# Feature weighting for data analysis via evolutionary simulation

Aris Daniilidis[1], Alberto Domínguez Corella[1,2], and Philipp Wissgott[3]

[1] Institut für Stochastik und Wirtschaftsmathematik, Variational Analysis, Dynamics and Operations Research Unit E105-04, Technische Universität Wien, Wiedner Hauptstraße 8, 1040 Vienna, Austria
aris.daniilidis@tuwien.ac.at
alberto.of.sonora@gmail.com

[2] Institut für Mathematik und Wissenschaftliches Rechnen, Universität Graz, Heinrichstraße 36, A-8010 Graz, Austria

[3] danube.ai solutions gmbh, 1040 Vienna, Austria
philipp@danube.ai

**Abstract.** We analyze an algorithm for assigning weights prior to scalarization in discrete multi-objective problems arising from data analysis. The algorithm evolves the weights (the relevance of features) by a replicator-type dynamic on the standard simplex, with update indices computed from a normalized data matrix. We prove that the resulting sequence converges globally to a unique interior equilibrium, yielding non-degenerate limiting weights. The method, originally inspired by evolutionary game theory, differs from standard weighting schemes in that it is analytically tractable with provable convergence.

**Keywords:** feature weighting, evolutionary algorithms, multi-objective optimization, data analysis.

## 1 Introduction, data problem, and theoretical results

In many applied domains—such as economics, energy systems, technological design, and consumer recommendation—the relative importance of features describing alternatives is often unknown and must be inferred from data or performance criteria.

We analyze an algorithm recently introduced in [22, Section III] for estimating the relative importance of dataset features. The feature weights are modelled as probability vectors and are iteratively updated from an initial distribution according to a replicator-type dynamic. The method is inspired by evolutionary game theory: features are interpreted as players whose weights evolve according to their relative performance, whereas the update rules depend on strategy functions determined by the data type. Here, we restrict attention to two simple strategies: in the first strategy features with strong variation should increase in weight; in the second one, datasets penalize over-reliance on a single feature. Following mathematical folklore and the so-called *No Free Lunch* theorem—here interpreted as a metatheorem stating that no optimization algorithm is universally superior or able to solve all problems—we remark that the relevance of the algorithm lies in its behavior on structured datasets, where inductive biases reflecting feature interactions can be effectively exploited.

We next describe the mathematical problem of assigning weights to dataset features and introduce the proposed algorithm. Related literature is then reviewed. For readability, the proofs are deferred to Section 2. An illustrative numerical example is presented in Section 3, while analogies with genetic algorithms and evolutionary systems are discussed in Section 4.

### 1.1 The data problem

Consider a finite set $\mathcal{X} = \{x_1, \dots, x_n\} \subset \mathbb{R}^m$ representing the available choices of a rational agent. The set can also represent different datasets. For each $j \in \{1, \dots, m\}$, the number $x_{ij} \in \mathbb{R}$ represents a feature of option/dataset $i \in \{1, \dots, n\}$. The agent seeks to choose an option that maximizes some features and minimizes others, however this is not always possible and one seeks for a combined concept of *optimality* in this context. This problem falls within the class of so-called multi-objective problems. While this framework offers several notions of solution—such as Pareto optima—these do not capture the relative importance of each feature. In this paper, we analyze the possible relevance (quantified as an index) of features that can be obtained from the data of the problem.

Below, we consider an algorithm that outputs a probability vector $\gamma \in \mathbb{R}^m$, where each $\gamma_j$ represents an index of the relevance in optimizing feature $j \in \{1, \dots, m\}$. The algorithm is inspired by evolutionary game theory, drawing an analogy in which datasets correspond to *organisms* and features to *genes*; this interpretation is described in detail in Section 4.

The input of the algorithm can be thought of as a matrix

$$X := \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix}.$$

To effectively process data and provide a consistent interpretation across features, the raw data is transformed into a normalized representation. The purposes of this normalization are twofold. First, we want to ensure that all feature values lie in a consistent scale, and second, we want to provide data with a percentage-based interpretation, thereby requiring $\Phi_{ij} \in [0, 1]$ for all $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$. Once the columnwise normalizations are fixed for a problem class, any dataset with the same structure can be processed dynamically without further manual tuning.

Consider the normalization represented by a matrix

$$\Phi := \begin{bmatrix} \Phi_{11} & \Phi_{12} & \dots & \Phi_{1m} \\ \Phi_{21} & \Phi_{22} & \dots & \Phi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{n1} & \Phi_{n2} & \dots & \Phi_{nm} \end{bmatrix}$$

with entries in $[0, 1]$. Such normalization transforms optimization of features to a multi-objective problem. For example, when dealing with matrices X containing non-negative data, for a column $j \in \{1, \dots, m\}$ not identically zero, a possible transformation could be given by

$$\Phi_{ij} \;=\; 1 \;-\; \frac{x_{ij}}{\max\limits_{s \in \{1,\dots,n\}} x_{sj}}, \qquad \text{for all } \; i \in \{1, \dots, n\}.$$

This transformation applies an *inverted normalization* to the column. Each entry is scaled by its feature's maximum, and this ratio is subtracted from 1. Consequently, $\Phi_{ij} \in [0, 1]$, where 0 denotes the feature's maximum value (no deviation) and 1 signifies an entry of 0 (100% deviation). This yields a measure that is useful when the objective is to minimize a feature rather than maximize it.

### 1.2 Algorithm and convergence results

Let us begin with some notation. We denote by

$$\mathcal{K}^{m-1} := \left\{ \gamma \in \mathbb{R}^m \middle| \gamma_j \geq 0 \;\; \forall j \in \{1, \dots, m\} \;\text{ and }\; \sum_{j=1}^{m} \gamma_j = 1 \right\}$$

the standard simplex in $\mathbb{R}^m$. The elements of $\mathcal{K}^{m-1}$ correspond to the scalarization vectors (feature weights) of the multi-objective problem. We write

$$\widetilde{\Phi_j} := \frac{1}{n}\sum_{i=1}^{n} \Phi_{ij}$$

for the column average of the matrix $\Phi \in M_{n\times m}(\mathbb{R})$. Notice that each average lies in the interval $[0,1]$.

For each $j \in \{1,\dots,m\}$, consider the functions $\Delta_j^{\mathrm{dom}}, \Delta_j^{\mathrm{bal}} : \mathcal{K}^{m-1} \to \mathbb{R}$ given by

$$\Delta_j^{\mathrm{dom}}(\gamma) := \gamma_j\Big(\widetilde{\Phi_j} - \frac{1}{2}\Big) \quad \text{and} \quad \Delta_j^{\mathrm{bal}}(\gamma) := -2\Big(\gamma_j\widetilde{\Phi_j} - \frac{1}{m}\sum_{s=1}^{m}\gamma_s\widetilde{\Phi_s}\Big).$$

The quantities $\Delta_j^{\mathrm{dom}}$ (dominance term) and $\Delta_j^{\mathrm{bal}}$ (balanced term) can be viewed as indices that evaluate the contribution of feature $j \in \{1,\dots,m\}$ under different criteria. The dominance term measures whether a feature tends to be high or low on average. It is positive when the column average is above 0.5, and its effect grows with the weight given to that feature. In this sense, it rewards features that are both strongly weighted and above the midpoint of the scale. The balance term compares the weighted contribution of a feature with the overall weighted mean across all features. It is positive when the feature contributes less than the global mean, and negative when it contributes more. This acts as a correction, it discourages features from dominating too strongly and favors those that are closer to balance with the rest.

We also consider the function $\Delta_j : \mathcal{K}^{m-1} \to \mathbb{R}$ given by

$$\Delta_j(\gamma) := \Delta_j^{\mathrm{dom}}(\gamma) + \Delta_j^{\mathrm{bal}}(\gamma) = -\gamma_j\Big(\widetilde{\Phi_j} + \tfrac{1}{2}\Big) \; + \; \frac{2}{m}\sum_{s=1}^{m}\gamma_s\widetilde{\Phi_s}\,. \tag{1}$$

This is an index of the relative contribution of feature $j \in \{1,\dots,m\}$, combining the effects of the dominance and balance terms.

**Algorithm 1**

**Require:** Initial weight $\gamma^0 \in \mathcal{K}^{m-1}$.
1: Initialize $\gamma \longleftarrow \gamma^0$
2: **for** $k \in \mathbb{N}$ **do**
3: Compute $\Delta \longleftarrow \Delta(\gamma)$
4: **for** $j \leftarrow 1$ to $m$ **do**
5: Update $\gamma_j \longleftarrow \gamma_j(1+\Delta_j)$
6: **end for**
7: Normalize $\gamma \longleftarrow \gamma/\sum_{s=1}^{m}\gamma_s$
8: **end for**

Given an initial weight $\gamma^0 \in \mathcal{K}^{m-1}$, we say that $(\gamma^k)_{k\in\mathbb{N}} \subseteq \mathcal{K}^{m-1}$ is a sequence generated by Algorithm 1 if, for each $k \in \mathbb{N}$,

$$\Delta_j(\gamma^k) + 1 > 0 \qquad \text{and} \qquad \gamma_j^{k+1} = \frac{\gamma_j^k\left(1+\Delta_j(\gamma^k)\right)}{\displaystyle\sum_{s=1}^{m}\gamma_s^k\left(1+\Delta_s(\gamma^k)\right)}, \quad \forall j \in \{1,\dots,m\}.$$

We see that the update rule defines a discrete dynamical system on the simplex. At each step, the weight of feature $j$ is multiplied by a factor proportional to $1 + \Delta_j(\gamma^k)$, and the whole vector is renormalized so that the updated weights again lie in $\mathcal{K}^{m-1}$. In this way, features with positive indices $\Delta_j(\gamma^k)$ are reinforced, while those with negative indices are diminished. This mechanism is reminiscent of the spirit of replicator dynamics, where the evolution of a distribution is driven by relative performance, with better-performing features gaining weight at the expense of weaker ones.

**Proposition 1 (Non-degeneracy of limit points of Algorithm 1).** *Let the relative interior of the standard simplex be denoted by*

$$\operatorname{relint} \mathcal{K}^{m-1} := \left\{ \gamma \in \mathbb{R}^m \middle| \gamma_j > 0 \ \ \forall j \in \{1, \dots, m\} \ \text{ and } \ \sum_{j=1}^{m} \gamma_j = 1 \right\}$$

*Let $(\gamma^k)_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1. If $\gamma^0 \in \mathcal{K}^{m-1}$ belongs to* $\operatorname{relint} \mathcal{K}^{m-1}$*, then any accumulation point of $(\gamma^k)_{k\in\mathbb{N}}$ also belongs to* $\operatorname{relint} \mathcal{K}^{m-1}$*.*

The previous proposition shows that the replicator-type updates preserve positivity of the weights in the limit; once the process starts in the relative interior of the simplex, it never assigns zero weight to any feature in the limit. Geometrically, the dynamics avoid collapsing onto the boundary of the simplex. This ensures that no feature is ultimately discarded.

With this invariance established, we now turn to the asymptotic behavior of the sequence. The key question is whether the updates converge to a stable distribution inside the simplex, rather than oscillating or drifting indefinitely.

**Theorem 1 (Convergence of Algorithm 1 to an equilibrium).** *Let $(\gamma^k)_{k\in\mathbb{N}}$ be a sequence generated by Algorithm 1 with $\gamma^0 \in \operatorname{relint}\left(\mathcal{K}^{m-1}\right)$. Then, there exists a unique $\gamma^* \in \operatorname{relint}\left(\mathcal{K}^{m-1}\right)$ such that $\gamma^k \longrightarrow \gamma^*$ as $k \longrightarrow +\infty$. Moreover,*

$$\gamma_j^* = \left( \sum_{s=1}^{m} \frac{\widetilde{\Phi_j} + \frac{1}{2}}{\widetilde{\Phi_s} + \frac{1}{2}} \right)^{-1} \qquad \forall j \in \{1, \dots, m\}. \tag{2}$$

Formula (2) reveals that the limiting weight of each feature depends monotonically on its column average; it decreases as the average increases. In particular, features with larger averages receive less weight, while those with smaller averages receive more.

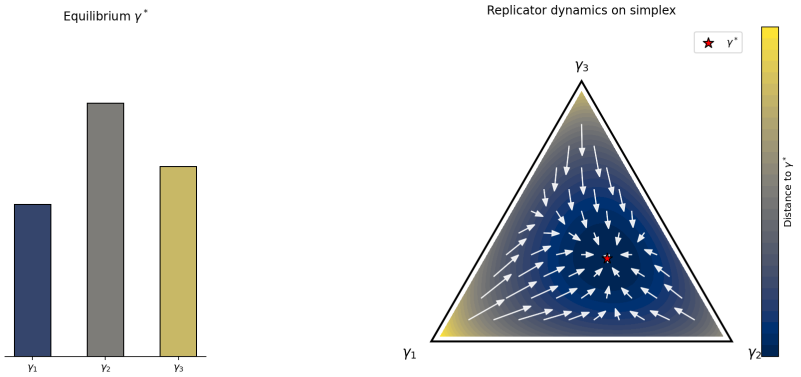


Fig. 1: Illustration of the replicator-type feature weighting algorithm.

Since the limiting weights generated by the algorithm remain strictly positive, they can be used to scalarize the multi-objective problem of maximizing or minimizing all features, thereby allowing recovery of Pareto–optimal points from the dataset.

**Corollary 1 (Pareto optimality of the equilibrium weights).** *Assume that the normalization is coordinatewise order–preserving, i.e., for each* $j \in \{1, \dots, m\}$ *and all* $i, k \in \{1, \dots, n\}$

$$x_{ij} \leq x_{kj} \iff \Phi_{ij} \leq \Phi_{kj}.$$

*Let* $\gamma^* \in \operatorname{relint}(\mathcal{K}^{m-1})$ *be the solution obtained by Theorem 1 and let* $\mathcal{X} = \{x_1, \dots, x_n\} \subset \mathbb{R}^m$ *be the original dataset. Let* $F : \mathcal{X} \to \mathbb{R}^m$ *be the multi-objective function given by*

$$F(x_i) := (x_{i1}, \dots, x_{im}).$$

*Then, any index* $i^* \in \{1, \dots, n\}$ *maximizing the scalarization*

$$r_i := \sum_{j=1}^{m} \gamma_j^* \, \Phi_{ij}$$

*yields a Pareto-optimal element* $x_{i^*} \in \mathcal{X}$ *for the problem of maximizing* $F$*.*

The same conclusion holds if the objective is to minimize all features instead of maximize them, with the adequate changes mutatis mutandis.

### 1.3 Related literature

Our algorithm determines (learns) a probability vector of feature weights by a multiplicative (replicator–type) update on the simplex and admits a closed-form interior limit. This places the algorithm between evolutionary multi-objective optimization, which typically evolves solutions, and feature-weighting methods in machine learning, which estimate weights but rarely via a dynamical system.

In evolutionary multi-objective optimization (EMO), many algorithms differ in how they approximate Pareto fronts. In [24] a comparison of methods is given, noting common mechanisms such as Pareto-dominance ranking, external archives, performance indicators, and decomposition into scalar subproblems. Algorithms such as the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [7] and the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [25] rely on Pareto-based strategies without incorporating weight vectors. In contrast, the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [23] explicitly uses weight vectors to decompose a multi-objective problem into scalar subproblems. These weight vectors determine search directions and, to a large extent, the distribution of the final solution set [17]. In its original form, MOEA/D uses fixed weight vectors, but subsequent research has introduced adaptive strategies. For example, pa$\lambda$-MOEA/D (Pareto-adaptive MOEA/D) [14] adjusts weight vectors using geometric features of the estimated Pareto front; DMOEA/D (Diversity-maintained MOEA/D) [12] employs projection/equidistant interpolation of reference (weight) vectors from the current nondominated set to preserve diversity; and MOEA/D with Adaptive Weight Adjustment (MOEA/D-AWA) [20] periodically removes weight vectors in crowded regions and inserts new ones in sparse regions during the search. Other notable contributions include the Reference Vector Guided Evolutionary Algorithm (RVEA) [3] and its later variants. Our approach diverges from the ones previously mentioned. Instead of adapting weight vectors to shape search directions, we evolve feature weights from data, with the interior equilibrium itself providing a scalarizer.

In machine learning, classical filter methods assign fixed importances to features. For instance, Relief [15] uses nearest-neighbor contrasts, and ReliefF [16] extends this to multi-class and noisy data. Another popular method, mRMR (short for Minimum Redundancy – Maximum Relevance),

selects features that are both highly informative about the target and minimally redundant with each other, typically using mutual information or F-statistics for relevance and pairwise redundancy measures [19]. In unsupervised learning, feature-weighted $k$-means methods span decades of variants [8]. Evolutionary strategies are also present, e.g., estimation-of-distribution methods applied to feature weighting [13], and comparisons between genetic and co-evolutionary schemes [2]. These methods generally optimize weights empirically but do not define a replicator-style dynamical system nor provide closed-form equilibria as in our approach.

## 2 Proofs

### 2.1 Proof of Proposition 1

We divide the proof in two steps. In the first one, we show that the sequence $\{\gamma^k\}_{k\in\mathbb{N}}$ remains in the relative interior of $\mathcal{K}^{m-1}$. In the second one, we demonstrate that any accumulation point must also lie in this relative interior.

**Step 1.** (*The sequence remains in the relative interior of the simplex*). We proceed by induction. For $k=0$, $\gamma^0 \in \operatorname{relint}(\mathcal{K}^{m-1})$ by assumption. Assume $\gamma^k \in \operatorname{relint}(\mathcal{K}^{m-1})$ for some $k \geq 0$. From Algorithm 1,

$$1+\Delta_j(\gamma^k) > 0 \quad \text{and} \quad \gamma_j^{k+1} = \frac{\gamma_j^k\left(1+\Delta_j(\gamma^k)\right)}{\sum_{s=1}^{m}\gamma_s^k\left(1+\Delta_s(\gamma^k)\right)} \quad \forall j \in \{1,\dots,m\}.$$

Since $\gamma_j^k > 0$ and $1+\Delta_j(\gamma^k) > 0$ for all $j \in \{1,\dots,m\}$, we obtain that $\gamma_j^{k+1} > 0$ for all $j \in \{1,\dots,m\}$. Normalization then ensures $\sum_{j=1}^m \gamma_j^{k+1} = 1$. Thus $\gamma^{k+1} \in \operatorname{relint}(\mathcal{K}^{m-1})$.

**Step 2.** (*The accumulation points remain in the relative interior of the simplex*). Suppose that there exists an accumulation point $\gamma^* \in \mathcal{K}^{m-1}$ that does not lie in the relative interior. Therefore, we have $\gamma_j^* = 0$ for some $j \in \{1,\dots,m\}$ and there exists a subsequence $(\gamma^{k_\ell})_{\ell\in\mathbb{N}}$ of $(\gamma^k)_{k\in\mathbb{N}}$ converging to $\gamma^*$ with $\gamma_j^{k_\ell} \longrightarrow 0$ as $\ell \longrightarrow +\infty$. We will now derive a contradiction.

Let $\ell_0 \in \mathbb{N}$ be such that $\gamma_j^{k_\ell} < (2m)^{-1}(\widetilde{\Phi_j}+1/2)^{-1}$ for all $\ell \geq \ell_0$. From the Cauchy-Schwarz inequality we have $\sum_{s=1}^m \gamma_s^2 \geq 1/m$, for any $\gamma \in \mathcal{K}^{m-1}$. Then, since $\widetilde{\Phi_s} \geq 0$, an easy calculation yields that for all $\ell \geq \ell_0$ we have:

$$\begin{aligned} 1+\Delta_j(\gamma^{k_\ell}) - \sum_{s=1}^m \gamma_s^{k_\ell}\Delta_s(\gamma^{k_\ell}) &= 1+\sum_{s=1}^m \left(\gamma_s^{k_\ell}\right)^2\left(\widetilde{\Phi_s}+\frac{1}{2}\right) - \gamma_j^{k_\ell}\left(\widetilde{\Phi_j}+\frac{1}{2}\right) \\ &\geq 1+\frac{1}{2m} - \gamma_j^{k_\ell}\left(\widetilde{\Phi_j}+\frac{1}{2}\right) > 1\,. \end{aligned}$$

From this, we deduce that, for $\ell \geq \ell_0$,

$$\gamma_j^{k_\ell+1} = \frac{\gamma_j^{k_\ell}\left(1+\Delta_j(\gamma^{k_\ell})\right)}{\sum_{s=1}^{m}\gamma_s^{k_\ell}\left(1+\Delta_s(\gamma^{k_\ell})\right)} > \gamma_j^{k_\ell}.$$

This contradicts that $\gamma_j^{k_\ell} \longrightarrow 0$ as $\ell \longrightarrow +\infty$. □

### 2.2 Proof of Theorem 1

We will prove that any subsequence of $(\gamma^k)_{k\in\mathbb{N}}$ must converge to the same limit, and therefore the sequence itself converges to that limit. Let $(\gamma^{k_\ell})_{\ell\in\mathbb{N}}$ be a subsequence of $(\gamma^k)_{k\in\mathbb{N}}$ converging to some $\gamma^* \in \mathcal{K}^{m-1}$. From the definition of Algorithm 1, the limit should satisfy

$$\gamma_j^* \Big(1 + \sum_{s=1}^{m} \gamma_s^* \Delta_s(\gamma^*)\Big) = \gamma_j^* \big(1 + \Delta_j(\gamma^*)\big), \quad \forall j \in \{1,\dots,m\}.$$

Since, by Proposition 1, $\gamma^*$ belongs to the relative interior of $\mathcal{K}^{m-1}$, the previous expression simplifies to

$$\sum_{s=1}^{m} \gamma_s^* \Delta_s(\gamma^*) = \Delta_j(\gamma^*), \qquad \forall j \in \{1,\dots,m\}.$$

Set $c := \displaystyle\sum_{s=1}^{m} \gamma_s^* \Delta_s(\gamma^*)$. Recalling (1) we have:

$$\Delta_j(\gamma^*) = -\gamma_j^* \Big(\widetilde{\Phi_j} + \tfrac{1}{2}\Big) \; + \; \frac{2}{m} \sum_{s=1}^{m} \gamma_s^* \widetilde{\Phi_s}\,, \qquad \forall j \in \{1,\dots,m\}.$$

Let us denote, for $j \in \{1,\dots,m\}$, $A_j := \widetilde{\Phi_j} + \frac{1}{2}$ and

$$B := \frac{2}{m} \sum_{s=1}^{m} \gamma_s^* \widetilde{\Phi_s}.$$

Then, $-A_j\,\gamma_j^* + B = c$ for all $j \in \{1,\dots,m\}$, so

$$\gamma_j^* = \frac{B-c}{A_j} \quad \forall j \in \{1,\dots,m\}.$$

Since $\displaystyle\sum_{s=1}^{m} \gamma_s^* = 1$, we obtain

$$1 = \sum_{s=1}^{m} \frac{B-c}{A_s} = \big(B-c\big) \sum_{s=1}^{m} \frac{1}{A_s}.$$

Therefore

$$B - c = \Big(\sum_{s=1}^{m} 1/A_s\Big)^{-1}$$

and consequently

$$\gamma_j^* = \frac{1}{A_j \displaystyle\sum_{s=1}^{m} 1/A_s}, \qquad \forall j \in \{1,\dots,m\}.$$

The result follows. □

### 2.3 Proof of Corollary 1

Define, for each $i \in \{1, \dots, n\}$,

$$r_i := \sum_{j=1}^{m} \gamma_j^* \Phi_{ij},$$

and choose $i^* \in \arg\max_{i \in \{1,\dots,n\}} r_i$. Suppose $x_{i^*}$ is not Pareto-optimal for maximizing our multi-objective function $F$. Then there exists $k \in \{1, \dots, n\}$ such that

$$x_{kj} \geq x_{i^* j} \quad \text{for all } j \in \{1, \dots, m\} \quad \text{and } x_{kj_0} > x_{i^* j_0} \text{ for some } j_0 \in \{1, \dots, m\}.$$

By the coordinatewise order preservation of the normalization, we have

$$\Phi_{kj} \geq \Phi_{i^* j} \quad \text{for all } j \in \{1, \dots, m\}, \quad \text{and } \Phi_{kj_0} > \Phi_{i^* j_0}.$$

Since $\gamma^* \in \operatorname{relint}(\mathcal{K}^{m-1})$, we have $\gamma_j^* > 0$ for all $j \in \{1, \dots, m\}$, hence

$$r_k - r_{i^*} = \sum_{j=1}^{m} \gamma_j^* (\Phi_{kj} - \Phi_{i^* j}) > 0,$$

which contradicts the maximality of $r_{i^*}$. Therefore $x_{i^*}$ is Pareto-optimal for maximizing $F$. □

## 3 An illustrative numerical experiment

We consider a real-world dataset of 15 office listings for rent in Vienna, collected from the public real estate platform `immoscout24.at`. Each listing is described by four numerical features: monthly rent (in euros), office size (in square meters), number of rooms, and if they have a balcony (which is here modeled as a binary variable with values 0 and 1). These features are heterogeneous in scale and meaning, so direct comparison is not meaningful. To address this, we normalize the data into a matrix $\Phi \in [0, 1]^{15 \times 4}$ such that higher values consistently represent more desirable attributes. For the rent feature, where lower values are preferred, we apply a shifted inverted normalization:

$$\Phi_{i1} = 1 - \frac{x_{i1} - \min_s x_{s1}}{\max_s x_{s1}}. \tag{3}$$

For the features size and rooms (columns 2 and 3), where higher values are preferred, we apply a normalization relative to their maximum values:

$$\Phi_{ij} = \frac{x_{ij}}{\max_s x_{sj}} \quad \text{for } j = 2, 3. \tag{4}$$

This choice of normalization functions aims to map the real-world dataset to a subset of $[0, 1]$ in a comparable way. In particular, equation (3) and (4) both map to $[\min_s x_{sj} / \max_s x_{sj}, 1]$. For the last column (balcony), which is already binary and appropriately scaled, no transformation is applied. This results in a matrix $\Phi$ with entries in $[0, 1]$, where higher values consistently correspond to more

desirable properties.

$$\Phi = \begin{bmatrix} 0.6160 & 0.3000 & 0.2069 & 0.0000 \\ 0.8175 & 0.2891 & 0.2759 & 0.0000 \\ 0.2529 & 1.0000 & 0.4828 & 0.0000 \\ 0.4624 & 0.7152 & 0.4138 & 0.0000 \\ 0.4228 & 0.5478 & 0.4138 & 1.0000 \\ 0.5368 & 0.4761 & 0.4138 & 0.0000 \\ 1.0000 & 0.2674 & 0.1379 & 0.0000 \\ 0.6087 & 0.3804 & 0.3448 & 0.0000 \\ 0.2180 & 0.5000 & 0.5517 & 0.0000 \\ 0.5218 & 0.3457 & 0.2759 & 0.0000 \\ 0.9356 & 0.2891 & 0.2069 & 0.0000 \\ 0.1913 & 0.8326 & 1.0000 & 1.0000 \\ 0.1938 & 0.6826 & 0.4828 & 0.0000 \\ 0.1310 & 0.7283 & 0.4828 & 0.0000 \\ 0.7498 & 0.3587 & 0.2069 & 0.0000 \end{bmatrix}$$

The original (unnormalized) data is shown in Table 2. Using $\Phi$, we apply the evolutionary algorithm described in Subsection 1.2, iterating for $N = 10$ steps. The resulting feature weight vector $\gamma^k$ converges rapidly to the analytical limit $\gamma^*$ from Theorem 1, as shown in Table 3.

Table 1: Column-wise means $\widetilde{\Phi}_j$ of the normalized matrix $\Phi$.

| Feature | Rent | Size | Rooms | Balcony |
|---|---|---|---|---|
| Mean $\widetilde{\Phi}_j$ | 0.5106 | 0.5142 | 0.3931 | 0.1333 |

The limit point in Theorem 1 is given by

$$\gamma_1^* = 0.2117, \quad \gamma_2^* = 0.2109, \quad \gamma_3^* = 0.2395, \quad \gamma_4^* = 0.3378.$$

Approximately 33% of the total weight is assigned to the feature 'balcony'. The reason for this behavior can be understood as 'balcony' has by far the lowest average, since only two offices have a balcony. Hence, as discussed in Section 4.5, the fixed point formula from (2) gives the highest relevance to features with the lowest average. In statistically distributed data, objectives with a low average tend to have rare and large statistical outliers. The presented evolutionary approach distributes more weight to these objectives, since it is assumed that rare traits give an organism an advantage in the evolution. Indeed, in our presented example, having a balcony is exceptionally rare, making offices with one highly desirable.

Table 2: Apartment dataset including balcony.

| Name as advertised | Rent (€) | Size ($m^2$) | Rooms | Balcony |
|---|---|---|---|---|
| Charmantes Altbaubüro in zentraler Lage Nähe Kärntner Straße und Oper zu mieten | 4348 | 138 | 3.0 | 0 |
| Exklusives Büro in der Börse! | 2647 | 133 | 4.0 | 0 |

Continued on next page

| Name as advertised | Rent (€) | Size ($m^2$) | Rooms | Balcony |
|---|---|---|---|---|
| Wunderschöne geräumige Bürofläche im energieeffizienten Gebäude | 7413 | 460 | 7.0 | 0 |
| Attraktive Bürofläche in ruhiger Grünlage | 5644 | 329 | 6.0 | 0 |
| Barrierefreie DG Bürofläche in der Favoritenstraße direkt bei der U1 | 5979 | 252 | 6.0 | 1 |
| Modernes Büro in zentraler Lage | 5016 | 219 | 6.0 | 0 |
| Bürofläche am Höchstädtplatz: Sehr gute Infrastruktur im Haus und Umgebung — U6 | 1106 | 123 | 2.0 | 0 |
| Repräsentatives Altbaubüro im Palais Schlick zu mieten | 4409 | 175 | 5.0 | 0 |
| Gekühlte Bürofläche am Bauernmarkt — unbefristet | 7708 | 230 | 8.0 | 0 |
| Gewerbefläche mit Straßenzugang in generalsaniertem Jugendstilhaus | 5143 | 159 | 4.0 | 0 |
| Modernes Büro/Praxis in Wien: Erstbezug, 132$m^2$, U-Bahn-Nähe, Top-Ausstattung! | 1650 | 133 | 3.0 | 0 |
| Repräsentatives Büro direkt am Karlsplatz in der Belle Etage | 7933 | 383 | 14.5 | 1 |
| Repräsentative Bürofläche oder Praxis beim Schottentor U2/ Servitenviertel | 7912 | 314 | 7.0 | 0 |
| Büro direkt auf der Dresdnerstraße im BC 20 - Bauteil B zu mieten | 8442 | 335 | 7.0 | 0 |
| Bürofläche im Bürokomplex Nähe Matzleinsdorfer Platz zu mieten | 3218 | 165 | 3.0 | 0 |

To quantify the deviation of the computed weights $\gamma^*$ from the uniform baseline, we define the *impact norm* as

$$\|\Phi\| := \frac{m}{m-1} \sum_{j=1}^{m} \left| \gamma_j^* - \frac{1}{m} \right| \cdot \widetilde{\Phi}_j,$$

where $\widetilde{\Phi}_j := \frac{1}{n} \sum_{i=1}^{n} \Phi_{ij}$ is the average of column $j$. This measures the unevenness of weights, adjusted by the average strength of each feature. The factor $m/(m-1)$ normalizes the index so that its maximum is 1, attained when a single feature with average 1 receives all the weight.

To highlight the influence of the top-performing agents, we define the *qualified impact norm*

$$\|\Phi\|_* := \frac{m}{m-1} \sum_{j=1}^{m} \left| \gamma_j^* - \frac{1}{m} \right| \cdot \left( \frac{1}{|X^*|} \sum_{x_i \in X^*} \Phi_{ij} \right),$$

where $X^* \subset X$ denotes the 10% of agents with the highest row-wise averages $\widetilde{\phi}_i = \frac{1}{m} \sum_{j=1}^{m} \Phi_{ij}$. Note that this index is not a norm in the usual sense. This version emphasizes how the best listings are affected by feature weights. In this experiment, we compute

$$\|\Phi\| = 0.0739 \quad \text{and} \quad \|\Phi\|_* = 0.1785.$$

This indicates a mild deviation from uniform weighting, with more pronounced feature bias among the best-performing listings.

We evaluate the contribution of each feature using the *feature impact*

$$\zeta_j := \left(\max_i \Phi_{ij} - \min_i \Phi_{ij}\right) \gamma_j^*.$$

The values for this example are given by

$$\zeta_1 = 0.18397, \quad \zeta_2 = 0.15454, \quad \zeta_3 = 0.20651, \quad \zeta_4 = 0.33780.$$

As for the weights $\gamma_j^*$, also for the feature impact, the feature 'balcony' dominates the example. This can be understood since an office cannot have half a balcony, but only 0 or 1. Hence, while other objectives show a gradual change within a certain range, 'balcony' has one single step-wise change, 'yes' or 'no'. Figure 2 illustrates the convergence of the weights $\gamma^k$ over the course of 10 iterations.

Table 3: Evolution of feature weights $\gamma^k$ over 10 iterations and comparison to the analytical solution $\gamma^*$.

| Iteration | Rent | Size | Rooms | Balcony |
|---|---|---|---|---|
| $\gamma^0$ | 0.2500 | 0.2500 | 0.2500 | 0.2500 |
| $\gamma^1$ | 0.2421 | 0.2419 | 0.2497 | 0.2664 |
| $\gamma^2$ | 0.2358 | 0.2354 | 0.2487 | 0.2802 |
| $\gamma^3$ | 0.2307 | 0.2302 | 0.2475 | 0.2917 |
| $\gamma^4$ | 0.2266 | 0.2261 | 0.2462 | 0.3011 |
| $\gamma^5$ | 0.2234 | 0.2228 | 0.2450 | 0.3088 |
| $\gamma^6$ | 0.2209 | 0.2202 | 0.2440 | 0.3149 |
| $\gamma^7$ | 0.2189 | 0.2182 | 0.2431 | 0.3198 |
| $\gamma^8$ | 0.2173 | 0.2166 | 0.2424 | 0.3237 |
| $\gamma^9$ | 0.2161 | 0.2154 | 0.2418 | 0.3267 |
| $\gamma^{10}$ | 0.2151 | 0.2144 | 0.2413 | 0.3291 |
| $\gamma^*$ | 0.2117 | 0.2109 | 0.2395 | 0.3378 |

Starting from the uniform initialization $\gamma^0 = (0.25, 0.25, 0.25, 0.25)$, each coordinate gradually drifts toward its fixed point value. The weights for *Rent*, *Size*, and *Rooms* decrease slightly and stabilize near 0.21–0.24, while the weight for *Balcony* steadily increases, converging to approximately 0.34. The dashed lines mark the analytical solution $\gamma^*$, showing that the iterative dynamics approach the fixed point with monotone convergence in each coordinate. For each office $x_i$, we define an aggregate score $r_i = \sum_{j=1}^m \gamma_j \Phi_{ij}$, which evaluates the overall desirability of $x_i$ given a weight vector $\gamma$. With uniform weights, $\gamma^{\text{uniform}} = (0.25, 0.25, 0.25, 0.25)$, all features contribute equally and produce the baseline scores $r_i^{\text{init}}$. With $\gamma = \gamma^*$, the fixed point solution, the scores $r_i^{\text{evol}}$ incorporate the learned feature importances. Ranking the offices by $r_i$ under these two regimes gives the comparison shown in Table 4.

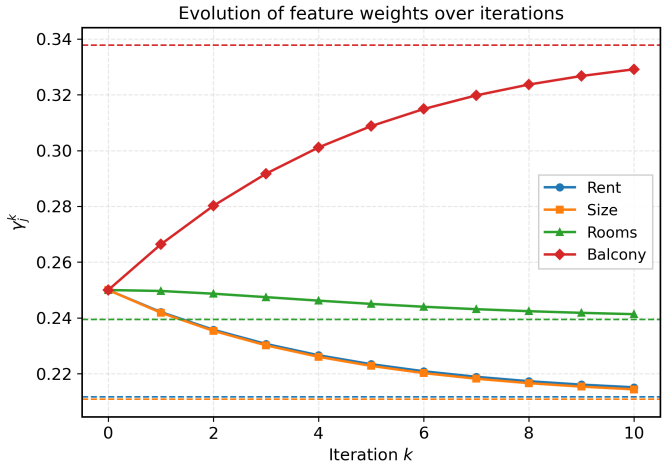


Fig. 2: Evolution of feature weights $\gamma_j^k$ over $k = 0, \ldots, 10$. Dashed lines indicate the analytical solution $\gamma^*$.

Table 4: Office rankings under uniform weights $\gamma^{\text{uniform}} = (0.25, 0.25, 0.25, 0.25)$ and under $\gamma^* = (0.2117, 0.2109, 0.2395, 0.3378)$.

| | $\gamma^{\text{uniform}}$ | | | | | $\gamma^*$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Rank | Score | Rent | Size | Rooms | Balc. | Score | Rent | Size | Rooms | Balc. |
| 1 | 0.755979 | 7933 | 383 | 14.5 | 1 | 0.793484 | 7933 | 383 | 14.5 | 1 |
| 2 | 0.596097 | 5979 | 252 | 6 | 1 | 0.641988 | 5979 | 252 | 6 | 1 |
| 3 | 0.433915 | 7413 | 460 | 7 | 0 | 0.380130 | 7413 | 460 | 7 | 0 |
| 4 | 0.397865 | 5644 | 329 | 6 | 0 | 0.347898 | 5644 | 329 | 6 | 0 |
| 5 | 0.357897 | 1650 | 133 | 3 | 0 | 0.313203 | 5016 | 219 | 6 | 0 |
| 6 | 0.356680 | 5016 | 219 | 6 | 0 | 0.308616 | 1650 | 133 | 3 | 0 |
| 7 | 0.351331 | 1106 | 123 | 2 | 0 | 0.301152 | 1106 | 123 | 2 | 0 |
| 8 | 0.345613 | 2647 | 133 | 4 | 0 | 0.300664 | 7912 | 314 | 7 | 0 |
| 9 | 0.339790 | 7912 | 314 | 7 | 0 | 0.300134 | 2647 | 133 | 4 | 0 |
| 10 | 0.335508 | 8442 | 335 | 7 | 0 | 0.297003 | 8442 | 335 | 7 | 0 |
| 11 | 0.333501 | 4409 | 175 | 5 | 0 | 0.291728 | 4409 | 175 | 5 | 0 |
| 12 | 0.328854 | 3218 | 165 | 3 | 0 | 0.283968 | 3218 | 165 | 3 | 0 |
| 13 | 0.317420 | 7708 | 230 | 8 | 0 | 0.283780 | 7708 | 230 | 8 | 0 |
| 14 | 0.285828 | 5143 | 159 | 4 | 0 | 0.249463 | 5143 | 159 | 4 | 0 |
| 15 | 0.280716 | 4348 | 138 | 3 | 0 | 0.243249 | 4348 | 138 | 3 | 0 |

## 4 Analogies with genetic algorithms and evolutionary systems

Given the fixed point $\gamma^*$ from Theorem 1, a natural question is how the update functions $\Delta_j^{\text{dom}}$, $\Delta_j^{\text{bal}}$, and the overall update rule are derived. In what follows, we provide an evolutionary interpretation of the matrix $\Phi$, which leads directly to the definitions in Section 1.2.

We interpret each agent $x_i \in \mathcal{X}$ as an organism in an evolutionary environment. The goal of the optimization is then understood as identifying the fittest organism. A natural definition for the

*global fitness* of organism $i$ is

$$r_i := \sum_{j=1}^{m} \gamma_j \Phi_{ij}, \tag{5}$$

where $\gamma \in \mathcal{K}^{m-1}$ is a weight vector assigning relevance to each feature. We assume that the normalized matrix $\Phi$ is constructed so that higher values of $\Phi_{ij}$ reflect better performance of organism $i$ in feature $j$. As all entries of $\Phi$ lie in $[0,1]$, it follows that $r_i \in [0,1]$ as well. Under this interpretation, the entry $\Phi_{ij}$ represents the *local fitness* of organism $i$ with respect to feature $j$. From an evolutionary perspective, we may view each feature as a gene, and $\gamma_j$ as the fitness or strength of expression of gene $j$. The global fitness $r_i$ thus reflects how well the organism performs, taking into account both the quality of its features and the importance assigned to them by $\gamma$.

In Dawkins [5], one central argument states that genes and not organisms are the 'unit' of competition in a species. That means that the more the phenomenological features of a gene help the overall fitness of an organism, the more successful a gene will be in spreading to a population through replication. In newer editions of [5] and also in the follow-up Dawkins [6], this focus on genes is somehow reduced and the trade-off between gene and organism behavior comes stronger into play.

One of the central paradigms of biological entities in evolution is that they have to efficiently use the available resources. From an organism point of view, this means that there has to be a trade-off between beneficial features for reproduction and their cost in terms of resources or energy consumption. For example, if the energy cost to grow wings is too substantial compared to the gain of collected food, evolution will not favor their development over the course of generations. From a mathematical perspective, this process is nothing more than an optimization of genes and organisms with a fixed overall amount of resources.

In the spirit of the work of Dawkins and in our evolutionary picture of data, let us conduct a 'thought experiment': what if, to analyze the fitness of genes and organisms, we convert $\Phi$ into an evolutionary simulation, where we iteratively test local and global fitness? To that end, we envision a virtual, dimensionless 'heap of resources' and calculate the share of resources a gene obtains. The more shares a gene can accumulate, the more fit it will be in the next iteration, i.e., the larger the fitness of the gene $\gamma_j$.

Continuing with an absolute heap size, would lead to unstable and inconsistent behavior, since the order genes accessing the heap would influence the result. Instead, we continue with a differential share $\Delta_j$ for the $j$th gene and require (i) larger positive shares $\Delta_j$ tend to increase $\gamma_j$ and vise versa, (ii) the total sum of gene fitness values is constant, (iii) the gene fitness does not change too abruptly to allow a fair comparison between genes. Taking these conditions into account, we can formulate a genetic replicator equation:

$$\gamma_j^{k+1} = \frac{\gamma_j^k \left(1 + \Delta_j^k\right)}{\sum_{s=1}^{m} \gamma_s^k \left(1 + \Delta_s^k\right)} \quad \forall j \in \{1, \ldots, m\}. \tag{6}$$

As a next step in our evolutionary representation, we have to define the functions $\Delta_j^k$. As described above, in biology, there is an interplay between the genes and organisms in evolution. On the one hand, evolution in connection with genes can be interpreted as *local* comparison of fitness in terms of a single feature. On the other hand, the evolution of organisms in a species can be understood as a *global* comparison with respect to all genes available in the population. Hence, also in our evolutionary interpretation there are two contributions to $\Delta_j$

$$\Delta_j^k = \Delta_j^{k,\mathrm{gen}} + \Delta_j^{k,\mathrm{org}}$$

where $\Delta_j^{k,\mathrm{gen}}$ denotes the gene contribution and $\Delta_j^{k,\mathrm{org}}$ denotes the organism contribution, respectively.

The functions $\Delta_j^{k,\mathrm{gen}}$ and $\Delta_j^{k,\mathrm{org}}$ govern the dynamic behavior of the system via the replicator equations (6). In the evolutionary picture these functions correspond to strategies of individual genes and organisms. Hence, the function are assembled as

$$\Delta_j^{k,\mathrm{gen}} = \frac{1}{n}\sum_{i=1}^{n} \Delta_{ij}^{k,\mathrm{gen}}, \tag{7}$$

$$\Delta_j^{k,\mathrm{org}} = \frac{1}{n}\sum_{i=1}^{n} \Delta_{ij}^{k,\mathrm{org}}, \tag{8}$$

where $\Delta_{ij}^{k,\mathrm{gen}}$, $\Delta_{ij}^{k,\mathrm{org}}$ denote the contribution to the differential share of a single gene or organism, respectively, and recalling that $n$ denotes the total number of organisms in the system.

In this paper, we will derive a simple variant for the strategy functions $\Delta_{ij}^{k,\mathrm{gen}}$ and $\Delta_{ij}^{k,\mathrm{org}}$, leaving more complex strategies for future work. Generally, as conditions for the choice of strategies, we require that not all gene fitness be shifted to a single gene. In that case a single $\gamma_{j'} = 1$ while all other weights in equation (5) would be zero. In the optimization this would not be desirable, since the information of $\Phi_{ij}$ for $j \neq j'$ would be lost.

### 4.1 Gene contribution $\Delta_{ij}^{k,\mathrm{gen}}$

Consider a gene describing a single feature in a real evolutionary system, e.g., the size of an animal in a species. Further, assuming that larger size is beneficial in the chosen ecosystem, the fitness of the gene 'size' in a population can be estimated by taking into account the sizes of all animals in the population and comparing them to genes describing other features of the animal. Qualitatively speaking, if for example many large animals in a population are to be found, one can deduct that size is genetically more important than other features therein.

In addition to the magnitude of a quantity such as size, the importance of the feature in a previous reproduction cycle also influences the future fitness of a gene in the evolution. Translating this dynamics to our optimization problem, we define 'dom' the dominant gene strategy function

$$\Delta_{ij}^{k,\mathrm{gen}} = \Delta_{ij}^{k,\mathrm{dom}} := \gamma_j^k \left(\Phi_{ij} - \frac{1}{2}\right). \tag{9}$$

Note that since $\Delta_{ij}^{k,\mathrm{dom}}$ is used to compute $\gamma_j^{k+1}$ via equation (6), the linear factor $\gamma_j^k$ in equation (9) is the only sequence-dependence factor that takes the current gene fitness into account for the next iteration. Additionally, the factor $(\Phi_{ij} - \frac{1}{2})$ will yield a positive differential share when $\Phi_{ij} > 1/2$ and vice versa.

Taking only $\Delta_{ij}^{k,\mathrm{dom}}$ for $\Delta_j^k$ into account, it can be easily shown that the objective $j'$ with maximum $\widetilde{\Phi_{j'}}$ will accumulate all the weight, i.e. $\gamma_{j'} = 1$. Hence, for a non-trivial result, we need $\Delta_{ij}^{k,\mathrm{org}}$ to counter the direct dominant effect of the gene strategy function $\Delta_{ij}^{k,\mathrm{dom}}$.

### 4.2 Organisms contribution $\Delta_{ij}^{k,\mathrm{org}}$

In evolution, organisms with only a few excellent features have difficulties to thrive. It is usually the lifeforms flexible enough to quickly adapt to a changing habitat to win the day. Applying this paradigm to our optimization problem $\Phi$, we need a quantity to measure the dependence of an organism on a particular gene

$$\mu_{ij} := \frac{\gamma_j \Phi_{ij}}{\sum_s \gamma_s \Phi_{is}} = \frac{\gamma_j \Phi_{ij}}{r_i} \tag{10}$$

Note that the larger $\mu_{ij}$, the more the fitness of an organism $i$ depends on a particular gene $j$. Furthermore, from an organism perspective, with $m$ features at hand, the ideal share would be $\mu_{ij} = \frac{1}{m}$ for all $j \in 1, \ldots, m$.

Consequently, we obtain as balanced organism strategy function

$$\Delta_{ij}^{k,\mathrm{org}} = \Delta_{ij}^{k,\mathrm{bal}} := -2r_i \left(\mu_{ij} - \frac{1}{m}\right). \tag{11}$$

Note that a perfectly balanced organism, where $\mu_{ij} = \frac{1}{m}$ for all $j \in \{1, \ldots, m\}$, will yield no differential change to any $\Delta_j^{bal}$.

### 4.3 Summary of evolutionary argument

It is important to observe that equation (9) for $\Delta_{ij}^{k,\mathrm{dom}}$ yields a *relative* change to the gene fitness $\gamma_j$. In particular, for a specific organism $i$, a gene $j'$ can only gain fitness, if $\Phi_{ij'}$ is larger than the other $\Phi_{ij}$, $j \neq j'$. Thus, the effect of $\Delta_{ij}^{k,\mathrm{gen}}$ is directly proportional to the rate of asymmetry of an agents data. In contrast, the minus sign in equation (11) for $\Delta_{ij}^{k,\mathrm{bal}}$ indicates that asymmetry in an agent is penalized. The total differential share $\Delta_{ij}^{k} := \Delta_{ij}^{k,\mathrm{dom}} + \Delta_{ij}^{k,\mathrm{bal}}$ is thus an evolutionary trade-off between these two effects related to the asymmetry of a feature $j'$ compared to other objectives in optimization.

Combining all contributions for the dominant strategy, we obtain

$$\Delta_j^{k,\mathrm{dom}} := \frac{1}{n}\sum_{i=1}^{n} \Delta_{ij}^{k,\mathrm{dom}} = \frac{1}{n}\sum_{i=1}^{n} \gamma_j \left(\Phi_{ij} - \frac{1}{2}\right) = \gamma_j \left(\widetilde{\Phi_{ij}} - \frac{1}{2}\right)$$

with the mean value $\widetilde{\Phi_{ij}}$, leading into the corresponding definition of Sec. 1.2. Analogously, we accumulate all contributions for the balanced strategy as

$$\Delta_j^{k,\mathrm{bal}} := \frac{1}{n}\sum_{i=1}^{n} \Delta_{ij}^{k,\mathrm{bal}} = -\frac{2}{n}\sum_{i=1}^{n} r_i \left(\mu_{ij} - \frac{1}{m}\right) = -\frac{2}{n}\sum_{i=1}^{n} \left(\gamma_j \Phi_{ij} - \frac{1}{m}\sum_{s} \gamma_s \Phi_{is}\right)$$

again leading to the corresponding expression in Sec. 1.2.

### 4.4 Asymptotic behavior in higher dimensions

The update terms $\Delta_j^{\mathrm{dom}}$ and $\Delta_j^{\mathrm{bal}}$ can be interpreted as two competing evolutionary pressures: the former rewards dominance of feature $j$ across the population, while the latter penalizes imbalance in how much feature $j$ contributes to the overall fitness of agents. For the algorithm to remain stable and unbiased as the number of features $m$ grows, both terms should scale similarly with respect to $m$. To that end, we examine the asymptotic behavior of the two update components.

As the number of features $m$ increases, and in the absence of strong preference for any particular one, the weights $\gamma_j$ are expected to distribute more evenly across all features. This implies that for large $m$, each $\gamma_j$ is typically of order $\mathcal{O}(1/m)$. Recall that the dominance term is given by

$$\Delta_j^{\mathrm{dom}} = \gamma_j \cdot \left(\widetilde{\Phi_j} - \frac{1}{2}\right),$$

where $\widetilde{\Phi_j}$ is the average value of feature $j$ across all agents. The quantity $\widetilde{\Phi_j} - \frac{1}{2}$ measures how much the average of feature $j$ deviates from a neutral baseline. Since $\widetilde{\Phi_j}$ remains in the compact set $[0, 1]$

as $m$ increases, the product $\gamma_j(\widetilde{\Phi_j} - \frac{1}{2})$ inherits the scaling $\mathcal{O}(1/m)$ from $\gamma_j$. We therefore conclude that

$$\Delta_j^{\mathrm{dom}} = \mathcal{O}\left(\tfrac{1}{m}\right).$$

This scaling reflects that, as the number of features grows, the influence of each individual feature on the evolutionary update diminishes—unless that feature significantly deviates from the average. The balancing term is defined as

$$\Delta_j^{\mathrm{bal}} = -\beta\left(\gamma_j\widetilde{\Phi_j} - \frac{1}{m}\sum_{s=1}^{m}\gamma_s\widetilde{\Phi_s}\right),$$

where the factor $\beta = 2$ is chosen to ensure that this penalty has the same scale as the dominance term. The term inside the parentheses measures how much the contribution of feature $j$ deviates from the average over all features. Since the mean $\sum_{s=1}^{m}\gamma_s\widetilde{\Phi_s}$ is of order $\mathcal{O}(1)$, both terms in the difference are $\mathcal{O}(1/m)$, so the result is again

$$\Delta_j^{\mathrm{bal}} = \mathcal{O}\left(\tfrac{1}{m}\right).$$

The choice $\beta = 2$ reflects a symmetry: since $\Delta_j^{\mathrm{dom}}$ is centered around $1/2$, the balancing correction must be scaled accordingly to counteract overdominance and ensure convergence to a stable equilibrium. More generally, if the dominance term were centered around a constant $a$, one would require $\beta = 1/a$ to retain this matching. Note that the introduced dimensional argument is a good analysis tool also for more complex evolutionary strategies. In general, all strategies included in an algorithm should show the same asymptotic behavior in terms of $m$ to allow for a fair comparison.

### 4.5 A minimal example and its evolutionary interpretation

Let us investigate the effect of the evolutionary approach on a minimal $2 \times 2$ example

$$X := \begin{bmatrix} 1 & 0 \\ 0.5 & 0.5 \end{bmatrix}.$$

Since the entries are already in the interval $[0, 1]$, we take $\Phi = X$. Inserting the mean values $\widetilde{\Phi_1} = 0.75$ and $\widetilde{\Phi_2} = 0.25$ into the fixed point formula (2) yields

$$\gamma_1^* = 0.375, \quad \gamma_2^* = 0.625$$

Most of the relevance is assigned to the second feature, which has the lower average. In our evolutionary interpretation, this means that a lower average local fitness for a given gene makes its rare, larger values even more significant.

As before, imagine an animal population where size is a relevant feature described by a specific gene. Furthermore, assume that the majority of animals in the population are small. Then, a single larger organism has a decisive evolutionary advantage, since there are only a few competitive opponents in terms of size when fighting for the available resources.

Let us now investigate variants of the previous example. For each $\xi \in [0, 0.5]$, consider the $2 \times 2$ matrix

$$X(\xi) := \begin{bmatrix} 1 & 0 \\ 0.5 + \xi & 0.5 - \xi \end{bmatrix},$$

As before, take $\Phi(\xi) = X(\xi)$. Note that as $\xi \to 0.5$ the matrix takes its maximally asymmetric form

$$X(0.5) = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}.$$

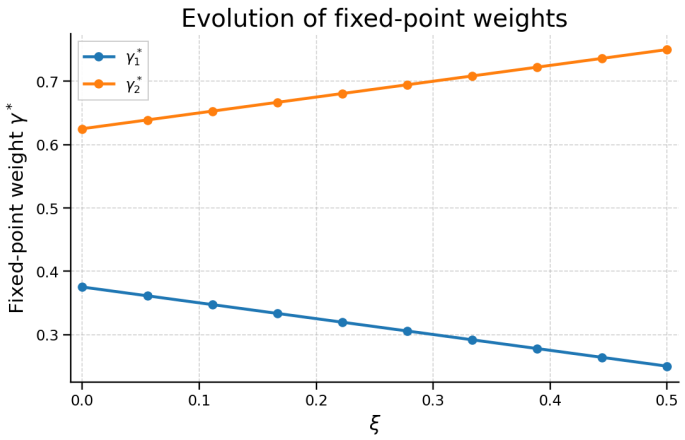


Fig. 3: Evolution of fixed-point weights $\gamma_1^*$ and $\gamma_2^*$ as a function of $\xi$.

Table 5: Evolution of feature weights $\gamma$ for 6 values of $\xi$.

| $\xi$ | $\gamma_1^*$ | $\gamma_2^*$ |
|---|---|---|
| 0 | 0.375 | 0.625 |
| 0.1 | 0.350 | 0.650 |
| 0.2 | 0.325 | 0.675 |
| 0.3 | 0.300 | 0.700 |
| 0.4 | 0.275 | 0.725 |
| 0.5 | 0.250 | 0.750 |

Table 5 shows the values of $\gamma^*$ as a function of $\xi$, showing linear behavior with respect to $\xi$ with slope 0.25.

The final values $\gamma_1^* = 0.25$ and $\gamma_2^* = 0.75$ are the maximally asymmetric weights for the two features.

*Remark 1.* In the extreme case $\xi = 0.5$, both features are constant across all samples. From a purely discriminative perspective, such features carry no information and should be assigned equal (or zero) weight. However, the evolutionary weighting rule still assigns $\gamma_1^* = 0.25$ and $\gamma_2^* = 0.75$, reflecting its inherent bias toward features with lower means. One may view the situation shortly before everything collapses to 1 or 0 as the more meaningful case, where the rule amplifies small deviations as rare traits. In the trivial case the unequal weights can be seen as a consequence of continuity of the formula.

### 4.6 Relation of evolutionary picture and input data

The method introduced in this paper, relates the raw input data $X$ with a well-defined equilibrium $\gamma^*$. The process can be summarized as

$$\text{input data } X \to \text{normalized data } \Phi \to \text{evolutionary fitness} \to \gamma^*.$$

Although some information is lost by using only the mean values $\widetilde{\Phi_j}$ to compute $\gamma^*$, there is often statistical value in comparing different objectives via $\widetilde{\Phi_j} \leftrightarrow \widetilde{\Phi_{j'}}$. In particular, differences in $\widetilde{\Phi_j}$ for $j \in \{1, \ldots, m\}$ often indicate distinct clustering of features in the original data $X$. In this sense, the introduced evolutionary approach provides a means to assess the relevance of features in

multi-objective optimization. It is important to emphasize that this evaluation is not rule-based but depends solely on the input data $X$ and the dynamics of the associated evolutionary representation.

### 4.7 Conclusion & Outlook

We have shown that the evolutionary approach to solving discrete multi-objective optimization problems first introduced in [22] converges to a well-defined equilibrium. In addition, we have analyzed the behavior with respect to larger number of objectives. Even in this limiting case, we have seen that the formalism allows for a meaningful evolutionary interpretation.

It has become clear that the introduced method represents a versatile way to analyze and solve multi-objective problems from a wide range of applications. Further investigation is needed into the comparison with other methods for choosing the feature weights.

In terms of limits to the evolutionary representation, we have seen that the existence of a normalization mapping $X \to \Phi$ is required to allow for a meaningful comparison of features in the optimization. In practice, for many cases of input data $X$, the choice of normalization is straightforward and does not appear to limit the evolutionary approach to specific fields of application.

Restricting to a single choice of evolutionary strategies that yield $\Delta_j^{dom}$ and $\Delta_j^{bal}$ has been done for introductory reasons. It is apparent that there exists a wide range of additional ways to govern the dynamics of genes and organisms - examples in biological systems are altruistic or selfish behavior of one gene or organism with respect to each other. We leave the analysis of the convergence of other evolutionary strategies to future work. When multiple evolutionary strategies are allowed in an optimization, the question arises of how organisms and genes determine which combination of strategies to adopt. As shown in [22], several definitions of this behavior exist, yet a thorough mathematical analysis and comparison of these approaches remains to be done.

Targeting a complete data-independent evolutionary method without externally imposed rules, parameters, or weights, the pre-defined choice of the normalization functions $X \to \Phi$ appears to be a central limitation. However, many statistical methods exist to choose the normalization according to the feature distribution of an objective in $X$. One goal for future improvement would therefore be to assign the objectives to just two categories, 'gain' or 'cost', and to automatically select the normalization from there.

In conclusion, the introduced evolutionary formalism for multi-objective optimization problems constitutes a promising direction for further theoretical and applied investigation. Evolutionary strategies provide new insight into the underlying mechanisms of general data problems.

## References

1. Arthur, W.B.: *Inductive Reasoning and Bounded Rationality.* Amer. Econ. Rev. 84(2), 406–411 (1994)
2. Blansché, A., Gançarski, P., Korczak, J.J.: Genetic Algorithms for Feature Weighting: Evolution vs. Coevolution and Darwin vs. Lamarck. In: Gelbukh, A., de Albornoz, Á., Terashima-Marín, H. (eds.) MICAI 2005. LNCS, vol. 3789, pp. 682–691. Springer, Berlin (2005). https://doi.org/10.1007/11579427_69
3. Cheng, R., Jin, Y., Olhofer, M., Sendhoff, B.: A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Trans. Evol. Comput. 20(5), 773–791 (2016). doi:10.1109/TEVC.2016.2519378
4. Czajkowski, K., Fitzgerald, S., Foster, I., Kesselman, C.: Grid information services for distributed resource sharing. In: Proc. 10th IEEE International Symposium on High Performance Distributed Computing (HPDC), pp. 181–194. IEEE, San Francisco (2001). doi:10.1109/HPDC.2001.945188
5. Dawkins, R.: The Selfish Gene. 30th anniversary edition. Oxford University Press (2006)
6. Dawkins, R.: The Extended Phenotype: The Long Reach of the Gene. Oxford University Press (1982)
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6(2), 182–197 (2002). doi:10.1109/4235.996017

8. de Amorim, R.C.: A survey on feature weighting based K-Means algorithms. Journal of Classification, 33(2), 210–242 (2016). https://doi.org/10.1007/s00357-016-9208-4
9. Foster, I., Kesselman, C., Nick, J., Tuecke, S.: The physiology of the grid: an open grid services architecture for distributed systems integration. Technical report, Global Grid Forum (2002)
10. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
11. Frank, S.A.: *Natural selection. IV. The Price equation.* J. Evol. Biol. 25(6), 1002–1019 (2012)
12. Gu, F., Wang, Y., Wu, Z.: DMOEA/D: A diversity-maintained decomposition-based multiobjective evolutionary algorithm. Soft Comput. 16, 1551–1569 (2012). doi:10.1007/s00500-012-0822-4
13. Inza, I., Larrañaga, P., Etxeberria, R., Sierra, B.: Feature subset selection by Bayesian network–based optimization. Artif. Intell. 123(1), 157–184 (2000). https://doi.org/10.1016/S0004-3702(00)00052-7
14. Jiang, S., Ong, Y.S., Zhang, J., Feng, L.: pa$\lambda$-MOEA/D: Pareto-adaptive weight vectors in decomposition-based multiobjective evolutionary algorithm. IEEE Trans. Evol. Comput. 15(6), 896–912 (2011). doi:10.1109/TEVC.2011.2148191
15. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Proc. AAAI-92, pp. 129–134. AAAI Press (1992)
16. Kononenko, I.: Estimating attributes: Analysis and extensions of RELIEF. In: Bergadano, F., De Raedt, L. (eds.) ECML'94. LNCS, vol. 784, pp. 171–182. Springer, Berlin (1994). https://doi.org/10.1007/3-540-57868-4_57
17. Ma, X., Yu, Y., Li, X., Qi, Y., Zhu, Z.: A Survey of Weight Vector Adjustment Methods for Decomposition-Based Multiobjective Evolutionary Algorithms. IEEE Trans. Evol. Comput. 24(4), 634–649 (2020). doi:10.1109/TEVC.2020.2978158
18. May, P., Ehrlich, H.-C., Steinke, T.: ZIB structure prediction pipeline: composing a complex biological workflow through web services. In: Nagel, W.E., Walter, W.V., Lehner, W. (eds.) Euro-Par 2006. LNCS, vol. 4128, pp. 1148–1158. Springer, Heidelberg (2006). doi:10.1007/11823285_121
19. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. 27(8), 1226–1238 (2005). https://doi.org/10.1109/TPAMI.2005.159
20. Qi, Y., Ma, X., Liu, F., Jiao, L., Sun, J., Wu, J.: MOEA/D with adaptive weight adjustment. Evol. Comput. 22(2), 231–264 (2014). https://doi.org/10.1162/EVCO_a_00109
21. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. J. Mol. Biol. 147, 195–197 (1981). doi:10.1016/0022-2836(81)90087-5
22. Wissgott, P.: Genetic AI: Evolutionary Games for ab initio dynamic Multi-Objective Optimization. arXiv preprint (2025). https://doi.org/10.48550/arXiv.2501.19113
23. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. 11(6), 712–731 (2007). doi:10.1109/TEVC.2007.892759
24. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results. Evol. Comput. 8(2), 173–195 (2000). doi:10.1162/106365600568202
25. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK Report 103, Computer Engineering and Networks Laboratory (TIK), ETH Zurich (2001). https://doi.org/10.3929/ethz-a-004284029