Return, Diversification and Risk in Cryptocurrency Portfolios using Deep Recurrent Neural Networks and Multi-Objective Evolutionary Algorithms

Ismael Estalayo[*], Javier Del Ser[*,†,‖], Eneko Osaba[†], Miren Nekane Bilbao[†],
Khan Muhammad[§], Akemi Gálvez[¶] and Andrés Iglesias[¶]
[†] University of the Basque Country (UPV/EHU), 48013 Bilbao, Bizkaia, Spain
[†] TECNALIA, 48160 Derio, Bizkaia, Spain
[§] Sejong University, Seoul 143-747, South Korea
[¶] Universidad de Cantabria, 39005 Santander, Cantabria, Spain
[‖] Contact email: javier.delser@tecnalia.com

***Abstract*—Nowadays the widespread adoption of cryptocurrencies (also referred to as Altcoins) has universalized the access of the society to trading opportunities in alternative markets, thereby laying a rich substrate for the development of new applications and services aimed at easing the management of personal investment portfolios. When selecting how much to invest and in which asset it is often the case that multiple criteria conflict with each other within a single decision making process, which calls for efficient means to optimally balance such contradicting objectives. In this paper we report initial findings around the combination of Deep Learning (DL) models and Multi-Objective Evolutionary Algorithms (MOEAs) for allocating cryptocurrency portfolios. Technical rationale and details are given on the design of a stacked DL recurrent neural network, and how its predictive power can be exploited for yielding accurate *ex ante* estimates of the return and risk of the portfolio. These two objectives are complemented by a measure of the diversity of the investment. Results are presented and discussed with real cryptocurrency data, showcasing the potential of our technical approach to produce near-optimal portfolios by balancing the aforementioned objectives. Our study stimulates further research towards incorporating other relevant decision factors in the design of predictive portfolios, such as the confidence of the output of the DL model.**

I. Introduction

Since the advent of Bitcoin in 2008, the last decade has witnessed a growing proliferation of new digital cryptocurrencies. As a consequence of the universal, unregulated access to these trading assets, an unprecedented outbreak of applications and services has allowed individuals to invest in these new markets in a personalized, agile and autonomous fashion [1]. Indeed, this democratization of trading has been a catalyst for the adoption of new technologies to optimize investment portfolios; data models for time series forecasting and multi-criteria optimization techniques are arguably among the most prominent technological breakthroughs in this field.

From a general perspective, portfolio optimization refers to the allocation of investments over a collection of financial assets under optimality criteria set beforehand [2]. Different theories have been developed so far in order to define under which objectives a portfolio can be declared to be optimal, such as a desired balance between the return of the portfolio over a time period and the risk associated to the investment. This example of optimality criterion lies at the core of the so-called Modern Portfolio Theory (MPT [3]). However, other objectives have been proposed ever since in the literature to characterize the investment universe, such as the diversification of the portfolio, risk parity and other metrics alike.

When several portfolio quality metrics collide into a single optimization problem, their eventually conflicting nature makes it more interesting to discover a set of portfolio solutions (a *Pareto front*) meeting differently yet optimally the trade-off between such metrics. From an algorithmic point of view many solvers have been applied to infer Pareto front approximations in multi-objective portfolio optimization problems [4], with Multi-Objective Evolutionary Algorithms (MOEAs) arguably among the most widely resorted approaches. In fact, single- and multi-objective optimizers propelled by Evolutionary Algorithms (EAs) and Swarm Intelligence (SI) methods have been at the core of manifold contributions in the last years, always dealing with traditional investing assets from general trading markets [5]-[12]. Indeed, this literature trend has been analyzed in a number of comprehensive surveys along the years, which have stressed on the suitability of these bio-inspired optimization algorithms to efficiently undertake complex portfolio selection problems [13]-[15]. As a result, many other areas in finance and economics have embraced the use of these heuristic solvers [16]-[18].

On another note, the prevalence of the return of investment as an optimization objective in portfolio management has given rise to the need for accurately predicting the selling price of assets over the considered time period. Although this has been traditionally a research arena for autoregressive statistical models, the community has lately embraced the use of machine learning to undertake this problem [19], [20], [21]. Surprisingly, scarce attention has been paid to Deep Learning (DL), even though recurrent DL neural networks are renowned for their capability to model long-term relationship within data.

This work takes a step further over the state of the art reviewer above by proposing a novel predictive portfolio optimization scheme for cryptocurrency markets based on recurrent DL models and MOEAs. Specifically, the selection of portfolios is driven by their economical return, the risk entailed by the investment and its diversity. Our formulated problem differs from the usual trend to consider the diversity of the portfolio as a constraint imposed to the problem underneath. Our work instead advocates for including diversity as a third objective to optimize by the solver at hand. To validate the design of the proposed scheme we perform extensive simulations over real cryptocurrency data with a benchmark of different predictive models and MOEAs. The obtained results 1) shed light on the superior performance of the developed DL model with respect to other machine learning models, and 2) evince which MOEAs provide better portfolios in terms of their quantified Pareto optimality.

The rest of the paper is structured as follows: Section II first formulates mathematically the multi-objective portfolio optimization problem tackled in this work, whereas Section III and subsections therein describe the proposed scheme, placing emphasis on the design of the DL model and the benchmark of MOEAs considered to address the aforementioned portfolio optimization problem. Sections IV and V respectively describes the experimental setup and discusses the obtained simulation results. Finally, Section VI concludes the paper and sketches future research paths departing from this work.

II. Problem Formulation

We consider N cryptocurrencies $\mathcal{C} = \{c_1, \ldots, c_N\}$, where c_n may refer to Bitcoin, Ethereum or any other coin alike. We consider that such currencies have historical closing purchase and selling prices $\mathcal{P}_n = \{p_n^t\}$ and $\mathcal{S}_n = \{s_n^t\}$, where for the sake of simplicity we consider daily prices, e.g., t is measured in days, with $t \in \mathbb{N}$ and $|\mathcal{P}_n|$ and $|\mathcal{S}_n|$ equal to each other and to a value depending on the lifetime of currency n. In this context, a *predictive* portfolio $\mathbf{W}^{t+\Delta} = \{w_1^{t+\Delta}, \ldots, w_n^{t+\Delta}\}$ can be defined as the vector of asset weights of each cryptocurrency that drives an investment made at time t on the assumption that a sell command of the entire portfolio will be issued at time $t + \Delta$. As usually assumed in portfolio management we enforce that $\sum_{n=1}^{N} w_n^{t+\Delta} = 1$, i.e. we require that the weights sum to one to keep the investment equal to the available budget.

Intuitively, the main issue in portfolio management is to establish proper criteria for the selection of the values of $\{w_n^{t,\Delta}\}_{n=1}^{N}$. The most straightforward goal could be thought of having the highest expected rate of return of the invested quantity at time $t + \Delta$, defined as:

$$\widehat{R}(\mathbf{W}^{t+\Delta}) = \sum_{n=1}^{N} w_n \left(\frac{\hat{s}_n^{t+\Delta} - p_n^t}{p_n^t} \right), \tag{1}$$

where $\hat{s}_n^{t+\Delta}$ denotes the *predicted* selling price for cryptocurrency n. This prediction can be produced for every cryptocurrency in the portfolio by means of a predictive model $M_{\boldsymbol{\theta}}^n$ learned from their historical selling price records, as done recently in [22], [23] with different machine learning algorithms. Here, $\boldsymbol{\theta}$ stands for the hyper-parameters of the model.

However, volatility of the cryptocurrencies and their historical correlation through time should be also accounted when designing an optimal portfolio. As assumed in the renowned Modern Portfolio Theory by Markowitz [3], investors prefer to avoid taking any risk when investing their funds, so that risk and expected return describe a clear Pareto relationship that should be balanced optimally. In this regard, many metrics have been proposed in the literature for the quantification of the portfolio risk. In this work we embrace the use of the Sharpe ratio [24], which measures the average return earned per unit of volatility or total risk. When framed within the aforementioned notation, the Sharpe ratio is given by:

$$\widehat{\Gamma}(\mathbf{W}^{t+\Delta}) = \frac{\widehat{R}(\mathbf{W}^{t+\Delta})}{f_\sigma(\mathbf{W}^{t+\delta}, \{\mathcal{P}_n\}_{n=1}^{N}, \{\mathcal{S}_n\}_{n=1}^{N})}, \tag{2}$$

where $f_\sigma(\mathbf{W}, \{\mathcal{P}\}_{n=1}^{N})$ is an estimation of the volatility of the expected return of the portfolio. This estimation can be provided by several means, but arguably the most widely adopted one is the variance of the rate of return of the portfolio $\mathbf{W}$ computed over the historical record of purchase and selling prices of its currencies. Specifically:

$$f_\sigma(\mathbf{W}^{t+\delta}, \{\mathcal{P}_n\}_{n=1}^{N}, \{\mathcal{S}_n\}_{n=1}^{N}) = \left(\mathbf{W}^\top \Sigma \mathbf{W}\right)^{1/2}, \tag{3}$$

with $\mathbf{W}^{t+\delta}$ arranged as a $[N \times 1]$ vector, and Σ denoting the $N \times N$ covariance matrix of the random variables representing the historical evolution of the rates of return $(s_n^{t+\Delta} - p_n^t)/p_n^t$ of cryptocurrencies $\{c_1, \ldots, c_N\}$ along time.

Several strategies can be devised in order to keep the risk of the portfolio at its minimum. Specifically, portfolio diversification can be thought of as the implementation of the popular proverb *'eggs should be always carried in different nests'*: this strategy seeks portfolios that achieve a certain level of variability, thus avoiding too much concentration on a reduced number of subsets. Such a variability is usually formulated as a set of constraints included in the definition of the underlying optimization problem, e.g., $\omega_{low} \leq w_n \leq \omega_{high}$, with ω_{low} and ω_{high} denoting thresholds tailored for the investment scenario at hand. However, this variability can be instead quantified by a function and regarded as a third objective for the portfolio design. The entropy defined by Shannon in the context of Information Theory can help us in this regard [25]:

$$H(\mathbf{W}^{t+\Delta}) = -\sum_{n=1}^{N} w_n^{t+\Delta} \log_N w_n^{t+\Delta}, \tag{4}$$

which equals 1 if $w_n^{t+\Delta} = 1/N \ \forall n$ (i.e. uniformly distributed investment) and 0 if $w_{n'}^{t+\Delta} = 1$ and $w_n^{t+\Delta} = 0$ for any $n \neq n'$. This being said, it should be made clear that maximizing a measure of portfolio diversity does not necessarily aligns with the strategy needed to maintain a low risk, as diversity does not account for the joint volatility of assets in the portfolio. Furthermore, decoupling them in two isolated objectives makes practical sense in cryptocurrency markets for the better understandability of diversification by a non-specialized, domestic cryptocurrency investor.

This being stated, the optimization problem addressed in this work focuses on obtaining a set of different portfolios $\{\mathbf{W}^{t+\Delta}\}$ that optimally balance the Pareto relationship between the rate of return, risk and diversity level. Specifically:

$$\begin{aligned} \max_{\mathbf{W}^{t+\Delta}\in\mathbb{R}^N} \quad & \left[\widehat{R}(\mathbf{W}^{t+\Delta}),\, \widehat{\Gamma}(\mathbf{W}^{t+\Delta}),\, H(\mathbf{W}^{t+\Delta})\right], \\ \text{subject to} \quad & \sum_{n=1}^{N} w_n^{t+\Delta} = 1, \end{aligned} \tag{5}$$

which formally poses the predictive multi-objective portfolio optimization problem that will be tackled by using DL models and MOEAs. We next describe such an approach.

III. Proposed Approach

As has been anticipated in the introduction, our designed scheme resorts to DL models for the prediction of the selling price $\hat{s}_n^{t+\Delta}$ required to compute the rate of return and risk of the portfolio as per (1) and (3). Following the schematic diagram in Figure 1, a model $M_{\boldsymbol{\theta}}^n$ per currency in the portfolio is trained and updated every day. Predictions are fed to the portfolio optimization module, which uses them to compute the objective values of portfolios produced by a multi-objective solver during its search process. Finally, the user is offered a set of different portfolios differently trading among return, risk and diversification, based on which he/she makes an investment decision. We now delve into the details of every compounding block of our proposed scheme.

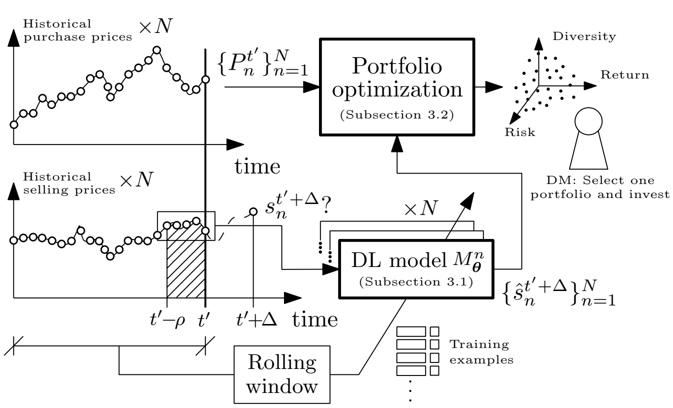

Fig. 1. Diagram of the multi-criteria predictive portfolio optimization scheme

A. Deep Recurrent Networks for Selling Prices Prediction

It has not been until recently when DL models have been used for predicting the value of cryptocurrencies [19], [21], [20] and other general stocks [26], yet none of these recent contributions have still gone beyond predictive modeling towards portfolio management. This is indeed one of the main novel aspects of the scheme proposed in this paper, which utilizes a stacked dual-layer GRU (Gated Recurrent Unit [27]) neural network architecture. This DL model allows capturing longer temporal relationships within input observations.

Specifically, the input of the first layer in our stacked GRU architecture is the historical selling price record $\mathcal{P}_n$ of the considered asset, which is processed through a rolling window to yield examples $(\mathbf{x}_n^t, y_n^t) = ([s_n^{t-\rho}, s_n^{t-\rho+1}, \ldots, s_n^t], s_n^{t+\Delta})$, with ρ denoting the number of past values fed to the network. In connection with Figure 2, the first GRU layer operates on the input examples as:

$$\mathbf{z}^t = \Psi\left(\mathbf{U}^z\mathbf{x}_n^t + \mathbf{V}^z\mathbf{h}_n^{t-1} + \mathbf{b}^z\right), \tag{6}$$

$$\mathbf{r}^t = \Psi\left(\mathbf{U}^r\mathbf{x}_n^t + \mathbf{V}^r\mathbf{h}_n^{t-1} + \mathbf{b}^r\right), \tag{7}$$

$$\mathbf{c}^t = \tanh\left(\mathbf{U}^h\mathbf{x}_n^t + \mathbf{V}^h(\mathbf{h}^{t-1} \otimes \mathbf{r}^t)\right), \tag{8}$$

$$\mathbf{h}_t = (1-\mathbf{z}^t) \otimes \mathbf{h}^{t-1} + \mathbf{z}^t \otimes \mathbf{c}^t, \tag{9}$$

where $\otimes$ denotes Hadamard product, $\Psi()$ is a non-linear activation function (e.g. sigmoid), $Q = |\mathbf{h}^t|$ is the number of features output by the network (i.e. the length of the GRU state vector $\mathbf{h}^t$), and $(\mathbf{U}^z, \mathbf{V}^z, \mathbf{U}^r, \mathbf{V}^r, \mathbf{U}^h, \mathbf{V}^h)$ and $(\mathbf{b}^z, \mathbf{b}^r)$ are weight matrices and bias vectors learned during the training process, respectively. The second layer is set to be identical to the first one, the difference being that it processes the output $\mathbf{h}^t$ of the first layer after a ReLU activation layer in between.

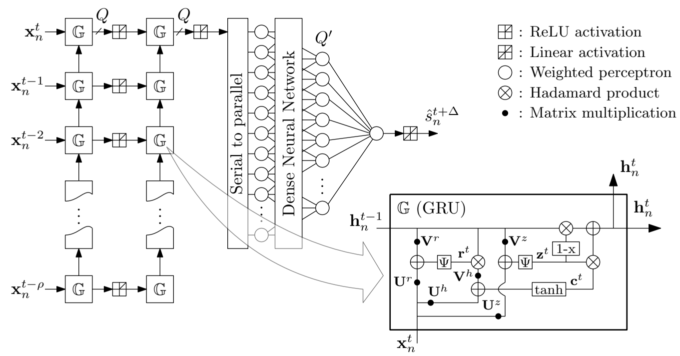

Fig. 2. Proposed stacked GRU architecture for cryptocurrency selling price prediction. For the sake of visual understanding weight vectors $\mathbf{b}^z$ and $\mathbf{b}^r$ have not been depicted in the internals of the exemplified GRU cell.

The model proceeds by flattening the output of the second GRU layer and processing the serialized vector of deep features through a fully connected neural layer with Q' neurons. Finally, the output of this third neuron layer is connected to a single linearly activated perceptron, whose output, after inversing the normalization, will render the sought predicted value $\hat{s}_n^{t+\Delta}$ for any t. This predicted value will be fed to the portfolio optimization module for computing return and risk of every produced portfolio. To efficiently explore the space of possible portfolios we will opt for assessing the performance of different MOEAs, as detailed in the next subsection.

B. Multi-Objective Evolutionary Algorithms

The solver at the core of the portfolio optimization module must undertake a search over $\mathbb{R}^N$ so as to infer the set of Pareto-optimal portfolios balancing the three objectives formulated in Expressions (1) (return), (3) (risk) and (4) (diversification). To this end, MOEAs have proven to undertake this search efficiently in many application domains [28]. Among them, portfolio optimization for traditional stock markets have been also prospected with this family of search algorithms. A comprehensive overview is provided in [29], [13], and more recently in [15]. However, to the best of our knowledge no results have been reported so far with cryptocurrency markets, nor such studies have considered DL models for predicting the evolution of portfolio assets.

This being said, we herein consider a benchmark of 6 different multi-objective solvers:

- Non-dominated Genetic Algorithm II (NSGA-II [30]), which retains potentially good solutions along the search process based on their front rank and crowding distance.
- Non-dominated Genetic Algorithm III (NSGA-III [31]), which substitutes the crowding distance criterion in NSGA-II with a clustering operator aided by a set of distributed reference points.
- Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO [32]), which falls rather on the family of swarm heuristics. SMPSO utilizes a velocity constraint procedure to avoid particles (solutions) to go beyond the limits of the search space. SMPSO also resorts to an external archive of finite size to store non-dominated solutions improvised during the search, from which the leader of the swarm is also selected.
- Multi-objective Evolutionary Algorithm based on Decomposition (MOEAD [33]), which decomposes the multi-objective problem in a number of single-objective subproblems, all solved simultaneously by considering neighborhood information of the produced solutions.
- Generalized Differential Evolution (particularly its GDE3 variant [34]), which extends the original DE/rand/1/bin method to problems with multiple objectives.
- Steady-state Epsilon-dominance Multi-objective Evolutionary Algorithm (ϵ-MOEA [35]), which maintains a subset of hardly dominated solutions and resorts to steady-state replacement rather than the usual Pareto ranking.

When deployed at the core of the portfolio optimization module in Figure 1, each of the algorithms described above provide a set of possible portfolios for the investor to select one of them depending on his/her preferences in terms of return, risk and diversification. It is important at this point to remark that this specific application opens up interesting extensions towards incorporating multi-objective solvers sensitive to decision making, in which the search is guided by a preference interactively instructed by the user along the run [36]. We will elaborate on this identified research path in Section VI.

IV. Experimental Setup

In order to assess the performance of our proposed predictive portfolio optimization scheme, two set of experiments were devised towards fulfilling two different albeit related goals: 1) to verify that the selected DL model (which we hereafter refer to as GRUx2) performs better than other predictive models from the state of the art; and 2) to discriminate which MOEAs among the 6 considered options described in Subsection III-B yield best results in terms of Pareto dominance and spread. In thse setup real data corresponding to $N = 7$ different cryptocurrencies were utilized: Bitcoin (BTC), Ethereum (ETH), Litecoin (LTC), Ripple (XRP), Dash (DSH), Starcoin (STR) and Monero (XMR). Historical daily purchase and selling prices were retrieved from the release date of every cryptocurrency to December 30th, 2018.

Regarding the first objective, for every currency we divide its data in 10 equal-sized parts (without shuffling), using the first 9 for training and the last one as a test holdout for score reporting. Architectural parameters of the proposed GRUx2 model are set to $Q = 256$ (size of the state vector of every GRU unit), $Q' = 30$ (number of neurons of the fully connected neural layer) and sigmoid activation $\Psi(\cdot)$ in Expressions (6) and (7). The model is trained for 20 epochs using a batch size of 256 examples, RMSProp [37] with learning rate equal to 0.001 and Mean Square Error (MSE) as the loss function to minimize by backpropagation through time. Examples $(\mathbf{x}_n^t, y_n^t)$ for all models in this first benchmark are composed by the past $\rho = 20$ values of the selling price at time t (features in $\mathbf{x}_n^t$), and the value of this price in $t+5$ as the target $\mathbf{y}_n^t$ (i.e. a prediction horizon of $\Delta = 5$ days). The comparison analysis will be done on the basis of two well-known regression scores:

$$R^2_{n,m} = 1 - \frac{\sum_{t=1}^{T_n}(s_n^{t+\Delta} - \hat{s}_{n,m}^{t+\Delta})^2}{\sum_{t=1}^{T_n}(s_n^{t+\Delta} - \bar{s}_n)^2} \quad \begin{bmatrix}\text{coefficient of}\\ \text{determination}\end{bmatrix}, \quad (10)$$

$$\text{SMAPE}_{n,m} = \frac{100}{T_n}\sum_{t=1}^{T_n}\frac{2|s_n^{t+\Delta} - \hat{s}_{n,m}^{t+\Delta}|}{|s_n^{t+\Delta}| + |\hat{s}_{n,m}^{t+\Delta}|} \quad \begin{bmatrix}\text{symmetric mean}\\ \text{absolute percen-}\\ \text{tage error}\end{bmatrix}, \quad (11)$$

where T_n is the length of the test subset of cryptocurrency n, $\hat{s}_{n,m}^{t+\Delta}$ the estimate of $s_n^{t+\Delta}$ produced by model m at time t, and $\bar{s}_n = 1/T_n \sum_{t=1}^{T_n} s_n^{t+\Delta}$, i.e. the average of test samples.

In order to undertake the second objective pursued in our experimental study, we perform 20 independent runs of each of the considered MOEAs on a fixed test date t' common to all test sets of the cryptocurrencies. For the sake of fairness all of them are configured with the same maximum number of allowed fitness evaluations ($20,000$). Inner parameters controlling the search behavior of every algorithm were tuned off-line towards finding their best performing configuration, thereby yielding the results discussed in this section. Solvers in this second benchmark will be compared to each other in terms of two widely utilized indicators for Pareto quality: hypervolume and Inverted Generational Distance (IGD) in its IGD+ variant [38]. For the latter we use the aggregated Pareto front approximation (computed over all algorithms and runs) as the reference front.

V. Results and Discussion

We begin our analysis with the results obtained to shed light on the first objective of the experiments. Such results are summarized in Table I, where values of $R^2_{n,m}$ and $\text{SMAPE}_{n,m}$ measured over the test set of every cryptocurrency are listed for the proposed GRUx2 model and other regression algorithms: three ensemble models, namely, a Random Forest Regressor (RFR), an AdaBoost regressor (ABR) and a

Gradient Boosting Regression (GBR), all composed by 100 fully-grown trees; and a ϵ-Support Vector Regression (ϵ-SVR) with Gaussian kernel. Hyperparameters of these models were optimized by 10-fold cross-validation over the training set looking for the model configuration with maximum average $R^2_{n,m}$ score. As can be observed from this table, GRUx2 outperforms the rest of models in the benchmark, specially for XRP and DSH cryptocurrencies.

TABLE I
$R^2_{n,m}$ (TOP) AND $\text{SMAPE}_{n,m}$ (BOTTOM) OBTAINED OVER THE TEST SET FOR EVERY LEARNING MODEL AND CRYPTOCURRENCY

$R^2_{n,m}$ $\text{SMAPE}_{n,m}$		Models $m \in \{1,\ldots,5\}$				
		GRUx2 (proposed)	RFR	ABR	GBR	SVR
Cryptocurrencies $n \in \{1,\ldots,7\}$	BTC	0.841 7.677%	0.802 8.277%	0.751 8.488%	0.820 7.680%	0.833 7.597%
	ETH	0.793 11.24%	0.716 12.31%	0.756 11.74%	0.737 12.50%	0.782 12.17%
	LTC	0.833 10.17%	0.759 12.12%	0.702 13.35%	0.778 11.35%	0.829 10.20%
	XRP	0.845 6.646%	0.615 12.85%	0.521 15.96%	0.776 9.932%	0.743 9.449%
	DSH	0.873 12.32%	0.778 14.25%	0.688 18.13%	0.802 13.42%	0.835 13.09%
	STR	0.705 12.02%	0.622 13.48%	0.627 13.92%	0.565 14.20%	0.703 12.13%
	XMR	0.875 10.98%	0.823 12.24%	0.308 22.97%	0.786 13.43%	0.852 11.28%

After verifying that the designed model excels at predicting selling prices over the test set, we proceed by inspecting the relative performance of the MOEAs to find a set of Pareto-optimal investment portfolios. This second phase of our experimental study departs from Figure 3, where we illustrate the statistics of the multi-objective quality indicators computed over the 20 executed runs of every algorithm in the form of boxplots. It can be noted that two multi-objective solvers perform better than the rest in the benchmark (SMPSO and MOEAD) in terms of hypervolume, yielding statistically significant gaps as confirmed by a non-parametric Wilcoxon rank sum test with a confidence level of 0.95. As for the results of the IGD+ indicator, differences are reduced as evinced by the overlapping distributions of the IGD+ values corresponding to GDE3, NSGAII, SMPSO and MOEAD. Nevertheless, Wilcoxon rank sum tests between the IGD+ results of SMPSO and those of the rest of the multi-objective supported the statistical significance of the gaps attained by this solver. NSGAIII and ϵ-MOEA remain notably worse than the rest of their counterparts in the benchmark.

We end up our study by assessing the Pareto front approximations produced by the two winning MOEAs of the above benchmark (MOEAD and SMPSO). To this end, we isolate the output front of the portfolio optimization module for one single run of both solvers. As depicted in Figure 4, the shape of the Pareto front produced by MOEAD seems to be more spread than that of SMPSO, specially for the volume slice corresponding to mid values of the entropy objective (which we recall reflects the diversity of the investment). On the contrary, SMPSO renders a more enriched group of portfolios in the region of low and high entropy values. We have checked that this interesting effect, with practical consequences on which we discuss later in this section, also holds for other runs of the algorithms.

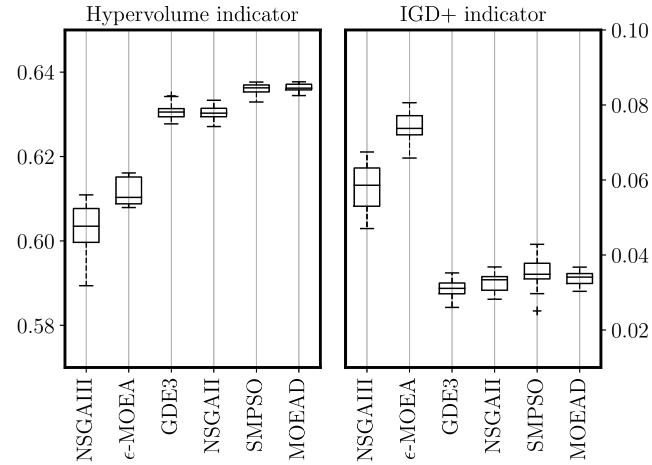

Fig. 3. Boxplot of the values of the hypervolume (left) and IGD+ (right) indicators achieved by every MOEA over 20 independent runs.

From a practical standpoint, the generally higher density of portfolios provided by SMPSO can be helpful for tracing investing strategies of finer granularity. This is specially appealing when dealing with cryptocurrencies, for which investments are executed in a more dynamic fashion than in traditional trading markets.

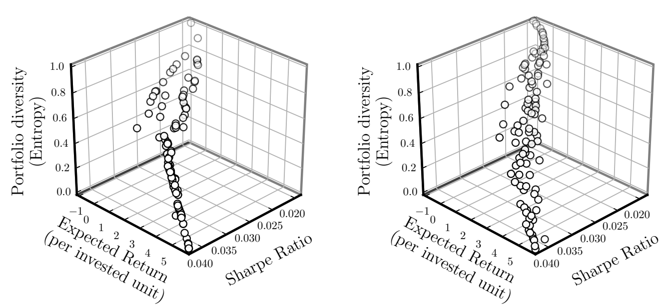

Fig. 4. Example of the Pareto front estimated by MOEAD (left) and SMPSO (right) in one isolated run.

VI. CONCLUSIONS AND FUTURE RESEARCH LINES

This manuscript has elaborated on the interplay between return, risk and diversity in portfolio optimization. Specifically, our work builds upon the renowned Pareto trade-off between the earnings provided by a portfolio and the risk assumed when performing the investment, giving rise to a Pareto relationship that lies at the core of Markowitz's theory of modern portfolio management. We have augmented this twofold criterion by including a third objective that quantifies explicitly and exclusively the diversity of the investment, which we advocate to serve as an intuitive metric for the average user of the system. Furthermore, our proposed framework leverages the use of multi-layered deep recurrent neural network regression models to provide more accurate predictive estimates of the selling price of every asset in the portfolio. Results obtained for a set of experiments carried out with real cryptocurrency data have verified the superior performance of our designed deep learning model with respect to other regression techniques. A second experimental phase has elaborated on the selection of the solver to balance among the aforementioned objectives, for which a benchmark of 6 different multi-objective evolutionary algorithms has been considered. MOEAD and SMPSO have been found to perform better than their counterparts, yet providing Pareto fronts with different characteristics in terms of distance (spacing) among the produced portfolios.

Different research paths root on the results reported in this paper. To begin with, the investing flexibility of this particular trading market allows managing portfolios whose invested amount in each of their assets can be sold at different instants in time. Bearing this observation in mind, efforts will be invested towards incorporating this functionality in the proposed scheme so as to intelligently decide not only *how much* to invest in every cryptocurrency, but also *when* to recover the investment. Another extension deserving further investigation in the future is to explore the confidence of the prediction provided by the deep learning model, so that the quantified measure of model uncertainty is also accounted in the risk assumed by the investor. Finally, the more recent flavor of decision-making MOEAs is another promising line to prospect for portfolio management, particularly for very short-term predictive trading (i.e. small values of Δ) in which automated and agile interaction between the investor and the portfolio management scheme must be ensured.

Acknowledgments

The work of Javier Del Ser and Eneko Osaba is partially supported by the EMAITEK program of the Basque Government, Spain. Akemi Gálvez and Andrés Iglesias acknowledge funding support from TIN2017-89275-R (AEI/FEDER, EU) and PDE-GIR (H2020, MSCA program, ref. 778035).

References

[1] A. H. Dyhrberg, "Hedging capabilities of bitcoin. is it the virtual gold?" *Finance Research Letters*, vol. 16, pp. 139–144, 2016.
[2] P. Jorion, "Portfolio optimization in practice," *Financial Analysts Journal*, vol. 48, no. 1, pp. 68–74, 1992.
[3] H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.
[4] A. M. Rather, V. Sastry, and A. Agarwal, "Stock market prediction and portfolio selection models: a survey," *OPSEARCH*, vol. 54, no. 3, pp. 558–579, 2017.
[5] F. Streichert, H. Ulmer, and A. Zell, "Evolutionary algorithms and the cardinality constrained portfolio optimization problem," in *Operations Research Proceedings*, 2004, pp. 253–260.
[6] J. Branke, B. Scheckenbach, M. Stein, K. Deb, and H. Schmeck, "Portfolio optimization with an envelope-based multi-objective evolutionary algorithm," *European Journal of Operational Research*, vol. 199, no. 3, pp. 684–693, 2009.
[7] K. P. Anagnostopoulos and G. Mamanis, "A portfolio optimization model with three objectives and discrete variables," *Computers & Operations Research*, vol. 37, no. 7, pp. 1285–1297, 2010.
[8] L. Diosan, "A multi-objective evolutionary approach to the portfolio optimization problem," in *International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, vol. 2, 2005, pp. 183–187.
[9] K. P. Anagnostopoulos and G. Mamanis, "Multiobjective evolutionary algorithms for complex portfolio optimization problems," *Computational Management Science*, vol. 8, no. 3, pp. 259–279, 2011.
[10] H. Zhu, Y. Wang, K. Wang, and Y. Chen, "Particle swarm optimization (PSO) for the constrained portfolio optimization problem," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10 161–10 169, 2011.
[11] D. Kumar and K. Mishra, "Portfolio optimization using novel covariance guided artificial bee colony algorithm," *Swarm and Evolutionary Computation*, vol. 33, pp. 119–130, 2017.
[12] D. Quintana, R. Denysiuk, S. Garcia-Rodriguez, and A. Gaspar-Cunha, "Portfolio implementation risk management using evolutionary multiobjective optimization," *Applied Sciences*, vol. 7, no. 10, p. 1079, 2017.
[13] A. Ponsich, A. L. Jaimes, and C. A. C. Coello, "A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 3, pp. 321–344, 2013.
[14] J. Doering, A. A. Juan, R. Kizys, A. Fito, and L. Calvet, "Solving realistic portfolio optimization problems via metaheuristics: A survey and an example," in *Modeling and Simulation in Engineering, Economics and Management*, 2016, pp. 22–30.
[15] O. Ertenlice and C. B. Kalayci, "A survey of swarm intelligence for portfolio optimization: Algorithms and applications," *Swarm and Evolutionary Computation*, vol. 39, pp. 36–52, 2018.
[16] M. G. C. Tapia and C. A. C. Coello, "Applications of multi-objective evolutionary algorithms in economics and finance: A survey," in *IEEE Congress on Evolutionary Computation*, 2007, pp. 532–539.
[17] R. Aguilar-Rivera, M. Valenzuela-Rendón, and J. Rodríguez-Ortiz, "Genetic algorithms and darwinian approaches in financial applications: A survey," *Expert Systems with Applications*, vol. 42, no. 21, pp. 7684–7697, 2015.
[18] A. Soler-Dominguez, A. A. Juan, and R. Kizys, "A survey on financial applications of metaheuristics," *ACM Computing Surveys*, vol. 50, no. 1, p. 15, 2017.
[19] S. McNally, J. Roche, and S. Caton, "Predicting the price of bitcoin using machine learning," in *Euromicro International Conference on Parallel, Distributed and Network-based Processing*, 2018, pp. 339–343.
[20] T. Phaladisailoed and T. Numnonda, "Machine learning models comparison for bitcoin price prediction," in *International Conference on Information Technology and Electrical Engineering*, 2018, pp. 506–511.
[21] L. Alessandretti, A. ElBahrawy, L. M. Aiello, and A. Baronchelli, "Anticipating cryptocurrency prices using machine learning," *Complexity*, vol. 2018, 2018.
[22] H. Jang and J. Lee, "An empirical study on modeling and prediction of bitcoin prices with bayesian neural networks based on blockchain information," *IEEE Access*, vol. 6, pp. 5427–5437, 2018.
[23] S. Karasu, A. Altan, Z. Saraç, and R. Hacioğlu, "Prediction of bitcoin prices with machine learning methods using time series data," in *Signal Processing and Communications Applications*, 2018, pp. 1–4.
[24] W. F. Sharpe, "Mutual fund performance," *The Journal of Business*, vol. 39, no. 1, pp. 119–138, 1966.
[25] C. E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
[26] X. Pang, Y. Zhou, P. Wang, W. Lin, and V. Chang, "An innovative neural network approach for stock market prediction," *The Journal of Supercomputing*, pp. 1–21, 2018.
[27] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[28] C. A. C. Coello and G. B. Lamont, *Applications of multi-objective evolutionary algorithms*. World Scientific, 2004, vol. 1.
[29] K. Metaxiotis and K. Liagkouras, "Multiobjective evolutionary algorithms for portfolio management: A comprehensive literature review," *Expert Systems with Applications*, vol. 39, no. 14, pp. 11 685–11 698, 2012.
[30] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
[31] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[32] A. J. Nebro, J. J. Durillo, J. Garcia-Nieto, C. C. Coello, F. Luna, and E. Alba, "SMPSO: A new PSO-based metaheuristic for multi-objective optimization," in *IEEE Symposium on Computational Intelligence in Multi-criteria Decision-Making*, 2009, pp. 66–73.

[33] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[34] S. Kukkonen and J. Lampinen, "GDE3: The third evolution step of generalized differential evolution," in *IEEE Congress on Evolutionary Computation*, vol. 1, 2005, pp. 443–450.

[35] K. Deb, M. Mohan, and S. Mishra, "Towards a quick computation of well-spread pareto-optimal solutions," in *International Conference on Evolutionary Multi-Criterion Optimization*, 2003, pp. 222–236.

[36] A. J. Nebro, A. B. Ruiz, C. Barba-González, J. García-Nieto, M. Luque, and J. F. Aldana-Montes, "InDM2: Interactive dynamic multi-objective decision making using evolutionary algorithms," *Swarm and Evolutionary Computation*, vol. 40, pp. 184–195, 2018.

[37] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[38] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Modified distance calculation in generational distance and inverted generational distance," in *International Conference on Evolutionary Multi-Criterion Optimization*, 2015, pp. 110–125.