

Prior to 1952, portfolio construction was based solely on expected return, and did not take risk as a variable. Since one of the characteristics of the problem at hand involves the investor's risk profile, we must understand the solution to this inconsistency: the **Markowitz Model[1]**.

The theory of portfolio formation consists of three stages:

1. Determination of the set of efficient portfolios.
2. Determination of the investor's attitude towards risk.
3. Determining the optimal portfolio.

It also relies on some starting assumptions:

1. The return of a portfolio is given by its mathematical expectation or mean.
2. The risk of a portfolio is measured through volatility (variance or standard deviation).
3. The investor always prefers the portfolio with the lowest risk given an expected return.

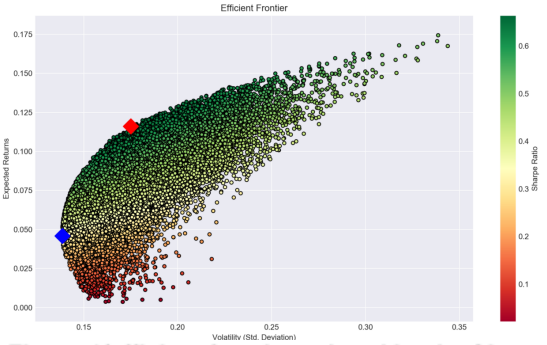


Figure 1(efficient frontier colored by the Sharpe Ratio of each portfolio's risks and returns)

An **efficient portfolio** is a portfolio that offers the minimum risk for an expected return value. On the **efficient frontier[2],** each portfolio minimizes risk for a given return. We should find it by tackling a mathematical optimization problem, which gives us a clue as to which approach we should follow. Ideally, an investor seeks to populate the investment portfolio with assets that offer exceptional returns, but whose combined standard deviation is smaller than the standard deviations of the individual securities. The less correlated the assets are (lower covariance), the lower the standard deviation. If this combination of optimizing the return versus risk calculation is successful, then that portfolio should align along the efficient frontier line. This is what we aim to achieve, building a model starting from this classical paradigm.

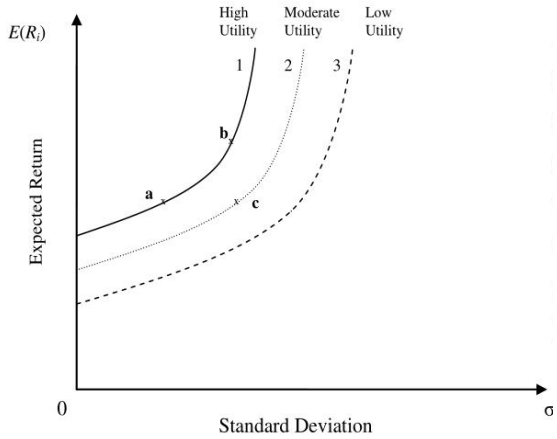


Figure 2(indifference curves)

The investor's attitude towards risk will depend on their map of **indifference curves[3]**.

Specifically speaking, a map of indifference curves is a set of contour lines for a utility function. If the utility function changes, then the map changes, unless it is an increasing monotonic transformation of the utility function.

Indifference curve maps must consist of curves whose utility level increases as they are located farther from the origin, strictly convex curves, to reflect the fact that scarce goods are valued more than abundant ones, continuous curves, to comply with the axiom of completeness, and curves that do not cut each other, to comply with the axiom of transitivity.

Therefore, each investor will have a different risk aversion and for each level of risk they are willing to assume, they will expect a certain return.

An investor's **optimal portfolio[4]** is determined by the tangent point between one of the investor's indifference curves and the efficient frontier. Curves below that point will give less satisfaction and those above that point are not feasible.

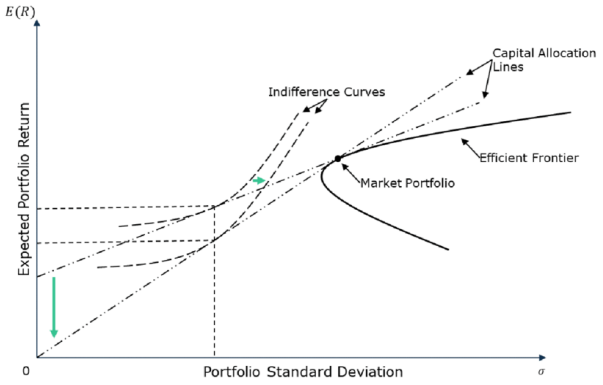


Figure 3(Impact of Changing Risk-Free Rate on Indifference Curve and Return)

This illustrates very well the concept we are trying to develop, and gives us a good understanding of this model, the fundamental basis of modern investment portfolio design theory, we move on to look for indices, or benchmark metrics, that can help us build our system and reliably optimize a set of solutions.

The first one we introduce is the **Sharpe Ratio[5]**, the relationship between the additional return of a mutual fund, measured as the difference between the fund's return and the return of a risk-free asset, and its volatility, measured as its standard deviation. We will take as the "risk-free asset" the return on short-term government bonds in the geographic area that most closely resembles the assets in which the fund invests.

The higher the Sharpe Ratio, the better the fund's performance relative to the amount of risk taken in the investment. If the Sharpe Ratio is negative, it indicates underperformance relative to the risk-free return. Any Sharpe Ratio less than 1 implies that the asset's return is less than the risk we are taking by investing in it. The Sharpe Ratio helps us to analyze whether the risk/return ratio of an investment is appropriate and to compare different funds within the same category, but it should always be considered in conjunction with other relevant metrics that are used as an industry standard.

However, knowing about the existence of metrics such as these has made us delve even deeper into the risk measurement we can perform, and this research has led us to determine certain key metrics that we will explain and formulate in detail in the following pages.

We came to understand that over more than seven decades, the methodology has evolved a lot, as well as the implementation strategies. Therefore, we decided to look for resources that, following the guidelines set by these critical initial models, would facilitate the development of a solution in a testing and benchmarking environment.

Available data overview

Working in a cloud environment

To design a suitable methodology, the partner firm for this project, IronIA Fintech, has provided seamless access to a collection of valuable resources. IronIA owns a database hosted on **Google Cloud** with very comprehensive data on tens of thousands of investment funds over the course of lustrums. The information available and its relevance will be detailed in the following paragraphs.

To access this data origin, we have been provided with a service account from which to initiate a connection to the Google endpoint. Therefore, instead of working locally and connecting the project through GitHub, the team has decided to start preliminary testing and pre-processing using **Google Collaboratory**. This allows us to leverage the in-cloud tools offered by Google, particularly the BigQuery API.

BigQuery supports a standard SQL dialect that complies with the ANSI:2011 standard, reducing the need to rewrite code. The tool also provides free ODBC and JDBC drivers to ensure that commonly used applications can interact with the platform's powerful engine.

Prices Data Frame

fund ID	fund name	currency	date	NAV

The **Net Asset Value[6]** of a fund is the unit price of each share in the fund at a given point in time. In our case, each date represents the daily **NAV** of each fund available in the dataset.

$$NAV = \frac{Value\ of\ Assets - Value\ of\ Liabilities}{Total\ Shares\ Outstanding}$$

Figure 4 (formula net asset value)

This value is the result of dividing the fund's net assets by the number of shares in circulation. The fund's NAV represents a "**per-share**" value of the fund, which makes it easier to be used for valuing and transacting in the fund shares. This is a dynamic concept, since it must be calculated on a daily basis according to the market prices of the assets comprising the fund's portfolio.

As in the case of IronIA, management fees charged to the clients are implicit for funds. In other words, the percentage corresponding to the fees is deducted from the result of the above formula. This is because these are charged directly and are already deducted from the NAV of the fund. Another nuance to highlight in this section is that the price at which a fund buy or sell transaction materializes may be different from the price on the day the order was actually given. We must bear in mind that the funds offered by IronIA are of a very diverse geographic nature, so it is essential that the market in which each fund is invested is open at the time the NAV is obtained. In this regard, days on which there is no market for assets representing more than 5% of the fund's assets are not considered business days for NAV purposes. In such a case, the NAV that would be available to us is, in effect, the NAV of the next business day.

This is the largest dataset (10.76 GB), as it contains the daily NAV record of each fund. However, it will be very useful for the calculations to be explained shortly.

Categories Data Frame

category	subcategory	benchmark	benchmark ID	morningstar ID

This dataset contains all the fund categories that IronIA considers. On the one hand, we find the Morningstar ID, in terms of the categories they use. Morningstar is a leading financial services provider that assigns ratings to funds based on an analytical estimate of a stock's target price, specifically:

1. Analysis of the company's competitive advantage.
2. Estimation of the target price of the stock.
3. Uncertainty about the estimated target price.
4. Current market price.

To evaluate the quality of the funds in a comparative way, these funds are grouped by Morningstar **categories**. These categories are referred to in the IDs presented in our database. We also find information about the **benchmark**. By definition, a benchmark is a comparative reference, a tool used to measure something against its comparables.

For **actively managed funds**, the benchmark is the reference index with which to compare their performance, but their portfolio does not have to be the same as the composition of their benchmark. In other words, a Spanish fund may have the IBEX 35 as its benchmark, but the fund's positions do not necessarily have to reflect exactly the same composition as the IBEX 35. The objective of the funds in this category is what is commonly known as "beating the benchmark", i.e. to outperform their reference index. In this way, we can test the consistency over time of an actively managed fund by seeing whether it consistently outperforms its benchmark over a moderate period of time or only on an ad hoc basis.

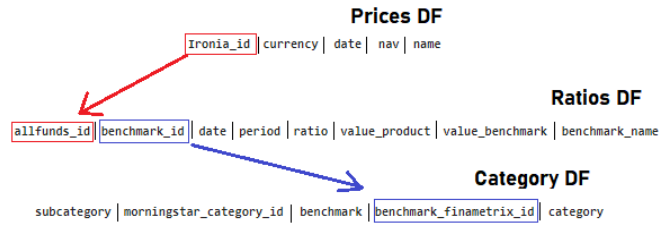
On the other hand, **index funds** and **ETF**s exactly replicate the index they track. This means that the composition of their portfolios does match that of their benchmarks, so that if the index goes up or down, the passively managed fund will register a very similar movement. Understanding this leads us to consider these types of products in our more conservative roboadvisor options, as we have historically found their growth to be more moderate but consistent over the long term.

Ratios Data Frame

id	benchmark id	date	period	value product	value benchmark	benchmark name

This dataset contains general information of the funds. It consists of 8 columns. The id, the benchmark ID, the date, the period (referring to the year), the . value product, the value benchmark and the benchmark name. This is the second biggest DF (7.05 GB) since it contains the yearly record of each fund for a given ratio. These three datasets are connected as the image shows.

Figure 5(structure of the dataset and connection between them)



Overcoming main dataset challenges

As we intend to test several approaches starting from the same initial conditions, it is of vital importance to have a complete and accurate dataset. To obtain it, we have had to face a very common problem in this type of project involving large amounts of data: the presence of **null** values.

Null values approach

We have null values when the value of a column is unknown or missing. A NULL value is neither an empty string (for character or date and time data types) nor a zero value (for numeric data types). The ANSI SQL-92 specification indicates that a NULL value must be the same for all data types, so that all of them are treated uniformly. It is by these guidelines that we have been guided in designing our null value handling strategy. Our first step is to get all the different ID's from the Ratios dataframe using a standard SQL query.

COVID19 impact

The second problem we face is the presence of irregularities in the behavior of financial markets in recent years. In this case, we are faced with a phenomenon that interferes abruptly and unpredictably in the global economy, a blackswan, the COVID-19 pandemic.

The impact of this pandemic has left a very particular situation in the markets. A priori, it seems that **fixed-rate products** have ignored the economic consequences of the virus. Last year, the COVID-19 crisis triggered one of the deepest recessions in history, with global growth declining by 3.6% year-on-year. After the initial widespread sell-off in equity markets, global fixed-rate equities (as calculated by the MSCI AC World index) went to offer a 15% return in 2020.

Secondly, investments in countries appear to have been highly influenced by the management of the sanitary situation and vaccine developments. A pattern can be identified in which countries where **economic growth expectations** have improved have shown the best performance in terms of developing a potential vaccine. However, this has not translated into higher **variable rate equity returns** in these markets. Surprisingly the other way around, the relationship between vaccination rates and growth expectations has developed most notably in **foreign exchange** (forex) markets.

In conclusion, the impact of the pandemic has been very diverse, slightly irrational and unpredictable. Therefore, we have decided to design our model with the extra assumption that no black swan like the one we have just witnessed will interfere with its performance. This allows us to perform our analysis with more consistent and robust data. Accordingly, we have looked for all the days that the stock market has been open **from 2016 to 2019** and we have generated a file containing those dates, thus avoiding the influence of 2020 and 2021. In addition, we consider that it can be highly representative, since markets have performed a "V-shaped recovery", returning to a state very similar to what would have been expected if growth had not been affected by the pandemic. To illustrate this, we show the growth of the S&P 500 (SPX), one of the most important stock market indexes in the United States; it is considered one of the most representative indexes of the real market situation. Source: Google Data.

Figure 6(Growth of S&P 500(SPX))

Cleanness vs representativeness tradeoff

Now that we have the right data collection approach, we begin the extraction process.

We iterate through each ID and we check that it has at least the length of the dates minus a delta parameter (for the tests is set to 30), otherwise, the ID is discarded. By doing this, we are ensuring that all the IDs have a small value of null values, getting just the funds that have **almost the complete time series**.

By having a substantial number of funds, we have considered the tradeoff between data cleanness and representativeness, since we are removing some options that might be optimal for the model. We have decided that this is a compromise we are willing to make, since there are so many funds, many of them share similar characteristics and we think they could be substituted in many portfolios with little or no change in performance.

Final linear interpolation

Nevertheless, there are still missing values, so we have to apply a new null processing technique. In this case, we could take the value closest to the NULL under consideration. However, we believe it is more appropriate to **approximate more closely by interpolation**. Interpolation, we expect, will give us a small error with respect to the true function value, but it will always be smaller than taking the nearest value

There is linear interpolation[7], quadratic interpolation[8], cubic and nth power interpolation (for data that have the characteristic of being raised to powers of order 2, 3, 4, ..., ..., and up to n), exponential interpolation, etc. In our case, observing linearity in the values we are studying, we choose to apply linear interpolation, which is a particular case of Newton's general interpolation. Moreover, the smaller the interval between the data, the better the approximation. This is due to the fact that, as the interval decreases, a continuous function will be better approximated by a straight line. Our intervals are very short, and the daily data provide a good basis for this interpolation option.

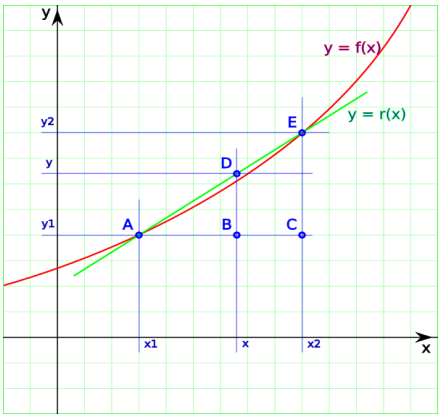


Figure 7 (linear interpolation)

In simplified terms, just for illustration purposes, linear interpolation consists of drawing a line through (x1,y1) and (x2,y2), y = r(x) and calculating the intermediate values along this line instead of the function y = f(x). For this purpose we rely on the similarity of triangles $\widehat{BAD}$ and $\widehat{CAE}$.

$\frac{\overline{AC}}{\overline{AB}} = \frac{\overline{CE}}{\overline{BD}}$ then, $\overline{BD} = \frac{\overline{AB}}{\overline{AC}}\,\overline{CE}$, or what is the same, $(y - y_1) = \frac{(x - x_1)}{(x_2 - x_1)}(y_2 - y_1)$, so $y = \frac{(x - x_1)}{(x_2 - x_1)}(y_2 - y_1) + y_1$

Once all the data is properly filled and cleaned, we split it into **two different data frames**, one with the train set (2016-2018) and the other with the test set (2019). The final result is a train and test set with a total of **12367 different funds**. The time needed to generate the dataset was around **10 hours**, as each ID query takes **3.5 seconds** to complete.

Initial Portfolio Allocation

Determining the initial structure of the portfolios is one of the instrumental parts of this project. We take our first steps in this direction, because it allows us to familiarize ourselves with basic and complex asset allocation techniques, and it will also produce a framework on which our robo-advisor can iterate and improve as time progresses.

As a general rule, **proper diversification** will provide greater long-term appreciation potential while keeping risk contained. Ideally, the robo-advisor should be based on an optimized pool of funds, a mix of publicly traded assets and debt instruments consistent with the parameterized financial objectives and needs.

The scale and complexity of portfolio optimization over many holdings means that the work generally requires a high computational cost, which must be mitigated by seeking efficiency. Fundamental to this optimization is the construction of the covariance matrix for the rates of return of the assets in the portfolio, and we highlight the two main methods: linear and nonlinear programming.

Linear vs nonlinear programming algorithms

Linear programming encompasses methods to achieve the best outcome in a mathematical model whose requirements are represented by **linear relationships**. On the other hand, nonlinear programming solves optimization problems where the constraints or the objective functions are nonlinear.

When it comes to finance applications, and more specifically the optimization of portfolios of traded assets, there are several situations that can be approached from both perspectives. However, the state-of-the-art of the industry suggests a linear approach, since it has demonstrated exceptional effectiveness and robustness. It has been shown that it can, and does, successfully handle portfolio allocation problems effectively with thousands of funds and scenarios, regardless of the risk measure employed.

In addition, there are two main reasons why we prefer to focus on this approach: the computational cost, and therefore the time spent on testing, is much higher in non-linear programming, especially if we are not able to optimize our model to be ultra-efficient. This, moreover, would incur a difficulty that we prefer to avoid in order to dedicate more time to other areas of exploration within the framework of this project. Furthermore, this is a type of optimization problem that we have previously studied in our bachelor's degree in Optimization and Analytics subject, thus it is a familiar realm that we are comfortable working on.

Risk management optimization

We will be working with a well-known one-parameter family of risk functions defined on portfolio return sample-paths, which is called conditional drawdown-at-risk (CDaR). These risk functions depend on the portfolio drawdown (underwater) curve considered in active portfolio management, and were originally proposed in *PORTFOLIO OPTIMIZATION WITH DRAWDOWN CONSTRAINTS*[9], a paper researched by Alexei Chekhlov, Stanislav Uryasev and Michael Zabarankin.

The CDaR[10] family of risk functions originates from the conditional value-at-risk (CVaR)[11] measure, as we will comment on later. The Mean-Absolute Deviation and Standard Deviation risk measures are very similar by construction – they both measure average deviation, so their efficient frontiers and transition maps will probably be very close. On the other hand, the Maximum Loss measures the extreme deviation. Thus, and as proposed in the aforementioned paper, we have two classical approaches with which to compare the performance of our specific model.

Conditional Value-at-Risk (CVaR)

CVaR is a statistical technique used to measure the level of financial risk within an investment portfolio over a specific time frame. It derives from the value-at-risk for a portfolio. VaR allows quantifying the exposure to market risk, i.e. to estimate the loss that could be suffered under normal market conditions within a time horizon, given a confidence level (1- α) -- usually 95% or 99%.

While VaR represents a worst-case loss associated with a probability and a time horizon, CVaR is the expected loss if that worst-case threshold is crossed. CVaR, in other words, quantifies the expected losses that occur beyond the VaR breakpoint.

VaR calculation (for just 1 fund)

Our VaR is computed using the NAV of the Prices dataframe. First, we compute the Rate of Return by subtracting each daily NAV, the directly previous NAV recorded, and divide by this same value. This generates a daily **Rate of Return vector**.

VaR depends on a risk level. Let's suppose that we want to know the VaR at a risk of 5%. Then, we need to find out the 5% percentile of the Rate of Return vector.

$$Rate\ of\ Return = \frac{NAV_i - NAV_{i-1}}{NAV_{i-1}}; being\ the\ i^{th} day$$

CVaR calculation (for just 1 fund)

Once we have the VaR, computing the CVaR is simple, as the CVaR is obtained by taking a weighted average of the "extreme" losses in the tail of the distribution of possible returns, beyond the value-at-risk (VaR) breakpoint. In order to do so, we obtain the mean of those Rate of Return values which are smaller or equal than the VaR.

The fundamental difference between VaR and CVaR as risk measures is that VaR is the "optimistic" low bound of the losses in the tail, while CVaR gives the value of the expected losses in the tail. In risk management, we prefer to be conservative rather than optimistic.

Conditional Drawdown-at-Risk (CDaR)

CDaR is relatively similar to CVaR, since both metrics measure the average value beyond a certain level in the distribution. Thus, similarly, CDaR is the average of all drawdowns, or cumulative losses, in excess of a certain threshold. That threshold is referred to as **drawdown-at-risk**.

Drawdown is a risk measure used to evaluate how long it typically takes an investment to stabilize its net asset value (NAV) from a temporary decline. In other words, measures the **current backward movement in the yield curve** with respect to the previous peak in the curve.

Drawdowns are inevitable in any portfolio: any liquid investment in open markets will, by the simple variation over time of its price, experience drawdowns. Drawdown-based indicators are difficult to implement in discretionary methodologies, as it is often complicated to have objective historical data to analyze drawdown, but in automated computing situations , such as ours, they can be very useful if the market tends to be more volatile.

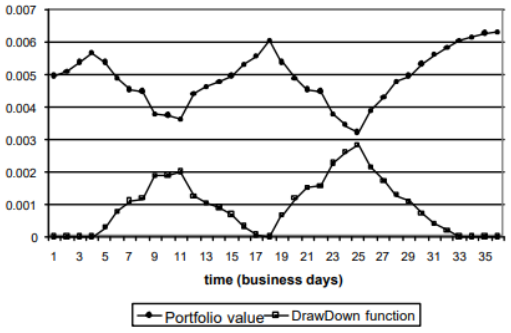


Figure 9

DaR calculation (for just 1 fund)

First, we compute the daily maximum cumulative NAV. This is a vector, in which each date it contains, has associated the maximum NAV until that day. Then, we subtract to each daily NAV its respective daily max cumulative NAV, and divide the result by this same value. Finally, we just have to find the Risk% percentile and that will be the fund DaR.

This approach reflects quite well the preferences of investors. For instance, an investor may consider it unacceptable to lose more than 10% of his investment. Another investor may excuse short-term DrawDowns in his account, but he will definitely worry in case his capital suffers a long-lasting DrawDown.

$$Drawdown = \frac{|NAV - Max\ Cumulative\ NAV|}{Max\ Cumulative\ NAV}$$

Figure 10 (formula of drawdown)

CDaR calculation (for just 1 fund)

Once we have the DaR, computing the CDaR is straightforward. We obtain the mean of those DrawDown values which are smaller or equal than the DaR for a specific risk level.

For instance, 0.95-CDaR can be thought of as an average of 5% of the highest drawdowns.

Mean-Absolute Deviation (MAD)

Generally speaking, the Mean-Absolute Deviation[12] is the average distance between each data point and the mean. That is, it allows us to calculate how much the values of a set of data vary from their mean. A low value for MAD is an indicator that the data values are concentrated close together, while a high value reflects that the values are more widely scattered. In finance it is used to measure the portfolio's volatility, and it is computed with the portfolio´s rate of return. In our case we will use the Rate of Return to compute it.

Figure 11 (formula mean-absolute deviation)

$$MAD = E[|\, r_p(\mathbf{x}) - E[r_p(\mathbf{x})]|]$$

It has been proven that portfolios on the MAD efficient frontier correspond to efficient portfolios in terms of the *second-order stochastic dominance*; i.e, for two bets A and B, bet A has *second-order stochastic dominance* over gamble B if the former is more predictable. In relation to portfolio optimization, investing in a portfolio which has second-order stochastic dominance implies making a “safer” investment.

Maximum Loss (ML)

Maximum Loss[13] is a method introduced for identifying the worst case in a given scenario space, called **Trust Region**. It is one of the simplest and most classic measures of risk, but at the same time it is also intuitive. However, we include it for comparative purposes, since the aforementioned Value-At-Risk indicator arose as a result of the need to improve the risk management of financial institutions on the part of regulatory bodies.

Market-neutrality (MN):

It is known that the market itself is a risk factor. If the instruments of the portfolio are positively correlated with the market, the portfolio will follow closely the market movements.

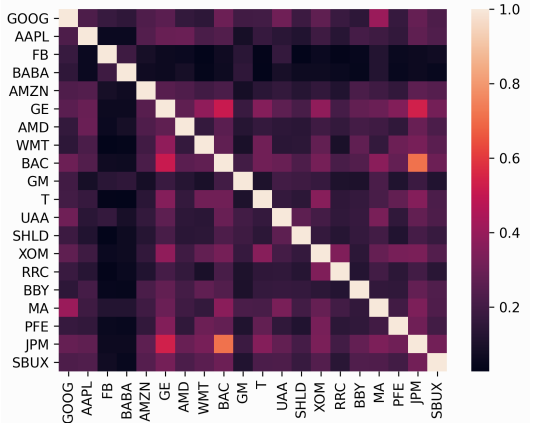
In order to avoid a scenario in which a portfolio's assets suffer large declines in the event of a market reversal in trend, portfolios can be designed in a *market-neutral[14] way.* This means that being moderately uncorrelated to the general market behaviour is a factor to be considered when selecting assets.

A fund's **beta parameter** measures the variability of the fund's performance compared to the variability of the benchmark's performance. In other words, it measures whether the fund is more or less volatile than its reference index. In this way, through beta, we can get an idea of the market exposure that each asset is assuming. We have already discussed this in detail previously, when we studied the composition of the dataset provided by IronIA. In order to be market-neutral, ideally,a portfolio should have a zero beta, or at least, close to zero.

Market-neutrality will be used as a constraint in the portfolio optimization problem. However, we will later raise a particular difficulty that we have encountered due to a discrepancy in the way we define the concept of "market neutrality" and how some Python libraries approach it.

The idea is to obtain a result with uncorrelated **funds** that should show a similar covariance matrix as the one shown here as example. Note that this covariance plot is made using individual stocks. Instead, we must study the covariance matrix of funds, which may be different, as they may contain similar stocks that perform correlatedly.

Figure 12 (correlation matrix of some funds)



Inherited assumptions

To design our model following a rational and well-founded process, we have considered all the assumptions derived from **classical portfolio theory[15],** including:

- We expect all investors to be rational and all have access to the same information.
- All are risk-averse and share the goal of maximizing returns.
- We expect no investor to be able to influence the market.
- All of them base all their decisions in the market on expected returns according to some measures of risk.

In addition, we consider **modern nuances** such as market players having access to unlimited funds at a risk-free rate, as well as assumptions inherited from the **aforementioned paper**, which we detail in depth as being the most relevant:

Liquidity

Liquidity considerations are not taken into account. We define an asset in terms of its expected return and risk, but liquidity should be taken for granted, as we are implicitly assuming assets are listed on a global-scale, liquid market. Indeed, this is the case for the funds that IronIA offers.

Transaction costs

Transaction costs are the trading costs of changing the weights of portfolio elements. Since the optimal portfolio changes over time, there is an incentive to re-optimize frequently. This variable is doubly neglected: because of the assumptions inherited from the model we are trying to replicate and because our partner IronIA Fintech does not charge transaction fees to its clients.

Investing restrictions

A country's law may prohibit certain investors from owning some assets. Sometimes, it is not practical to hold an asset because the associated tax cost is too high. However, as Spain is not a country with restrictive legislation in terms of the profile of investors allowed access to financial products, and the tax burden of the State is not a point we particularly want to delve into in this project, we will assume that there are no investing restrictions.

Credit and non-reflected risks

Credit and other risks which directly are not reflected in the historical return data are not taken into account. **Credit Risk[16]** is the possibility of a loss resulting from a borrower's failure to repay a loan or meet contractual obligations.

Survivorship bias

In finance, survivorship bias[17] is the tendency for failed companies to be excluded from performance studies because they no longer exist. In our model, it is not considered.

Core approach

It is assumed that we have '***N'*** funds available in order to generate our optimal portfolio. The general idea is to maximize the expected return of the portfolio subject to different operating, trading and risk constraints. Let's go one by one.

First, the **Objective function** which represents the expected return of the portfolio. x_i is the position of asset (fund) i in the portfolio. r_i is the rate of return of asset i.

$$\max_x \; E\left[\sum_{i=1}^{n} r_i x_i\right]$$

This objective function is subject to four different constraints. These are:

Fund limitation

We limit the number of funds which can be used ($i = 1,..., n$)and also, the weight of the fund must be a value in the interval [0,1] ($0 \leq x_i \leq 1$).

Budget constraint

It ensures that the sum of the fund weights does not surpass one.

$$\sum_{i=1}^{n} x_i \leq 1,$$

Risk of financial loss

In this constraint we can use one of the four previously mentioned methodologies (CVaR,CDaR,MAD or MaxLoss). For instance, let´s work with the CVaR methodology. In this case, we will compute the CVaR function over the selected fund´s weights and upbound this resulting value with a risk tolerance level (this is the fraction of the portfolio value that is allowed for risk exposure). $\Phi_{RISK}(x_1,..., x_n) \leq \omega$

Market neutrality

This constraint also controls the risk of financial losses. In this constraint we force the portfolio to be market-neutral. We bound the portfolio´s correlation with the market, and as a result the portfolio won't follow significant market drops. β_i represents market´s beta for the fund i . k is a small number to ensure that the beta sum remains close to zero.

In the paper, they follow two approaches. Using this last constraint or not using it. They anticipate that using this constraint significantly improves the out-of-sample performance of the algorithm.

Linearization

The problem with these four risk constraints is that they aren't linear (either due to maximum functions or absolute values). Thus we have to linearize them in order to be able to program them in our optimization modeling language (we will use Pyomo) . In the below images you will see their linear to non-linear transformations. We have obtained the linearizations from the paper, except from the **CDaR linearization**, which we have obtained through our own calculations.

CVaR calculation (for the Portfolio)

$$\zeta + \frac{1}{(1-\alpha)J}\sum_{j=1}^{J}\max\left\{0, -\sum_{i=1}^{n} r_{ij}x_i - \zeta\right\} \le \omega \longrightarrow \begin{cases} \zeta + \dfrac{1}{1-\alpha}\dfrac{1}{J}\displaystyle\sum_{j=1}^{J} w_j \le \omega, \\ -\displaystyle\sum_{i=1}^{n} r_{ij}x_i - \zeta \le w_j, \quad j = 1,\ldots,J, \\ \zeta \in \mathbb{R}, \quad w_j \ge 0, \quad j = 1,\ldots,J. \end{cases}$$

CDaR calculation (for the Portfolio)

$$\eta + \frac{1}{1-\alpha}\frac{1}{J}\sum_{j=1}^{J}\max\left[0, \max_{1\le k\le j}\left\{\sum_{i=1}^{n}\left(\sum_{s=1}^{k} r_{is}\right)x_i\right\} - \sum_{i=1}^{n}\left(\sum_{s=1}^{j} r_{is}\right)x_i - \eta\right] \le \omega$$

$$\begin{cases} \eta + \dfrac{1}{1-\alpha}\dfrac{1}{J}\displaystyle\sum_{j=1}^{J} w_j \le \omega\ ; \\ Z - \displaystyle\sum_{i=1}^{n}\left(\sum_{s=1}^{j} r_{is}\right)x_i - \eta \le w_j \quad j = 1,\ldots,J\ ; \\ \displaystyle\sum_{i=1}^{n}\left(\sum_{s=1}^{K} r_{is}\right)x_i \le Z \quad K = 1,\ldots,j\ ; \\ \eta \in \mathbb{R},\ w_j \ge 0 \quad j = 1,\ldots,J. \end{cases}$$

MAD calculation (for the Portfolio)

$$\frac{1}{J}\sum_{j=1}^{J}\left|\sum_{i=1}^{n} r_{ij}x_i - \frac{1}{J}\sum_{k=1}^{J}\sum_{i=1}^{n} r_{ik}x_i\right| \le \omega \longrightarrow \begin{cases} \dfrac{1}{J}\displaystyle\sum_{j=1}^{J}(u_j^+ + u_j^-) \le \omega, \\ \displaystyle\sum_{i=1}^{n} r_{ij}x_i - \frac{1}{J}\sum_{j=1}^{J}\sum_{i=1}^{n} r_{ij}x_i = u_j^+ - u_j^-, \quad j = 1,\ldots,J, \\ u_j^{\pm} \ge 0, \quad j = 1,\ldots,J. \end{cases}$$

MaxLoss calculation (for the Portfolio)

$$\max_{1\le j\le J}\left\{-\sum_{i=1}^{n} r_{ij}x_i\right\} \le \omega \longrightarrow \begin{cases} w \le \omega, \\ -\displaystyle\sum_{i=1}^{n} r_{ij}x_i \le w, \quad j = 1,\ldots,J. \end{cases}$$

Technical methodology

PyPortfolioOpt library

This is a python library that implements portfolio optimization methods. With it we have been able to develop the approaches of CVaR and CDaR. It relies on two main design principles: it should be easy to swap out individual components of the optimization process with the user's proprietary improvements and that it is better to be self-explanatory than consistent. They present some advantages over existing implementations:

- Easy to combine with our strategies and models.
- Includes both classical methods (Markowitz 1952 and Black-Litterman[18]), suggested best practices (e.g covariance shrinkage), along with many recent developments and novel features, like L2 regularisation, exponential covariance, hierarchical risk parity.
- Provides native support for pandas dataframes.
- It is quite robust to missing data, and price-series of different length.

Usability

In order to solve this specific problem is a quite good library in order to start playing with the data we have.

It uses as input data the mean historical return (mean of the returns of each fund along the days), the returns covariance matrix, the risk level that the client is willing to take and the amount of capital the client is willing to invest.

Also it includes the option of imposing an L2 regularization term. By doing this, the optimization algorithm spreads more the weight values, creating a portfolio with more funds. The amount of regularization can be modified using the gamma parameter.

The method will give us as output the weights given to each of the funds, the amount of capital inverted on each of the funds, the expected annual return (in percentage) and the remaining amount of capital that the client will keep after investing (since the algorithm may determine that investing all the budget isn't necessary) .

Limitations

In addition to that it does not allow us to implement the optimization algorithm with the MAD and MaxLoss constraints, it also limitates us in the Market Neutral constraint. This library enforces the portfolio to be market neutral by making the sum of its fund´s weights equal to zero. In order to achieve this, they modify the weight range of values to [-1,1]. Thus now weights can be negative (a negative weight would mean to perform short selling for that specific fund). Performing short selling can be a riskier activity, and since we want to have control over the risk of our portfolio it isn't a good approach to use the market neutrality provided by *PyPortfolioOpt*.

Pyomo library

Pyomo is a Python-based software commonly used for formulating, solving, and analyzing optimization models. We have decided to use this software for the formulation and solving of our Linear Problem due to the versatility and easy use it offers. Pyomo has a notation similar to what we would use in the mathematical definition of these problems.
With it, we have been able to program the linear problem from scratch. The most challenging part was to program the 4 risk constraints in an efficient way, since some of the constraints took very long time to be executed because of their linear formulas (i.e. MAD, CDaR).

Experiment

In this subsection we will compare the results obtained with the two approaches. Also, we want to make clear which is the best risk measure (CVaR, CDaR, MAD or MaxLoss) and check to what extent adding the market neutral constraint improves the portfolios. To carry it out, tests have been performed in different environments: using the **CPLEX** solver, locally, and **ECOS-BB** solver in the preliminary tests carried out with Google Collaboratory.

It is important to note that we have noticed that on many occasions, the result of the optimization yields a very low number of funds, i.e. the vast majority of weights allocation is 0. This is a problem, mainly due to lack of diversification. We decided to choose a diversification strategy for several financial reasons:

1. **Mitigates investment risk**. Spreading capital over several instruments allows offsetting losses on some assets with gains on others.
2. **Allows to set several investment objectives**. In case of further customization of these portfolios, in the future, investing in instruments of various maturities facilitates the achievement of profitability goals.
3. **Allows for currency hedging**. This will potentially improve returns and allow hedging against exchange rate fluctuations.

We can illustrate this situation with the following plot, where we show the variance using only 1 fund, compared to a portfolio of 30 optimized funds. Although it is true that we could obtain more gains at some moments, we prefer the stability of a portfolio to the risk of putting all eggs in the same basket.

Figure 13 (test returns with random weigths)

To solve this problem, we use a built-in objective function which borrows the idea of **regularization** from Machine Learning. The loss of precision needs to be accounted for by something else to maintain accuracy levels. This trade-off will be borne by the bias portion of the model equation, being the bias the part of the model equation that does not depend on the feature data.

In order to coerce the optimizer to produce more non-negligible weights, *pyportfolioopt* library adds what can be thought of as a "small weights penalty" to all of the objective functions, parameterised by γ **(gamma)**. This is the parameter we fine-tuned.

Considering, for example the minimum variance objective for instance, the library performs as follows:

$$\text{minimize } w\ \{wT\Sigma w\} \ \longrightarrow\ \text{minimize } w\ \{wT\Sigma w+\gamma wTw\}$$

After applying regularization, we set our risk measure and our risk value. With those values, we pass our train set but keeping just 500 funds.

The reason why we need to limit the selection of funds to only 500 funds is because, when the number of funds is increased, due to the L2 regularization, **the problem becomes non-linear**, therefore, not solvable.

In further expansions of this approach we will be trying different techniques involving hierarchical computation methods and sharding to solve this problem, thus making our model more accurate and scalable.

Results

With this initial procedure, we are getting a set of optimal efficient portfolios using the weights obtained in the train (years 2016 to 2018) and evaluating them on the test (year 2019). In the following figure, we compare the different approaches proposed. For a correct analysis, it is necessary to take certain considerations into account:

- **Test Return** is the return that the optimal portfolio, for each risk measure to be optimized, would have offered to an investor in 2019.
- **Expected Return** is the return that the *pyportfolioopt* library model predicts that an optimal portfolio could achieve, for each risk measure to be optimized, also in the year 2019, without knowing the ground-truth.
- The models called **Random** are a comparative benchmark that simply provides an approximation to the average return of random funds selected with random weights. Basically, it measures the expected return for a certain random sample. It serves for comparison purposes.
- Some risk methods do not show any expected return. On the one hand, random models are not provided with one, due to their relative triviality. On the other hand, MAD and ML approaches were solved using pyomo library, which in our current implementation isn't able to provide this comparative additional information.
- The notation '**_r*X***' refers to the *X*% risk tolerance that is associated with that optimal portfolio. For random portfolios, '**_di1**' means that only 1 fund is selected, while '**_di30**' is the random allocation for a 30-fund portfolio.

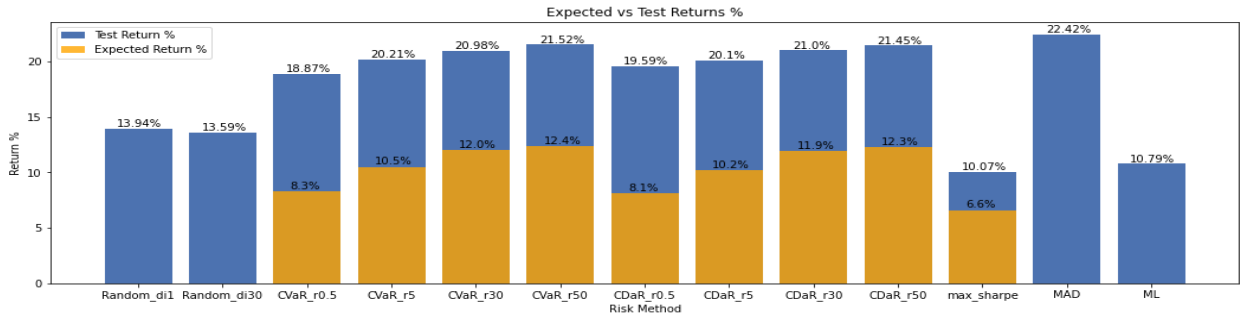
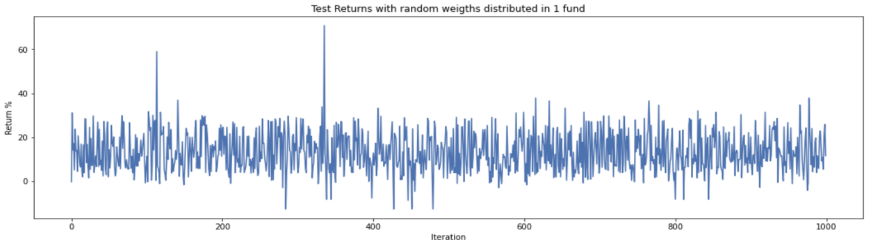


Figure 14 (expected vs test returns)

The first noteworthy finding is that the optimal efficient portfolios obtained through our comparative models **outperform the returns that could be expected from them**. (Test Returns, in blue, over Expected Returns, in yellow). At least, this would have been the case during 2019. This, we believe, is due to the particular choice of risk measures chosen in the scope of this project; we have fortuitously targeted a segment of financial market activity where value and drawdown at-risk have turned out to be a more representative metric than perhaps might be expected.

Within this framework, the family of special interest was the drawdown-related. The CDaR-optimized portfolios outperformed the CVaRs in the **0.5%** and **30%** risk categories, while the opposite happens in the **5%** and **50%** risk categories, leaving us with almost even results. Compared to the rest, our "conditional-at-risk" family portfolios **outperform** those optimized for **2 of the 3 classical comparison metrics** (Sharpe Ratio and ML) and **all randomly selected ones**. They are only bested by those where MAD is used, which is satisfactory enough for us.




```
Mean: 13.375350000000001
Variance: 86.3397482775
Min value: -12.65
Max value: 70.86
95% Quantile: 28.4
75% Quantile: 20.58
25% Quantile: 6.3125
5% Quantile: 0.5835000000000004
```

Figure 15 (test returns with random weights in 1 fund)


```
Mean: 13.6053
Variance: 3.88936191
Min value: 7.48
Max value: 19.66
95% Quantile: 16.83
75% Quantile: 14.9925
25% Quantile: 12.21
5% Quantile: 10.4795
```

Figure 16 (test returns with random weights distributed in 30 funds)

To conclude this analysis, we delve a little deeper into the experiment performed with the random selection of portfolios with 1 and 30 funds. As the number of iterations is large, we observe similar means, but we know that this is not a real market situation. We can observe especially the volatility derived from a 1-asset portfolio, while the range in which potential gains and losses oscillate is much narrower in a diversified portfolio (**[7.48% - 19.66%]** vs **[(-12.65) - 70.86]**) . This confirms our previous assumption that it is safer to have a diversified portfolio, even if we sacrifice a potential triple return.

Phase 2: Model refinement

At this point we had a well-founded algorithm that provided an optimal investment portfolio with the possibility of defining risk with different metrics.

However, there were some key points we needed to work on to improve the details and solve some conceptual and performance problems.

Fund classes adjustment

The classes of a mutual fund are different types of fund units, each of which carries a different management fee for the unitholders. The different classes of a fund share the same name, to which is added a letter identifying the class (e.g. A., B, C...). There may be differences between classes as a consequence of:

- Minimum investment amount
- Investment fund fees
- Minimum duration of investment in the fund
- Existence of classes with hedged currency

Conceptual adversity

Fund managers normally use the letters to differentiate between investment funds aimed at people who can afford a high minimum investment (institutional class) or people who cannot (or retail class).

The ideal solution would be to filter those funds according to the budget that the client is willing to invest. For instance, if a client inserts an initial budget of 2000$ he can't invest in certain funds which have a minimum investment limit. Since we don't have this information in our DBs we would need to check which is the meaning of each of the letters and establish certain conditionals depending on the budget.

What we have decided is to eliminate those funds whose name contains the letters “A”, “I”, or “P” (Remark that there were other names which also contained these letters but weren’t between comas. Since we didn´t know if these were in the same situation as the ones between commas, we decided to keep these). We decided to eliminate some of the funds that we knew corresponded to asset classes that were not accessible to our target clientele, although we are aware that we may have left out quite a few, as each fund manager assigns the letter that it considers appropriate to mark this differentiation between classes.

- “A” : for >30,000€ of capital only
- “I”: for institutional investors only
- “P”: for private bank only

After deleting these funds we passed from **12367** funds to **8285.** However, this wasn’t the unique problem we had with respect to the fund´s names.

Optimization adversity

In our initial approach, the optimizer created Portfolios with very similar funds. As can be appreciated in the figure, there are funds that belong to the same group (i.e. *THREADNEEDLE LATIN AMERICA "Z"* and *THREADNEEDLE LATIN AMERICA "2" INA (GBP)*).

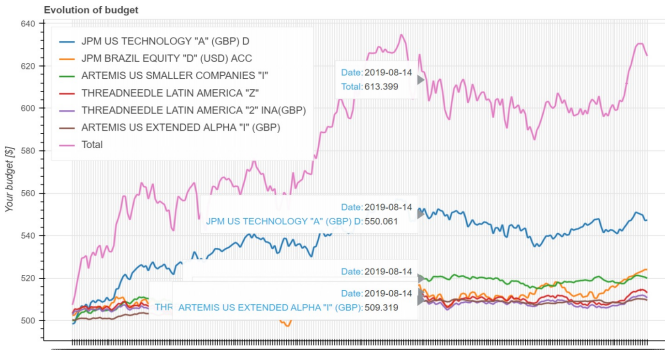


Figure 17 (Evolution of the budget distributed in different funds)

This issue appeared in every optimization approach performed and impacted the results :

Why was it decided to avoid this from happening?

We noticed that those funds which belong to the same group behaved in a similar way (both increased and decreased at the same time and in the same proportion). Thus, in order to have more diversity in our Portfolio, we decided to eliminate those funds which were very similar to others.

What criterion is used to sort out these specific funds?

To decide which funds' names were similar to others, we have decided to use the **Levenshtein distance** [19]. In information theory and computer science, the Levenshtein distance is a string metric for measuring the difference between two sequences. It has multiple applications, especially in recent years, due to the increasing interest in natural language processing problems.

The core concept is pretty straightforward; in simple words, the distance between two words is determined by the number of operations (these are letter removal, addition, or replacement) that are needed for transforming one word to the other (i.e. the distance between *"risk"* and *"think"* would be 3). The **complexity of the algorithm is O(m*n)**, where n and m are the length of the strings to be compared. This is a considerably satisfactory efficiency, given the moderate accuracy it presents in this situation, compared to more efficient but costly methods.

Again, it is important to mention that this solution is optimal only for Minimum Viable Product deployment, and may not be as efficient on a large scale. To this end, we have found that there are ways to use the **cosine similarity[20]** to approximate these distances for a hypothetical scaled tool.

Having this clear, the following strategy was followed: fixing a distance threshold (7, 8, and 10 are the ones we have used) iterate along all our fund´s names, and for each fund, we searched those funds which were similar to the given fund (that is which had a distance w.r.t the fund smaller than the threshold).

Having found the more similar funds for each individual fund name, we just have to delete similar funds from the list of names, and once we delete a certain fund, we no longer have to delete the funds that are similar to the deleted fund.

Finally, the filtered list of names was stored in a pickle so it is not necessary to repeat this process always. The list is used to filter the columns of our Pandas where we have all the series of returns. As we mentioned, we used three thresholds. In the below table, you can check the remaining funds after the filtering.

Threshold	Number of funds left	Filtering Quality
10	2632	High
8	3323	Medium
7	3705	Medium

After checking the quality of the filtering of each one of the thresholds (how similar were the chosen funds for the Portfolio) and taking into account the number of remaining funds, we decided to keep the filtering performed by threshold 7.

Hierarchical computation

As we have lots of funds, we have encountered some computational issues. The first one, was that managing a covariance matrix that big, causes in some occasions a complete crash of the solver. The second one, was the computational time required to compute those matrices.

In order to solve both problems, we have created a function that computes in a hierarchical way the optimal portfolios regrouping the selected funds in each iteration. This is achieved by cutting the funds in groups of x by x (x given by a parameter, usually we have used 500). Once we have a number of selected funds under 500 we proceed with the normal optimization and results gathering as always.

We know that this solution could lead us to a suboptimal solution but for having a minimum impact in this solution, we have selected a higher gamma than normal during the iterative process in order to take a higher amount of funds and don't lose one that could be of potential interest. In practise, we have tested this method vs the non hierarchical approach and the results when they were different, they were pretty close. This technique, thus, will give consistency to our project and help us to speed up obtaining more reliable results.

Betas challenges

With respect to the betas, we faced several challenges before being able to implement this approach correctly. First of all, let's remember that we want betas for performing the Market neutrality constraint. It's important to remark that what we needed to fulfill this constraint is just one beta per fund, and in the IronIA´s Ratios Database we had lots of beta ratios for each fund.

Remember that ratios are computed taking into consideration data between the day indicated going back the number of years it´s period (1y-5y) indicates This means that if for instance, the beta ratio has been computed on the day *"01-01-2021"* and its period is *"3y"*, the computed beta is computed with the data between the dates *"01-01-2018" and "01-01-2021"* . Initially we were unaware of this fact. We just saw the date column, and since it just contains dates of 2021, we thought that we didn't have betas of our period of time [2016-2019].

Having this clear, now we must decide which betas we want to keep for our approach, and how to get from all these betas just one per fund. Our first approach was, for each fund, take all it´s betas and compute the mode of these, not mattering the period it belonged to. Since this wasn't very precise (probably because we were taking betas of different periods), we decided to use just betas of the period *"5y"* which contains all the information between 2021 and 2016 (covers the temporal range of our data). Furthermore, we thought it was more convenient computing the Median instead of the Mode. This was because betas have 4 decimal values, thus it wasn´t very probable to have repeated ratios.

Finally, having one beta per fund, the only thing that remained was to use these in our algorithm. As expected, when adding this constraint to our optimizers, the funds obtained are more conservative and thus the expected annual returns are reduced (by around 7%). The truth is that the results aren't as good as we thought they were going to be after adding this constraint. We think this may be due to the fact that the betas we get from IronIA may not be the same betas to which the article refers. Because of this, we have decided not to include this constraint in our final Demo APP.

Risk-balanced portfolio adjustment

One of the most interesting features of our solution aims to integrate is that, since risk is the factor to be optimized, it allows the addition of auxiliary constraints that dynamically restrict the risk interval in which a portfolio is accepted as optimal.

In other words, depending on the client investor's profile, we aim to provide them with a **portfolio adjusted to their risk tolerance**. To this end, we pretend to ensure that the total risk balance of this portfolio is always between certain values. Determination of these values is explained later on.

Calculating the risk of each fund individually is an unbearable computational cost, so we studied the 5 types of funds IronIA offers, their characteristics, ideal investor profiles and behavior:

Monetary Funds

Monetary Funds[21] are fixed-income investment models that aim to assume the lowest possible risk, preserve capital and drive profitability on the evolution of capital market interest rates.

- High liquidity.
- High availability.
- Lowest risk exposure of all investment funds.

Fixed Income Funds

Fixed income funds[22] offer a balanced choice to the investor, by selecting assets that are more stable over time and less volatile to sudden changes.

- Low-medium diversification.
- Relatively low return.
- Low risk, but possibility of default.

Alternative Funds (Hedge Funds)

Alternative funds are highly specialized, state-of-the-art investment vehicles that seek to always deliver absolute returns with hedging strategies. They are not affected by whether the market goes up or down because they can bet on the downside of any other asset.

- High performance only in uncertainty situations
- Reduced general risk, depending on the quality of the strategy.
- Selling is always dangerous.

Mixed Funds

Mixed funds are those that invest in both fixed income and equities. Being a product with great flexibility in terms of its investment portfolio, it is less interesting for our quantitative approach to the problem, since our pure optimization model sacrifices customization potential.

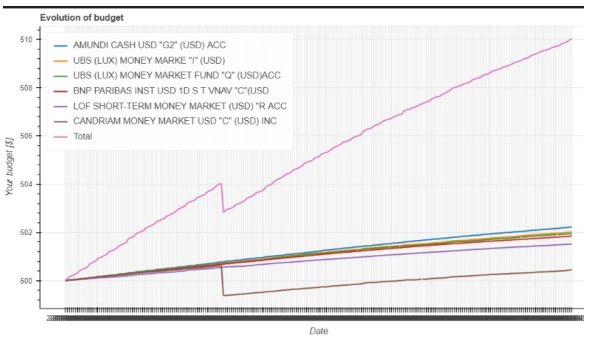
Equity Funds

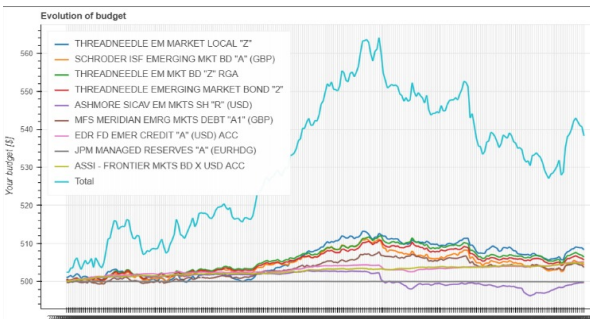
Equity funds can offer the investor significant growth and profitability options, both in the long and short term, but do not guarantee any particular return in the future. The most common for this type of investment are stocks.

- High diversification, exposure to global industries.
- Historically higher yields.
- Riskier drawdown potential, recommended for more aggressive profiles.

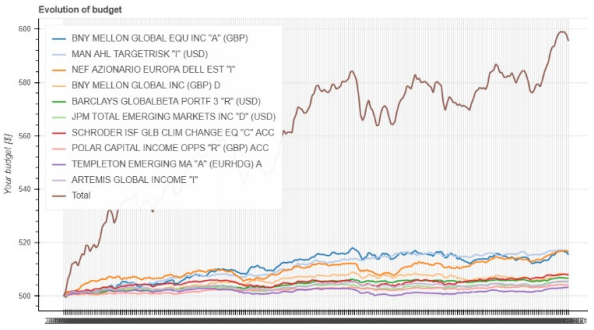
With this overview it is possible to estimate the order of risk in which these five funds could be allocated, but we will address this determination by weighting also the quantitative factors, studying all the categories segmented performance and volatility experienced over time.

MONETARY FUNDS	FIXED INCOME FUNDS
Volatility: **0.0026264** Total YoY return: **2.0%** Optimal fund number: **6**	**Volatility:** 0.054261 **Total YoY return:** 7.66% **Optimal fund number:** 9



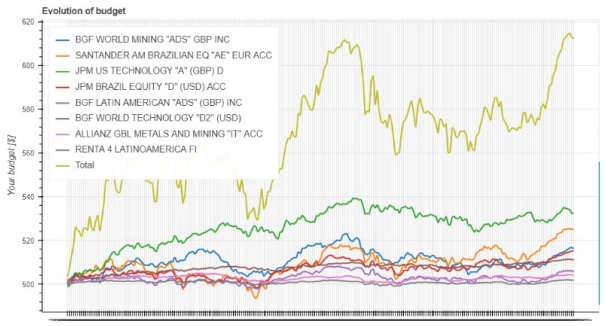


ALTERNATIVE FUNDS	MIXED FUNDS
Volatility: 0.021318 **Total YoY return:** 3.03% **Optimal fund number:** 10	**Volatility:** 0.064259 **Total YoY return:** 19.11% **Optimal fund number:** 11



EQUITY FUNDS

Volatility: 0.149200
Total YoY return: 22.44%
Optimal fund number: 8



The aforementioned observations have been considered to qualitatively assess the risk that each of these types of funds may represent. In addition, combined with the numerical results obtained by evaluating their segmented and grouped performance, we have consequently derived the following risk ranking, from lowest to highest:

Monetary Funds	Risk 1
Fixed-income Funds	Risk 2
Alternative Funds	Risk 3
Mixed Funds	Risk 4
Equity Funds	Risk 5

Balancing technique

As commented previously, it has been designed a system that profiles each user according to a risk assessment approach. This will be detailed when explaining the MVP implementation. Now the focus will be in the balancing algorithm used to tailor the optimization result to the client's risk profile.

Once we have all the risk profile data, according to the score that the client obtains from the risk assessment, we attribute a risk range through which his portfolio will move. The idea is to add a new constraint which will regulate the mean risk of the portfolio.

Since every fund belongs to a risk level according to its type, as explained above, we perform the dot product between the vector of weights and the respective vector of Risk Levels. In this way we multiply the estimated weight of a fund by its risk level, and we restrict the result of this dot product to the below shown intervals (which depends on the risk assessment evaluation deployed in the final implementation).

Result of the risk assessment	Interval constraining the risk mean
Minimum risk profile	$R \in [1, 2]$
Mid-low risk profile	$R \in [2, 3]$
Mid-high risk profile	$R \in [3, 4]$
Maximum Risk profile	$R \in [4, 5]$

The addition of this new constraint has greatly improved our Minimum Viable Product, since we are able to generate quite different Portfolios according to the Client's answers.

Multi-characterization of global markets

Introduction and key ideas

The world economy is characterized at this time not only by technological advances but also by the phenomena of **globalization**, which has increased these last decades with the entry of new agents into the world market.

Therefore, today more than ever, financial markets can have sudden changes due to some of the variables that are related to what is happening on the planet. From elements that affect the economy of a large nation, to industrial breakthroughs, they are enough to produce significant changes. Although not all the factors that influence the stock market are openly evident, some of them can be studied, and we believe that they should be taken into account to generate the best strategies, and adjust our optimization model to a framework that is real, variable and endowed with nuances that can be game-changing.

As a result, we are evolving our algorithm in order to be prepared for these changes in the global market ecosystem. In this way, this tool will be conceptually prepared to offer a portfolio recommendation adjusted not only to the client's risk profile, but also to the global situation that is being experienced at that precise moment, based on historical data on the evolution of the markets under different circumstances. We call these circumstances **variables that characterize the market.**

As mentioned above, there are hundreds or thousands of factors that condition or represent the behavior of markets. Therefore, the first important challenge is to define a characterization vector, made up of variables that, in our opinion, speak of the world economies and their markets, given our limited financial, economic and monetary knowledge.

Market evaluation

Macroeconomics

The term macroeconomic variables[23] is an expression used in macroeconomics, which represents economic and monetary aggregates. Macroeconomic variables refer to the economy as a whole of the country or a group of countries.

The study of macroeconomic variables aims to discover what type of economic activity a country, or region of interest, has and how it will evolve. In order to carry out these statistics, some indicators are taken into account through which the economic, financial and monetary situation of the region will be known, and where the specific economic area is heading.

The most important macroeconomic variables are as follows:

- Gross Domestic Product
- Inflation
- Exchange Rate
- Unemployment
- Public Spending
- Interest Rate

We have conducted a study on the impact that these variables have on the evolution of the financial markets and the macroeconomy, and we have decided to keep the 3 factors that we consider most important: GDP, inflation and exchange rate.

Gross Domestic Product

Gross Domestic Product[24] is the value of all final goods and services produced in an economy during a given period of time, usually a year.

Gross Domestic Product is one of the most important variables in macroeconomics, because it is used as a measure of economic activity, and its value per capita indicates the welfare of the population. In fact, the proportional variation over time of the Gross Domestic Product is called Economic Growth, which is exactly what we are looking for.

Inflation

Inflation[25] is the proportional variation of the consumer price index over a period of time, usually a year or a month. A high inflation rate is detrimental to economic growth because it affects relative prices, resulting in an inefficient allocation of resources in the long run.

A negative inflation rate is also undesirable because, in addition to affecting the allocation of resources, it causes people to use capital as a form of savings that increases in value over time.

Exchange Rate

All economic sectors that produce goods and services that can be imported or exported, i.e. tradable goods, are affected by changes in the exchange rate. The exchange rate[26] influences the international competitiveness of the various economic sectors of an economy, and therefore, economic growth.

Indexes

A financial index[27] is an index that reflects the evolution of the prices of a set of financial assets listed on a given market.

For example, in Spain the benchmark stock market index is the Ibex 35. This index groups the 35 Spanish companies with the highest capitalization and trading volume. In the Ibex 35, the evolution of the price of the Spanish market can be observed in a representative and approximate way.

In other words, an index provides information on what the financial market is doing at any given moment, whether it is for a country, a geographical area or a sector. The indexes can be composed of all the securities that make up a market or of a group of them that represent it according to various factors such as market capitalization or trading volume of the securities.

The underlying securities that make up the indexes may vary from year to year. A committee reviews the stocks, both those that make up the indexes and those that are on the verge of entering, and depending on the aforementioned criteria, makes changes to their composition. They represent therefore powerful benchmarks, updated and competently designed to be faithful to the financial and economic reality of the planet.

These indexes are usually associated with the representation of a country's economy, but they are much more than that. There are different types of indexes that provide evaluation angles for a given region or market:

1. According to **geographic origin**:

- **Global**: composed of financial assets from all over the world.
- **International**: composed of financial assets from different countries or regions (Europe, America, Asia, emerging countries...).
- **National**: composed of assets from a specific country (Spain: Ibex 35, France: Cac 40, Germany: Dax 30...).

2. According to **type of assets**:

- **Variable income**: composed of financial assets whose price varies (equities).
- **Fixed income**: composed of financial assets whose price does not vary (bonds and debentures).
- **Commodities**: gold, silver, cocoa, oil, wood...

3. For stocks, according to the **type of companies**:

- **Sectoral**: composed of companies in the same sector.
- **Intersectoral**: composed of companies from different sectors.

These stock market indexes indicate in a simple and graphic way the behavior and trend of a specific financial asset, region or industry in the world. Global indices, along with U.S. indices, are benchmarks for many types of funds. Actively managed funds focus on them with the objective of outperforming their returns, and passively managed funds aim to mimic their compositions, and therefore also their returns.

The importance of an index is frequently related to the wealth produced in the country or countries that make up the index (same applies for industries, or asset types). Although the generalized opinion about the stock market is that it is a speculation game, the reality is that financial markets maintain a very close relationship with the economy, and we will take advantage of this characteristic in this stage of the project to define the variables we need.

Specifically, we will focus on two index providers, due to their reliability and historical recognition: MSCI and, to a lesser extent, Bloomberg. In addition, we will incorporate other general indexes that are generic and widely adopted as benchmarks, mainly from the United States.

MSCI indices

MSCI[28] are indices developed by MSCI Inc, formerly Morgan Stanley Capital International, hence the acronym. They are currently used as a reference for many funds and as a benchmark to evaluate the performance of most actively managed funds. Part of their success derives from the fact that they are easy to compare, since they use the same methodological basis. This makes them ideal for tackling our problem.

Nonetheless, they take into account the particularities of each market to faithfully represent its evolution, which fits perfectly with the need for differentiation that we are facing. Actually, MSCI indexes seek to cover up to 85% of the capitalization of the market they represent. To this end, 70% of the assets that make up the index are large-cap companies and the remaining 30% are divided equally between mid-cap and small-cap companies. Once again, this is exactly the balance between variability and representativity our approach needs.

Bloomberg indices

Bloomberg has a powerful service that offers news, communication between terminals and different applications that facilitate the work of those involved in the financial markets. Most companies in the financial world subscribe to Bloomberg's service and their terminals have become an essential part of their business. They also elaborate indices, and given their high reputation in the financial world, we have decided to select some of their contributions to include in our characterization.

To wrap up, we believe that the effective and rational aggregation of these indexes can be a differential element to characterize the market. From the point of view of the evolution of the main companies and assets, we seek to cover all possible angles and include the main representatives of all the types listed above.

Government bonds

A government bond is a type of bond issued by a country and its government as a means of financing and which earns the holder fixed interest for the duration of the bond until maturity. Government bonds are generally considered low-risk investments, as the chance of a government defaulting on its loan payments is usually low. Even so, this can happen and a bond with higher associated risk will normally trade at a lower price than a bond with lower risk and a similar interest rate. This inherent characteristic, we believe, is key to determining the solvency of a government and, therefore, we have decided to use it as a representative metric of the political and monetary conditions of the world's major nations.

Commodities

Gold is a commodity that meets all the requirements as a currency and as a store of value. Annual production is limited for physical reasons of extraction, so it cannot be devalued. This makes it a key indicator when assessing market sentiment; when investors are afraid of volatility or downward trends in financial markets, the price of gold rises as it is bought as a moderately safe store of value. We have decided to use this characteristic, along with the same reasoning but to a lesser extent for silver, to represent retail confidence in the market at any given time.

Variables selection

Macroeconomic variables

Yearly GDP by geographical area		
Africa Eastern and Southern	United States	East Asia & Pacific
Latin America & Caribbean	Africa Western and Central	Europe & Central Asia
Daily US Dollar exchange rate by main forex pairs		
USD/AUD (Australia)	USD/CHF (Switzerland)	USD/INR (India)
USD/CAD (Canada)	USD/JPY (Japan)	USD/GBP (Great Britain)
USD/EUR (Eurozone)	USD/CNY (China)	
Monthly US inflation		

Indexes

In the case of the indices, by narrowing our search to a very specific group, we have the advantage that they all share similar characteristics. Therefore, we were able to access daily data for about a decade for the **opening**, **closing**, **high** and **low** price of each of the indices, with relative consistency and without an abundance of outliers.

By industry (global)		
MSCI World (all)	MSCI World Industrials	Bloomberg Agriculture
MSCI World Energy	MSCI World Telecom	Bloomberg Brent Crude
MSCI World Financials	MSCI World Utilities	Bloomberg Natural Gas
MSCI World HealthCare	MSCI World Consumer Goods	
By market capitalization (global)		
MSCI World Large Cap	MSCI World Mid Cap	MSCI World Small Cap
By geographical area		
MSCI AC Americas	MSCI AC Pacific	MSCI AC Far East
MSCI AC Asia	MSCI AC Europe & Mid East	
Other relevant indexes		
S&P500: DOW JONES: Nasdaq: EuroMTSEurozoneIG7-10YGovernment: CBOE Vix Volatility: US 10Y Treasury Vix:		

Other variables

For the price of commodities the same situation as with the indices has been presented, so the abundance of data has been satisfactory. Also for bonds, having the daily closing price on practically every day of the period decided (explained below).

5-year government bonds by nation (G7 + 2 extra)		
Canada	France	Germany
Italy	Japan	United Kingdom
United States	India	China
Commodities		
Gold	Silver	

Scraping methodology

Data sourcing

Our initial interest was in gaining access to some specific stock market APIs, either from information services or trading platforms. The reason for this is that stock APIs can connect us relatively efficiently to accurate and relevant data sources. On top of that, obtaining stock market data via APIs is simple, consistent and predictable in a properly structured format.

However, we quickly realized that many of the APIs we found were focused on the financial aspect of the markets, whereas our approach encompasses macroeconomic variables as well.

On the one hand, getting access to trading-related quality APIs in many cases entailed an economic cost, or a difficulty in applying for them that was not worth it considering that we would not have all the data. On the other hand, our search did not yield many results that, free of charge, incorporated all the data we needed. In addition, the idea was to minimize the number of endpoints needed to set up the scraping system, for efficiency and management of the team's time and resources.

It was then decided to extend the search to large-scale financial, economic and monetary information free service platforms with no public API, but which would allow the scraping of information from their websites. The goal was to overcome this challenge by designing a manual scraper that emulated a web browser and collected the data needed from hundreds of thousands of publicly available data. This would have cost too much development time, as it is only an intermediate step in the development process.

Fortunately, we located a tool that had been designed and uploaded open-sourced to GitHub with precisely the solution to our problem: **python package investpy**. In the words of the Spanish developer, **Alvaro Bartolome del Canto**, who created and uploaded it for use:

"*This Python package has been made for research purposes to fit the needs that Investing.com does not cover, so this package works like an Application Programming Interface (API) of Investing.com developed in an altruistic way. Conclude that investpy is not affiliated in any way to Investing.com or any dependent company, the only requirement specified by Investing.com to develop this package was to "mention the source where data is retrieved from"*.

Data gathering

Although the tool we found obtained the desired information from the Investing.com page, it required further adaptation to work correctly according to our specific queries. The objective was to scrape all the information available during **each day from 2000 to 2020 for the 50 variables** explained above. In addition, we discovered that many of them not only have a daily price, but are broken down into high, low, open and close prices.

This exponentially increases the volume of data required, so developing an efficient adaptation was essential. As we did not know if we would later need to scrape new variables, an automatic system was designed in which the name of any asset, variable or index is entered, together with the type of financial product it is, and it automatically compiles the collected data, processes it and saves it in the specific format for that set of variables. The end result is 50 CSV files with thousands of rows, corresponding to the value of each variable each day of the interval.

Data preprocessing and fine-tuning

After the market characteristics have been acquired we can start performing some preprocessing.

On the one hand, the granularity at which some prices were reported was an unnecessary additional complexity. While it is true that over the course of 24 hours the price of an asset varies significantly, there are different ways of interpreting what the price of the day is. In many cases we had the freedom to choose between the highest price of the day, the lowest price of the day, the price at *00:01* or the price at *23:59*. In our case, only 1 price was interesting, albeit an approximate one for the day. Therefore, it was decided to **average the high and low**, incorporate it as a column for each day and drop the other prices, for storage optimization reasons.

Data augmentation

Time interval amplification

At first we thought about getting the characteristics for one year and for **9, 5, 3, 2 and 1 months ag**o, but at last we thought that instead of having one variable for each period of time (where there was correlation between them) we decided to create some weights.

It was decided to give higher weights to the closest past and smaller weights to the further past. The reason behind this is because it is more probable that the nearest future is more similar to the closest past than to the further past.

Aggregation statistics

The next step was to design specific metrics to posteriorly perform clustering, as explained in coming sections. The first statistic proposed was the **mean**, but when clustering was being tested, we noted that the result consistently resembled a time series, which makes sense because we were getting the average through the same past but with different period lengths. This issue is here visualized.

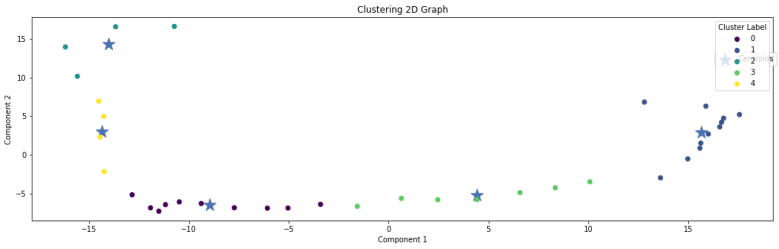


Figure 18 (clustering obtained with mean metric)

The final decision was to use the **variance** and the **difference**. These variables performed really well with the available data, and the results seemed to be interesting and insightful.

Normalization

Now that the data has been treated with the necessary computations, it is normalized because different ranges of values were present.

Data visualization

Here we have a visualization of all the market characteristics we have scraped and with difference and variance matrix. We can see that all the variables are uncorrelated , but middle to down right we can see that there are more variables more correlated. This is because they are indexes of different countries that are correlated with the world indexes.

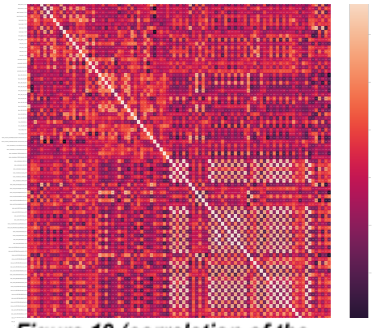


Figure 19 (correlation of the market characteristics difference and variance matrix)

Proximity-based portfolio assimilation

The next step was to use these variables in some way that would actually allow us to differentiate the characteristics of different time periods, segment them and differentiate them as clusters, or as similar observations, depending on the approach. The idea was to input a new vector of updated market variables so that the current market moment would be assimilated to one of the clusters, or observations, previously studied, so that we could have an optimal portfolio previously calculated.

Clustering: first approach

The first approach we have decided to develop is **clustering[29]**. We have decided this approach because we think that making a proper characterization of the market will help our robo advisor to obtain the most suitable funds for the user and the ones with the best performance in the future when the user is going to invest.

There are a few advantages to using a clustering algorithm in this context. First, it can help to identify patterns in the data that might not be otherwise apparent. This could be valuable for understanding the factors that are influencing the markets and for making implementations that take into account how these might change in the future. We want to verify whether these variables are able to clearly characterize and differentiate some time phases that share traits.

By calculating optimal portfolios for these clusters, we can feed new data, updated up to date, to be assigned to a cluster for which we already know the optimal compositions. This would allow the algorithm to produce a portfolio that is tailored to the current market conditions.

There are also a few disadvantages to using clustering in this context. First, it can be time-consuming to run the algorithm and to analyze the results. Additionally, it can be difficult to interpret the clusters that are generated. This could make it difficult to determine which variables are most important for understanding market changes.

Clustering methodology

After the normalization, we have proposed the option to use some dimensionality reduction techniques in order to avoid correlation between the remaining variables. In this case, we have chosen to use Principal Component Analysis[30] with enough components to keep the 90% of variance explained. Furthermore, for some visualization purposes we have also used some 2 components PCA. This is the final step before computing the clusters.

Clustering design

There are many different algorithms for clustering data. K-Means[31] with PCA is a simple algorithm that can be used to cluster data with many different types of distributions. It works by dividing the data into two parts: the first part consists of the data points that are close to each other, and the second part consists of the data points that are far from each other. The algorithm then finds the best way to divide the first part into two sub-parts, and repeats this process until all of the data points are divided into clusters.

PCA is a very efficient algorithm and can be usually applied to a clustering algorithm. It improves the performance by identifying and extracting the principal components of the data.

The principal components are the dimensions that account for the most variation in the data. It also works well with noisy data, and can often find clusters that other algorithms cannot find. PCA is especially useful for clustering problems with high-dimensional data, which is data that has more than three dimensions, as in our case.

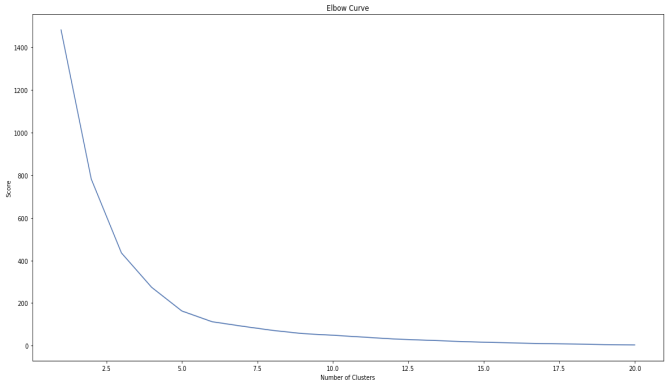


Figure 20 (elbow curve to decide the number of clusters)

An elbow curve, as shown here, is the graphical representation of the distribution of data points in a scatterplot. It will be used to decide the number of clusters to perform clustering. The elbow curve is created by plotting the number of data points in each cluster against the size of the clusters.

A bend in the curve signals the point where the number of clusters starts to level off, indicating that additional clusters would not result in a significant change in the distribution of data points. The number of clusters at this point is then used to perform clustering.

In this case, the decision is to parametrize the number of clusters with **5 centroids**.

Clustering testing

We had previously computed the dates associated with each dot, so later we could use those dates to test our cluster. Then we have trained our “cluster” period and we have assigned that value to its corresponding place in the test value. With this approach we are able to train within the period of each cluster.

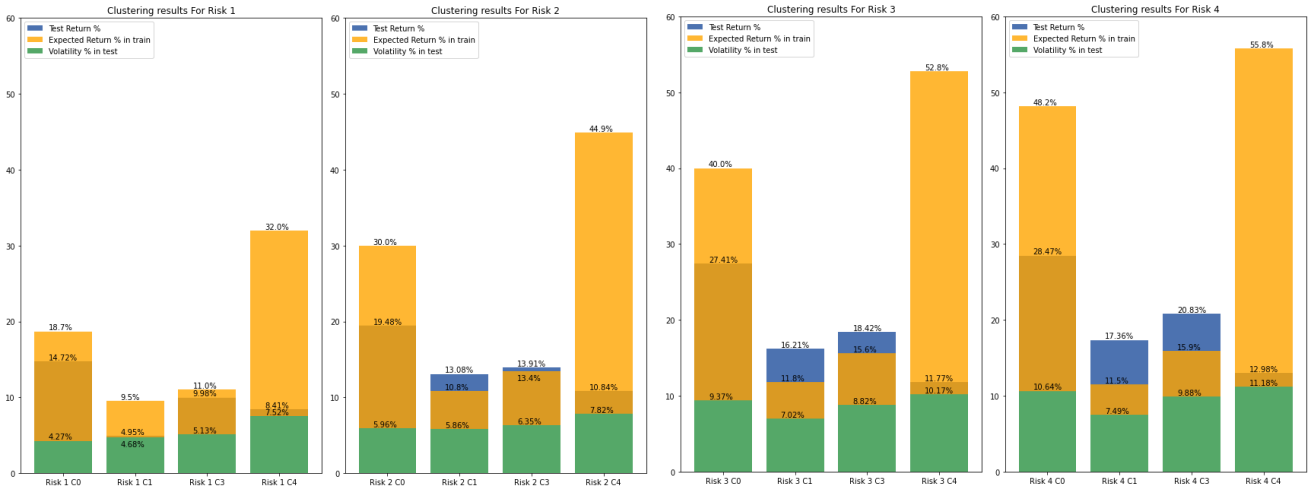


Figure 21(clustering results for risks 1 and 2)

Figure 22 (clustering results for risk 3 and 4)

In this first way we have faced some problems. We are taking the dates within each cluster but they are not always consecutive. This would not allow us to give a solution. In order to solve it we have taken the minimum date and the maximum date of that period. Then we trained in that period. Even though there are other cluster periods in the middle, we have achieved a good solution.

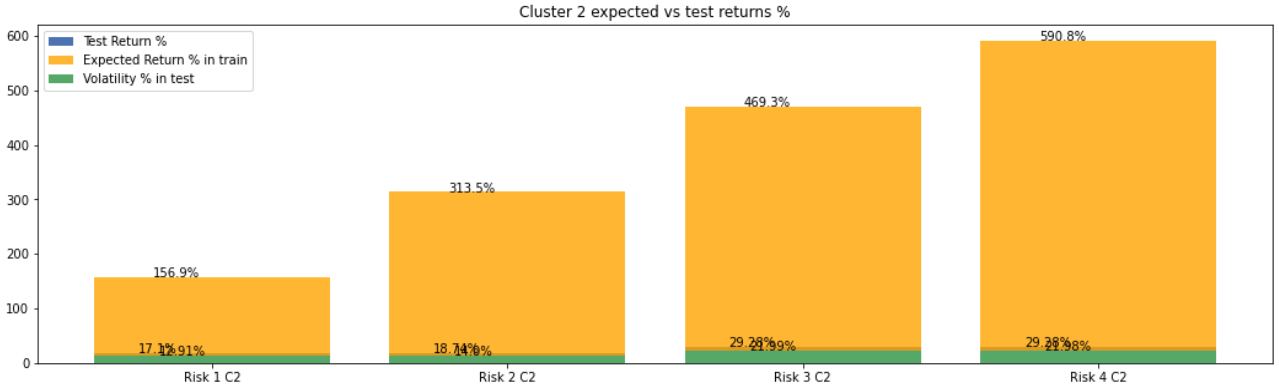


Figure 23(clustering results for all risks in cluster 2)

In the figures 21 and 22 we can see that for clusters 0, 1, 3 and 4 we are obtaining quite good results. Even though the expected return on the train is higher than the test return in some cases, they are not really far away. But in figure 23 we can see that cluster 2 is performing really poorly. This is because cluster 2 is just taking two months so we are dealing again with lack of data.

Distance matching: second approach

In this second approach, instead of assigning each test observation to a particular cluster, we directly compare the distance from each test feature vector with all the features vectors on the train. By taking the euclidean distance between vectors we get which train sample is the closest to the test one. Once we have this time point, we just calculate the portfolio based on the previous year of the train sample and put the same fund allocation to the test.

After trying this method, we saw some flaws. First, as the train and test can now have some time discrepancy, the algorithm thinks that the increment between years is huge so that showing us expected return of more than 500% (which is obviously impossible). Second, we have analysed that using just the closest year to the one in the present does not make much sense as it is more useful to have the complete time series data and giving more weight to the closest past.

Comparing approaches: clustering conclusions

After using these two methods we can say that we are getting better results with the first approach than with the second one. This is because in the second approach we are not taking into account the clusters, we are just taking into account the nearest distances of the dates from the test to the train. So all the market characterization and the metrics would have been calculated in vain.

To conclude we will say that given the current state of the clustering, in which we will have to get deeper, and the lack of data clustering for now is not a good approach. But in the future when each cluster will be labeled, probably clustering is going to be a really good option to work with in these kinds of problems.

IroAdvisor_v1.01: Minimum Viable Product

Aim of the MVP

A minimum viable product (MVP) is a product with just enough features to **satisfy early customers** and to **provide feedback** for future development. The goal of this MVP is to get our product into the hands of users as quickly as possible, then gather feedback and make changes based on what they say. This tool could be released as a beta version to a limited number of users, or it might be made available as an open-source project. In either case, the goal is to get feedback from users so that the idea can be improved.

It is important to note that the algorithm used in it has been **trained with data from the years 2016-2018, and tested with 2019 data**. There are a number of ways that this minimum tool could be improved and scaled afterwards. One way is to focus on developing a more user-friendly interface. Another way is to focus on developing more features for the product. Additionally, the team can focus on improving the product's performance and scalability. One way to do this would be to automate a data collection system that would keep all parameters up to date, both on fund performance and on the variables that macro-characterize the market.

Risk aversion assessment

As a starting point, the team has designed a 7-question test. This **risk tolerance questionnaire** is important because it helps investors figure out how much risk they are willing to take on with their investments.

By taking a risk tolerance questionnaire, we can get a better idea of what the user should be looking for in an investment portfolio, and then apply the mathematical constraints to the optimization model to limit the allowed risk interval, as explained in previous sections.

These questions have been approached from the **psychological scope** of the main investor profiles, and the importance of each of the questions and their answers is detailed below.

1. If you had to choose between more job security with a small pay increase and less job security with a big pay increase, which would you pick?	
A. Definitely more job security with a small pay increase B. Probably more job security with a small pay increase C. Probably less job security with a big pay increase D. Definitely less job security with a big pay increase	If a user chooses more job security with a small pay increase, this indicates that they are risk averse and would rather have a stable job and less monetary compensation than the alternative. As an investor, this individual would likely be more inclined to invest in low-risk securities.
2. Imagine you were in a job where you could choose to be paid salary, commission, or a mix of both. Which would you pick?	
A. All salary B. Mainly salary C. Mainly commission D. All commission	If the user is mainly interested in commission-based income, it may suggest that they feel better about uncertainty and thus are more aggressive in their investment approach.

3. When investing, you are primarily concerned about:	
A. Not losing capital (combat inflation) B. Keeping the capital you invest and making a bit more C. Relatively consistent growth over time D. Making as much capital as possible from your investments	Someone who is concerned primarily about not losing capital may have a more conservative investment strategy, while someone who is focused on making as much capital as possible may be more willing to take on more risk.
4. Of the following investments, which of the following scenarios would you be most comfortable with:	
A. You can lose down to -2%, and gain up to +9% B. You can lose down to -7%, and gain up to +13% C. You can lose down to -15%, and gain up to +26% D. You can lose down to -31%, and gain up to +48%	Those users who are comfortable with scenarios in which they can lose more capital, but have the potential to make more capital, may be more aggressive investors. Those who are more comfortable with scenarios in which they can lose less capital, but have the potential to make less capital, may be more conservative investors.
4. Which of the statements better reflect the way you feel in situations in which you have little to no control over the outcome?	
A. I tend to panic and start making bad decisions. B. I feel powerless and start overthinking. C. I get a bit nervous but I let the situation develop. D. I remain completely calm.	As our algorithm is based on the optimization of risk measures of the conditional drawdown family, calm in low control situations usually indicates high drawdown tolerance and, therefore, lower risk aversion.
6. Back in 2008, the market took a major hit and stocks went down nearly 30%. If you had owned stocks at that time, how would you have reacted (or your real reaction if you actually did have capital invested).	
A. You prefer losing some capital than risk losing any more: sell everything! B. Just to be safe, you prefer to sell some of your assets and keep a small part. C. Do nothing! Let the market flow and see how it plays. D. Buy more, now that the price is low!	An investor who reacts angrily and sells everything when the market takes a hit may be more risk-averse than one who does nothing or buys more when the prices are low.
7. Your current age is...	
A. Over 50 B. Between 35 and 49 C. Between 25 and 34 D. Under 25	Age can be a useful indicator for investors when assessing risk. The older an investor is, the more likely they are to have a shorter time horizon until retirement, and thus may be more risk-averse.

Once answered, each answer A will add 4 aversion points, 3 for B, 2 for C and 1 for D. A risk interval will be assigned according to this parameterized in the optimization, as discussed above.

Tool deployment

SDK selection

For developing the MVP online application, we have used a tool called Streamlit [3]. Streamlit is a Software Development Kit (SDK) created specifically for building Web apps coded in python. It is a very useful and easy-to-use tool, as it makes it possible to write an app the same way a python script is coded. It is the main SDK used for developing Machine Learning Applications. Because of these reasons, it has been decided to use it for developing our Minimum Viable Product. It is more than enough to materialize our project into an application that is already useful for customers. Of course, in case we want to develop our project further, we should look for other software to develop a more professional app, thus also improving the UX/UI customization potential.

App implementation and usability

We wanted the application to be as flexible as possible for the end user, so features can be tested and feedbacked. This is why many model customization options have been made available. The user inputs will then be fed into our algorithm parametrization and it will generate a Portfolio accordingly. Our app consists of three main sections:

Sidebar

In it, the user is asked to decide which Risk Measures wants the algorithm to use (CVaR, CDaR, MAD, ML, or Sharpe Ratio). It is recommended leaving CVaR as default if the user is not an advanced investor. Additionally, the user can input the amount of capital willing to invest. For this purpose, the Streamlit Select Box Buttons [3] are used.

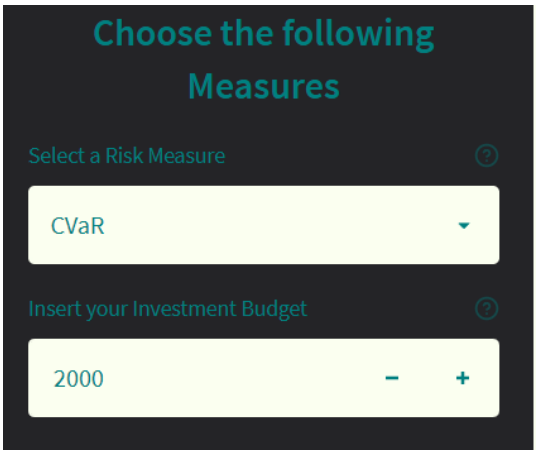


Figure 24(Sidebar to select risk and investment budget)

We also thought of adding that the user could decide the **percentage of risk** that our algorithm should take. However, since it is a key input, we have **decided to fine-tune this value by ourselves**. We use the Streamlit Number Input [4].

Figure 25(Selection of the risk level)

Therefore, the way the user will choose the risk tolerance is based on their answers to the questionnaire.

Questionnaire

In order to create the questionnaire in a way that the user can give us the information for assessing their risk profile, the Streamlit Radio buttons [5] are used. They return the string of the answer clicked, thus what we do is to search in the returned string the letters “A.”, “B.”, “C.”, or “D.” in order to add the corresponding amount of points from each answer, as explained previously.

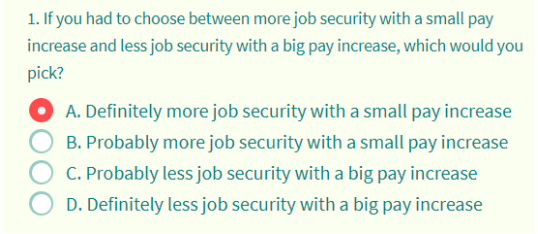


Figure 26 (First question of the questionnaire)

Once the user has finished filling out the questionnaire, he must click on the select box shown in the image. This has been done because of how Streamlit works. If we

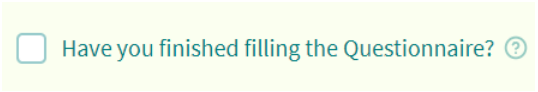


Figure 27 (Check button to finish the questionnaire)

don't use this procedure, the program would be recalculating the portfolio whenever the user answers a question. Thus in order that he can answer the Questionnaire in a comfortable way we have decided to insert this checkbox, which once clicked will allow the program to continue running and calculate the results with the new risk profile that we have drawn based on the answers of our user. The way we map the answers of the user into an input of our Algorithm has been already explained in the balancing technique section.

Clarify that the user can fulfill the questionnaire if he desires to (we recommend filling it, especially for those financially inexperienced individuals). In case he doesn't want to fulfill it, all funds will be taken into account in the computation of the Portfolio, obtaining worse performances (knowing more our user improves our portfolio recommendations).

Feedback / Results

In this final section, we show the results of our Portfolio Optimization Algorithm, which has taken the user input, plus some hyperparameters that we have decided (explained in the next section).

What we show to the user first is two charts. The first one is a **line chart** in which the user can observe how his Portfolio will behave in the following year. In this case, the outputs are of how the user Portfolio will behave in 2019. It shows the annual performance of all the funds of the Portfolio as well as the annual performance of the Portfolio (**total** in the legend). The second one is a **Pie Chart** which shows the percentage of capital invested from its budget. Funds are shorted from higher inversion to lowest inversion.

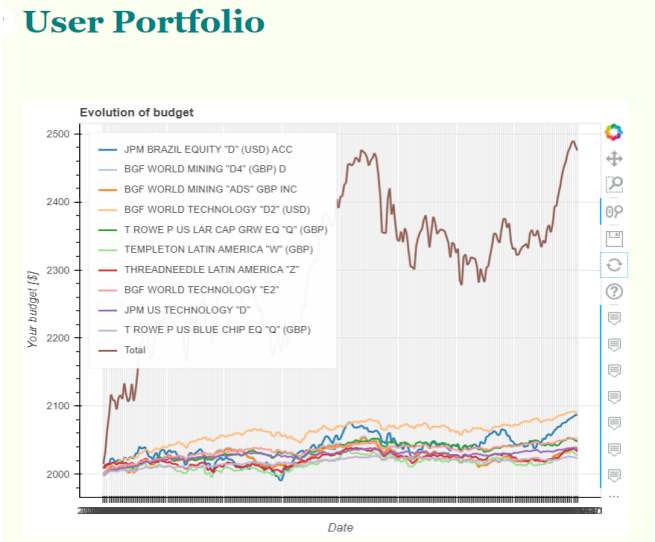


Figure 28(Evolution of the budget for a user)

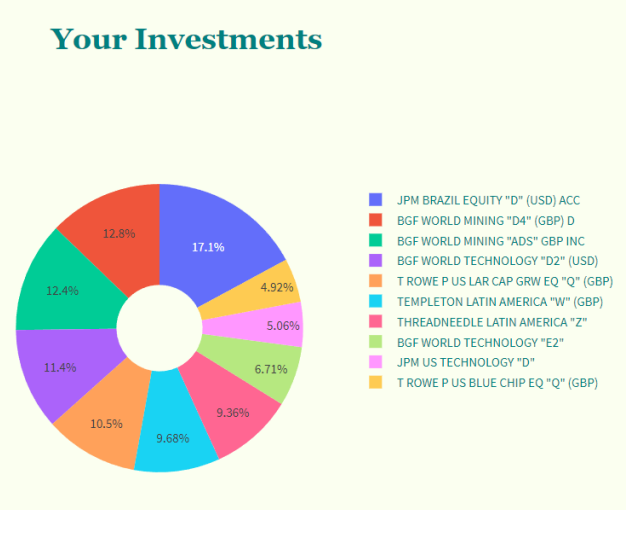


Figure 29(Funds selected for a user)

Below these two Charts, we show the user several metrics which for sure are of his interest. We show him the **volatility** of it´s portfolio which will be higher or lower depending on how risky we consider the customer to be. Also we show him the **Total Returns** which is a percentage estimated by our Algorithm once the Portfolio has been created. Finally the **capital Obtained** which is the expected dollars he will obtain from its investment and the **Total capital** which is the capital obtained plus the budget he has invested.

Names	Benchmark Id	Budget Inversion	Risk Level	Category	Benchmark	Morningstar Category Id
THREADNEEDLE LATIN AMERICA "Z"	5e809c91c6805b4d89f1686b	187.256$	5	Renta Variable	MSCI EM Latin America Net Total Return USD Index	EUCA000524
JPM BRAZIL EQUITY "D" (USD) ACC	5f5f81bba2871d8612946884	342.576$	5	Renta Variable	MSCI Brazil Net Total Return USD Index	EUCA000699
BGF WORLD TECHNOLOGY "D2" (USD)	5e32b9f59fd58c15144cdcfa	227.676$	5	Renta Variable	NASDAQ Composite Index	EUCA000542

Figure 30(information about the funds selected for the user)

Finally, we show the user further information from the funds which compose his Portfolio. The most important columns are; the **Budget Inversion** (in case the user wants to know the exact amount of capital invested per fund), the **Risk Level** (in this way the user can know which funds are riskier and which aren't in case he wants to make certain modifications), and the **Morningstar Category ID** (in case the user wants to get further information from Morningstar).

That would be all with respect to the Application. It is basic but could be very useful, especially if more functionalities are added, such as being able to modify your portfolio by removing or adding the funds of your choice.

Hyperparameters and other Function Entries

In addition to the inputs we get from customers and the test and filtered training data (**filtered** with the output funds of the **Hierarchical Computing Function**), our optimization function also takes other inputs that are important for its proper functioning. The most important are:

gamma

When optimizing using the PyPortfolioOpt library, it is apparently common that the majority of the weights are assigned to a small number of funds leaving the vast majority with a weight of 0. Fortunately, this library has a regularization term, that is, **gamma** (concretely uses L2 regularization). What happens is that by adding an additional cost function to the objective, the optimizer can be "encouraged" to further diversify the different weights. After some tests it has been concluded that the best value for this regularization term is around 0.15.

min_weight

The only disadvantage of using the gamma parameter is that when diversifying the weights on the funds in this manner, some funds have very small weights assigned. To give you an idea, these weights were so low that they implied investing around one dollar in these funds (individually). Thus we decided to set a **minimum weight threshold** which we have set to 0.04.

The capital that was going to be allocated to these funds added to the capital that PyPortfolioOpt decides not to invest has been reinvested in the rest of the funds in proportion to the weight assigned to each of them.

risk

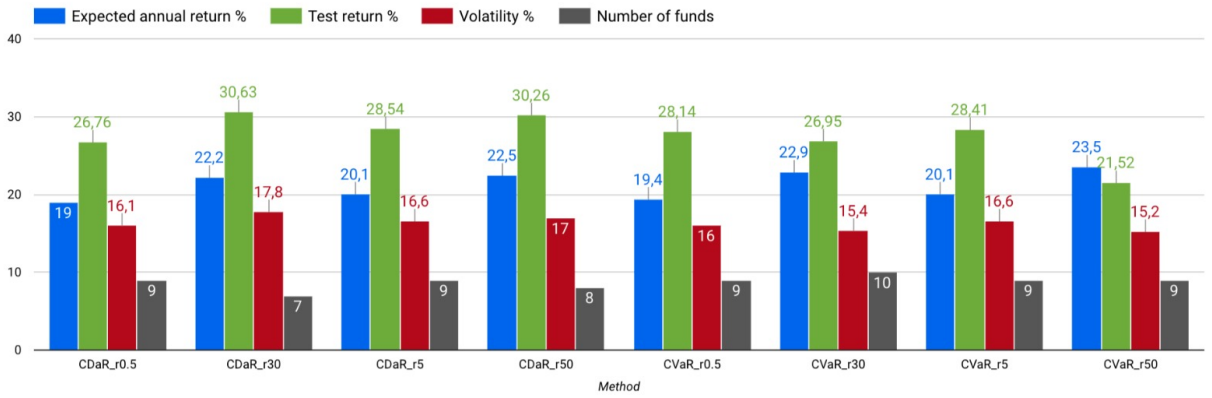
The risk Measures CVaR and CDaR in the PyPortfolioOpt library, need a risk percentage. Initially we expected that this parameter would be modifiable by the user, and that he could adjust the risk percentage as he wishes. However, the truth is that this parameter does not seem to have much influence on the risk level of the portfolios. It is true that increasing it gives riskier portfolios, but the risk variation is very low. Is because of this reason that we decided to fix it to 0.05 and to make the risk selection in another manner (using the Risk Assessment Questionnaire).

Final results

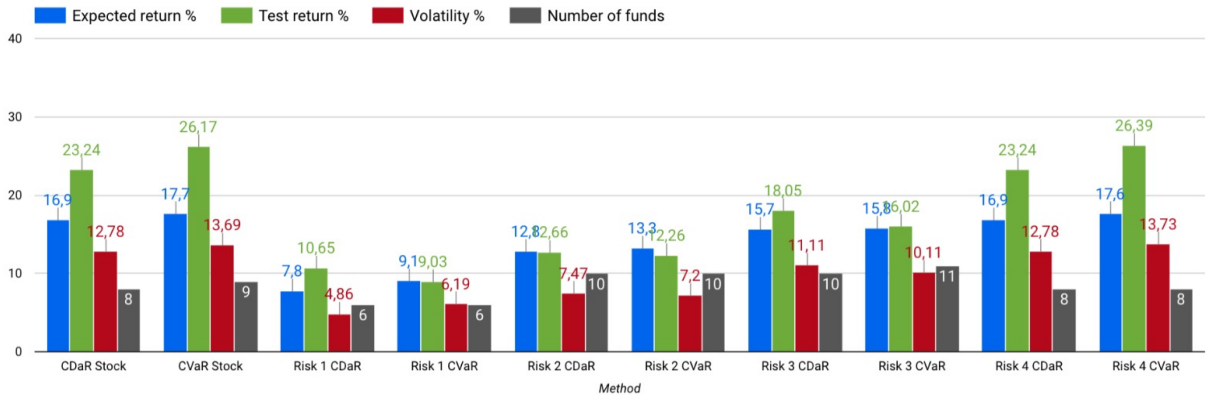
As for the final results, we have discarded the clustering approach as we don't consider it as a mature enough viable product. For this final results section we are going to compare the original approach we had with the one that has all the preprocess funds along with a more appropriate balance of risk, adjusting to each client's investor profile and consequent risk tolerance.

This significantly helps to ensure that the portfolio offered is not only mathematically optimal, but also optimal for the psychology and risk aversion of each specific IronIA client.

Previous approach

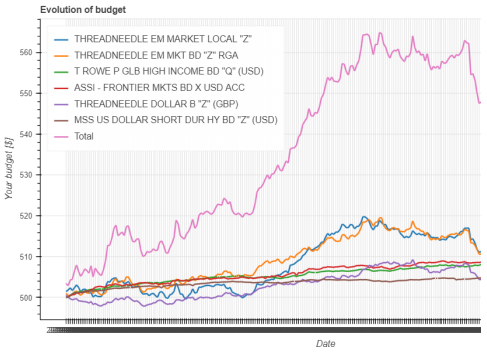
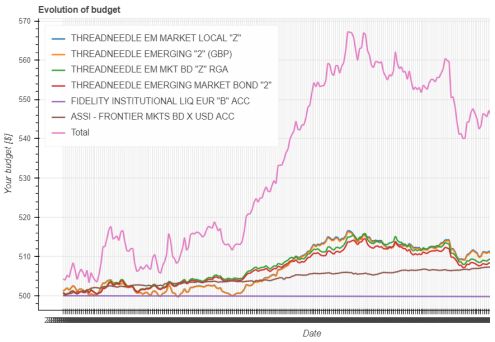
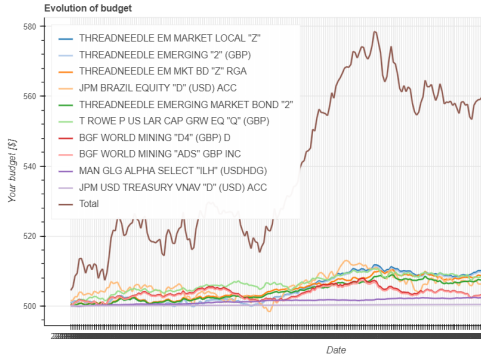
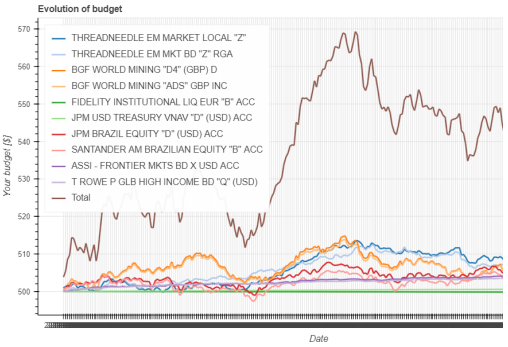
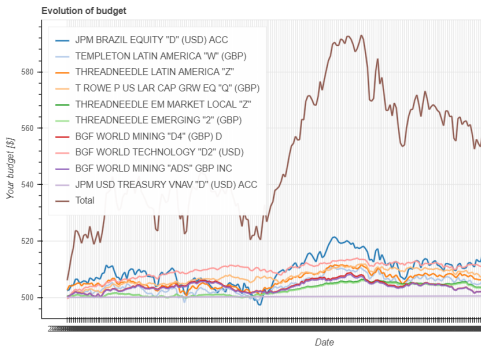
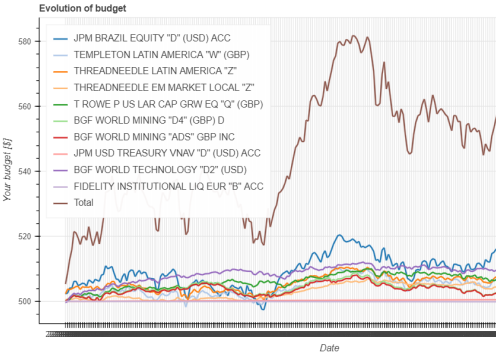
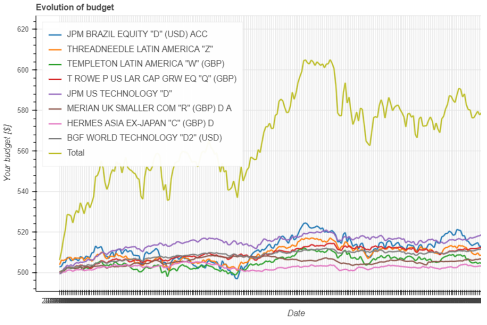
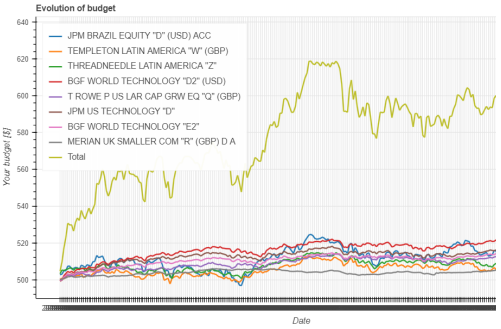


Final approach



In this second interpretation of risk management, it can be clearly seen how the risk growth curve accompanies the return curve. While it is true that there is a technically optimal investor risk profile, which is the one that should be satisfied with the optimal result of the algorithm (stock risk metrics), this segmentation helps to tailor the volatility suffered and the expected return proportionally to the risk tolerance. This, we recall, has been collected qualitatively and mathematically parameterized as described in the second phase of this project.

Below we can see the funds the algorithm has chosen depending on the risk metric and the risk level.

	CDaR	CVaR
Risk 1		
Risk 2		
Risk 3		
Risk 4		

Future improvements

While working on this project we have thought of some improvements which can be done in the future.

Data extraction

First of all as a general improvement we could design a data mining algorithm. This would allow refreshing every certain period of time (e.g. every year) in a simple and efficient way. In this project we have been working on a fixed period of time (2016-2019), but our algorithm could be applied on more current data after its due preprocessing.

Parallelize the algorithm

Another improvement will be related to the algorithm. We have thought about implementing the hierarchical computation method in a parallelized way. This will allow us to perform all possible fund combinations, which will improve the portfolio elaboration. Of course, a more suitable approach to tackle this problem directly might be migrating the whole pipeline to a parallelizable environment such as, for example, **spark**.

Enhance the clustering

We could also improve the clustering approach. In this new implementation our idea is to analyze and label each cluster. Then we will be able to identify if in those dates the funds are dropping down or going up or if they are varying or not. At last we could identify what is going to happen in the future(test).

Professionalize the web application

The last improvements we have considered are about the app. We would like to let the customers modify, save and share their portfolios. It will be great to add a way where the user can contact IronIA managers as well as adding a helper to solve doubts regarding the use or the inconveniences with the application.

Moreover, improving the app interface, either by polishing its sections in streamlit or by using a more advanced SDK with more functionalities. At last we would like to add more advanced profiling methods, in order to understand our customers better.

Final conclusions

This has been one of the first works we have faced, in which It has been proposed to us a real problem with a huge amount of data (most of which was temporal data, so we have to work with it in a somewhat different way than with other types of data).

Also, It has been one of the first works in which we have done an End-to-End implementation.

We have had to perform a lot of data cleaning to obtain a dataset that was useful to us, in addition to doing research looking for approaches that are usually followed in this type of problems or implementations that are more state of the art.

Furthermore, once we had tackled this problem using techniques found by doing research, we tried to implement a Clustering methodology conceived by us, using a large number of market descriptors.

For these reasons we consider that the realization of this work has been a very enriching experience from which we can learn a lot to face future jobs we may encounter.

We knew from the beginning that this problem was not going to be easy to solve (mainly because most of us did not have a great financial knowledge).

However, this has motivated us even more because it is very likely that in the future we will find ourselves in very similar situations in which we will have to be able to give effective solutions without having much initial background.

We consider that we have done a great job, giving an innovative solution to the problem, developing an App to make our work useful for our company's customers and devising a new way to solve this financial scenario.

References

[1] BESTE, Allison, et al. The Markowitz Model. *Selecting an Efficient Investment Portfolio". Lafayette College, Mathematics REU Program*, 2002.

[2]BAILEY, David H.; LOPEZ DE PRADO, Marcos. The Sharpe ratio efficient frontier. *Journal of Risk*, 2012, vol. 15, no 2, p. 13.

[3]KUMAR, Nand, et al. Portfolio optimization: Indifference curve approach. *International Journal*, 2014, vol. 2, no 1, p. 127-133.

[4]ELTON, Edwin J.; GRUBER, Martin J.; PADBERG, Manfred W. Simple criteria for optimal portfolio selection. *The Journal of Finance*, 1976, vol. 31, no 5, p. 1341-1357.

[5]SHARPE, William F. The sharpe ratio. *Journal of portfolio management*, 1994, vol. 21, no 1, p. 49-58.

[6]TERRAZA, Virginie; NEUBERG, Luc; LOUARGANT, Christine. Timing inconsistencies in the calculation of Funds of funds Net Asset Value. *Fundexpert*, 2006, p. 0-5.

[7]BLU, Thierry; THÉVENAZ, Philippe; UNSER, Michael. Linear interpolation revitalized. *IEEE Transactions on Image Processing*, 2004, vol. 13, no 5, p. 710-719.

[8]DONOGHUE, William F. The interpolation of quadratic norms. *Acta Mathematica*, 1967, vol. 118, no 1, p. 251-270.

[9]CHEKHLOV, Alexei; URYASEV, Stanislav; ZABARANKIN, Michael. Portfolio optimization with drawdown constraints. En *Supply chain and finance*. 2004. p. 209-228.

[10]LEAL, Ricardo Pereira Câmara; DE MELO MENDES, Beatriz Vaz. Maximum drawdown: Models and applications. *The Journal of Alternative Investments*, 2005, vol. 7, no 4, p. 83-91.

[11]ROCKAFELLAR, R. Tyrrell; URYASEV, Stanislav. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 2002, vol. 26, no 7, p. 1443-1471.

[12]KONNO, Hiroshi; KOSHIZUKA, Tomoyuki. Mean-absolute deviation model. *Iie Transactions*, 2005, vol. 37, no 10, p. 893-900.

[13]STUDER, Gerold. *Maximum loss for measurement of market risk*. 1997. Tesis Doctoral. ETH Zurich.

[14]PAPAGEORGIOU, Nicolas; REEVES, Jonathan J.; XIE, Xuan. Betas and the myth of market neutrality. *International Journal of Forecasting*, 2016, vol. 32, no 2, p. 548-558.

[15]MARKOWITZ, Harry M. Foundations of portfolio theory. *The journal of finance*, 1991, vol. 46, no 2, p. 469-477.

[16]BROWN, Ken; MOLES, Peter. Credit risk management. *K. Brown & P. Moles, Credit Risk Management*, 2014, vol. 16.

[17]GARCIA, C. B.; GOULD, F. J. Survivorship bias. *Journal of Portfolio Management*, 1993, vol. 19, no 3, p. 52.

[18]HE, Guangliang; LITTERMAN, Robert. The intuition behind Black-Litterman model portfolios. *Available at SSRN 334304*, 2002.

[19]HEERINGA, Wilbert Jan. *Measuring dialect pronunciation differences using Levenshtein distance*. 2004. Tesis Doctoral. University Library Groningen][Host].

[20]RAHUTOMO, Faisal; KITASUKA, Teruaki; ARITSUGI, Masayoshi. Semantic cosine similarity. En *The 7th International Student Conference on Advanced Science and Technology ICAST*. 2012. p. 1.

[21]CAI, Yu; WANG, Qing. Money funds manage returns. *Pacific-Basin Finance Journal*, 2022, vol. 71, p. 101682.

[22]DERWALL, Jeroen; KOEDIJK, Kees. Socially responsible fixed-income funds. *Journal of Business Finance & Accounting*, 2009, vol. 36, no 1-2, p. 210-229.

[23]BILSON, Christopher M.; BRAILSFORD, Timothy J.; HOOPER, Vincent J. Selecting macroeconomic variables as explanatory factors of emerging stock market returns. *Pacific-Basin Finance Journal*, 2001, vol. 9, no 4, p. 401-426.

[24]O'NEILL, Dan. Gross domestic product. En *Degrowth*. Routledge, 2014. p. 131-136.

[25]BARRO, Robert J., et al. Inflation and growth. *Review-Federal Reserve Bank of Saint Louis*, 1996, vol. 78, p. 153-169.

[26]DEVEREUX, Michael B.; ENGEL, Charles. Exchange rate pass-through, exchange rate volatility, and exchange rate disconnect. *Journal of Monetary economics*, 2002, vol. 49, no 5, p. 913-940.

[27]WHITE, Alan G. *Economic and financial indexes*. 1999. Tesis Doctoral. University of British Columbia.

[28]NEUKIRCH, Thomas. Alternative indexing with the MSCI World Index. *Available at SSRN 1106109*, 2008.

[29]ROKACH, Lior; MAIMON, Oded. Clustering methods. En *Data mining and knowledge discovery handbook*. Springer, Boston, MA, 2005. p. 321-352.

[30]WOLD, Svante; ESBENSEN, Kim; GELADI, Paul. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 1987, vol. 2, no 1-3, p. 37-52.

[31]CAMBRONERO, Cristina García; MORENO, Irene Gómez. Algoritmos de aprendizaje: knn & kmeans. *Intelgencia en Redes de Comunicación, Universidad Carlos III de Madrid*, 2006, vol. 23.

Extra bibliography consulted

What is Portfolio Optimization

COMPARATIVE ANALYSIS OF LINEAR PORTFOLIO REBALANCING STRATEGIES :AN APPLICATION TO HEDGE FUNDS

Conditional Value at Risk (CVaR)

Conditional Drawdown at Risk (CDaR)

Maximum Loss (ML) and Mean-Absolute Deviation risk (MAD)

CDaR and CVaR python approaches

How to compute CVaR

PyPortfolioOpt Documentation

Pyomo Documentation

Levenshtein Explanation (Short Video)

Streamlit Web

Streamlit Selectbox Button

Streamlit Number Input

Streamlit Radio Buttons