

12-1-2023

Hyperbolic matrix factorization improves prediction of drug-target associations

Aleksandar Poleksic
University of Northern Iowa

Let us know how access to this document benefits you

Copyright ©2023 Aleksandar Poleksic. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

Follow this and additional works at: <https://scholarworks.uni.edu/facpub>

Recommended Citation

Poleksic, Aleksandar, "Hyperbolic matrix factorization improves prediction of drug-target associations" (2023). *Faculty Publications*. 5354.

<https://scholarworks.uni.edu/facpub/5354>

This Article is brought to you for free and open access by UNI ScholarWorks. It has been accepted for inclusion in Faculty Publications by an authorized administrator of UNI ScholarWorks. For more information, please contact scholarworks@uni.edu.



OPEN

Hyperbolic matrix factorization improves prediction of drug-target associations

Aleksandar Poleksic

Past research in computational systems biology has focused more on the development and applications of advanced statistical and numerical optimization techniques and much less on understanding the geometry of the biological space. By representing biological entities as points in a low dimensional Euclidean space, state-of-the-art methods for drug-target interaction (DTI) prediction implicitly assume the flat geometry of the biological space. In contrast, recent theoretical studies suggest that biological systems exhibit tree-like topology with a high degree of clustering. As a consequence, embedding a biological system in a flat space leads to distortion of distances between biological objects. Here, we present a novel matrix factorization methodology for drug-target interaction prediction that uses hyperbolic space as the latent biological space. When benchmarked against classical, Euclidean methods, hyperbolic matrix factorization exhibits superior accuracy while lowering embedding dimension by an order of magnitude. We see this as additional evidence that the hyperbolic geometry underpins large biological networks.

Computational methods for biological relationship inference use dimension reduction techniques to represent biological objects as points in a low-dimensional space. The underlying assumption is that biological systems have low intrinsic dimension. For instance, it has been well established that most variations in genomic databases can be explained by a small set of features, such as the cell state, the cell type, or a gene program¹. In a different example, the low dimensionality of databases of drugs' adverse reactions is due to associations of side-effects to chemical substructures and their combinations^{2,3}. To put it differently, it is known that drugs sharing chemical substructures give rise to same adverse reactions.

The research on dimensionality reduction and associated relationship prediction has traditionally focused on the development and applications of advanced computational and statistical techniques while taking the Euclidean geometry of the native biological space for granted. However, recent theoretical studies challenge the flat geometry assumption⁴⁻¹⁰. According to these studies, complex systems exhibit tree-like topology with high degree of clustering. Therefore, embedding those systems into the Euclidean space inevitably leads to distortion of distances between individual objects and, in turn, compromises the accuracy of relationship inference. In contrast, a negatively curved space can accommodate the exponential growth in the number of relevant network features since the area of a hyperbolic circle is an exponential function of its radius (Fig. 1).

Recent years have seen the development of practical algorithms that use hyperbolic geometry to model complex networks¹¹⁻¹⁸. Papadopoulos et al. developed the HyperMap method for mapping a complex network into a hyperbolic space⁵. Muscoloni et al. address the same problem using a technique based on the angular coalescence principle¹⁴. Monath et al. use a representation of tree structures in the Poincaré ball to design more accurate hierarchical clustering methods¹⁵. Mirvakhabova et al. propose a hyperbolic autoencoder algorithm for the classical collaborative filtering task¹⁶. Vinh Tran et al. propose a novel way of exploring metric learning for recommender systems in a hyperbolic space¹⁷. Schmeier et al. use Poincaré embeddings of hierarchical entities to develop and prioritize playlists for users of digital music services¹⁸. Hyperbolic distance learning has also been incorporated into artificial neural network models, for instance to encode the chemical structures of drugs¹⁹.

In this paper, we show how hyperbolic latent space can be utilized to increase the accuracy of matrix factorization. While our algorithm has been benchmarked on drug-target interaction datasets, the same technique can be applied to other relationship inference tasks (e.g., to predict drug-disease or drug-side effect associations, user preferences to movies or songs, etc.).

We emphasize that improving matrix factorization techniques is of particular importance in recommender systems, since a carefully designed matrix factorization method is known to outperform deep learning in many

Department of Computer Science, University of Northern Iowa, Cedar Falls, IA 50614, USA. email: aleksandar.poleksic@uni.edu

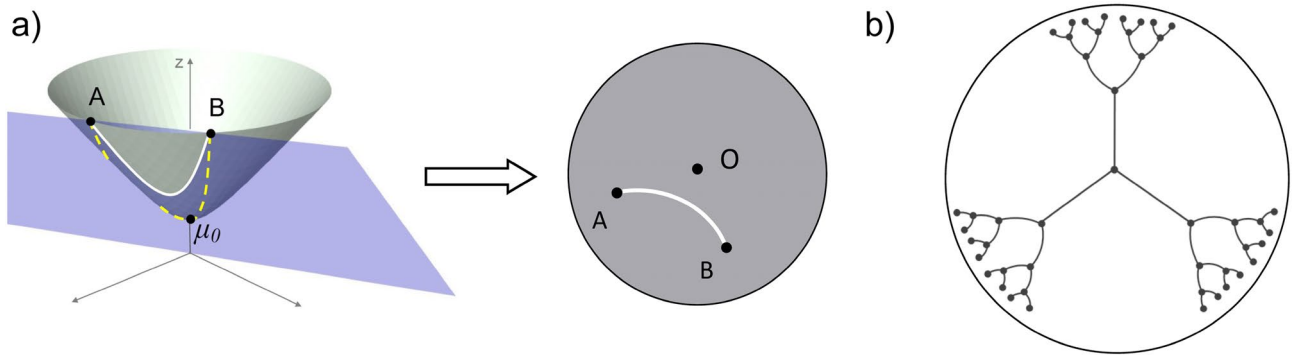


Figure 1. (a) Left: Hyperboloid model of \mathbb{H}^2 . The shortest path between points A and B is the line of intersection of the hyperboloid with the plane (blue) that passes through A, B and O. As A and B are moving away from the vertex μ_0 , the length of this geodesic line (white) is almost the same as the length of the path through μ_0 (orange). Right: Projection onto the Poincaré disk. (b) Effective embedding of a tree into the hyperbolic space (all tree edges are of the same length).

collaborative filtering applications. Specifically, while deep learning can theoretically optimize any function, learning a simple Euclidean dot product (employed in matrix factorization) is shown to be a non-trivial task²⁰.

We incorporated hyperbolic latent space representation into the logistic matrix factorization framework, which is widely used in drug-target association prediction methods. We demonstrate that using the hyperbolic distance in place of the Euclidean distance results in significant accuracy improvements, while lowering the latent space dimension by more than an order of magnitude.

The rest of this article is organized as follows. "The theoretical foundation" section provides a short introduction into the hyperbolic geometry. In "Computing the prior distribution" and "The loss function" sections, we derive a hyperbolic variant of the logistic loss function used in several state-of-the-art matrix factorization method^{21–24}. "Alternating gradient descent in hyperbolic space" section describes an alternating gradient descent procedure for minimizing the loss function. In "Hyperbolic neighborhood regularization and cold-start" section, we develop the hyperbolic versions of the neighborhood regularization and cold-start procedures. Finally, in the Results section we discuss the accuracy of hyperbolic and Euclidean matrix factorization algorithms on some widely used drug-target interaction test sets.

Methods

The theoretical foundation. Hyperbolic geometry can be modeled on the n -dimensional hyperboloid in the Lorentzian space $\mathbb{R}^{n,1}$ (Fig. 1), where $\mathbb{R}^{n,1}$ is a copy of \mathbb{R}^{n+1} equipped with a bilinear form $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ defined as

$$\langle x, y \rangle_{\mathcal{L}} = x_1 y_1 + \dots + x_n y_n - x_{n+1} y_{n+1}. \tag{1}$$

Hyperbolic space is represented by one sheet of the two-sheeted hyperboloid

$$\{x \in \mathbb{R}^{n,1} \mid \langle x, x \rangle_{\mathcal{L}} = -1\} \tag{2}$$

(which can be thought of as a sphere of radius $i = \sqrt{-1}$), namely,

$$\mathbb{H}^n = \{x \in \mathbb{R}^{n,1} \mid \langle x, x \rangle_{\mathcal{L}} = -1, x_{n+1} > 0\}. \tag{3}$$

It can be shown that the bilinear form $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ restricted on the tangent space $T_p \mathbb{H}^n$ at a point $p \in \mathbb{H}^n$, defined by

$$T_p \mathbb{H}^n = \{x \in \mathbb{R}^{n,1} \mid \langle p, x \rangle_{\mathcal{L}} = 0\}. \tag{4}$$

is positive definite, thereby providing a genuine Riemannian metric on \mathbb{H}^n . The distance between two points $x, y \in \mathbb{H}^n$ is given by

$$d_{\mathbb{H}^n}(x, y) = \operatorname{arccosh}(-\langle x, y \rangle_{\mathcal{L}}). \tag{5}$$

An interesting (and in the biological context insightful) property of the hyperbolic space is that the shortest path between two random points in \mathbb{H}^n that are far away from the vertex μ_0 has almost the same length as the path through the vertex (Fig. 1). This resembles the property of the distance function on trees, where the shortest path between two randomly selected nodes deep in the tree is almost of the same length as the path through the root.

While the hyperbolic matrix factorization, outlined below, is applicable to different loss functions, we illustrate it in the framework of logistic matrix factorization. Logistic factorization technique is statistically sound, simple to present, and highly accurate in biological applications^{21–28}.

Let $A = \{a^i\}_{i=1}^m$ be the set of drugs and $B = \{b^j\}_{j=1}^n$ the set of targets (proteins). Denote by $R = (r_{ij})_{m \times n}$ the matrix of relationships (edges) between the elements of A and B. Specifically, $r_{ij} = 1$ if a^i interacts with b^j and $r_{ij} = 0$ otherwise (no interaction or unknown). Let $u^i, v^j \in \mathbb{H}^d$ be the latent vector representations of a^i and b^j , respectively, where $d \ll \max(m, n)$. Denote by $e_{i,j}$ the event that a^i interacts with b^j . In line with the classical

(Euclidean) logistic matrix factorization technique^{21–33}, we model the probability p_{ij} of e_{ij} as the logistic function in the Lorentz space $\mathbb{R}^{d,1}$

$$p_{ij} = p(r_{ij} = 1|u^i, v^j) = \frac{\exp(-d_{\mathcal{L}}^2(u^i, v^j))}{1 + \exp(-d_{\mathcal{L}}^2(u^i, v^j))}, \tag{6}$$

where $d_{\mathcal{L}}^2(x, y)$ denotes the squared Lorentzian distance³⁴ between the points $x, y \in \mathbb{H}^d$, namely

$$d_{\mathcal{L}}^2(x, y) = \|x - y\|_{\mathcal{L}}^2 = \langle x - y, x - y \rangle_{\mathcal{L}} = -2 - 2\langle x, y \rangle_{\mathcal{L}}. \tag{7}$$

Denote by $W = (w_{ij})_{m \times n}$ our confidence in the entries r_{ij} of the interaction matrix R . In many practical applications, $w_{ij} = 1$ if $r_{ij} = 0$, and $w_{ij} = c$ if $r_{ij} = 1$, where $c > 1$ is a constant²¹. In general, the idea is to assign higher weights to trustworthy pairs i.e., those for which we have higher confidence of interaction. Given the weights w_{ij} , the likelihood of r_{ij} given u^i and v^j is

$$p(r_{ij}|u^i, v^j) = p_{ij}^{w_{ij}r_{ij}} (1 - p_{ij})^{w_{ij}(1-r_{ij})}. \tag{8}$$

Thus, assuming the independence of events e_{ij} , it follows that

$$p(R|U, V) = \prod_{ij} \left(p_{ij}^{r_{ij}} (1 - p_{ij})^{1-r_{ij}} \right)^{w_{ij}}, \tag{9}$$

where U and V represent the matrices of latent preferences of elements from A and B , respectively (in other words, the i th row of U is the vector u^i and i th row of V is v^i).

Computing the prior distribution. Similar to the Euclidean case^{21,31}, our goal is to derive the probability $p(U, V|R)$ from (9) through the Bayesian inference.

Utilizing the recent work on wrapped normal distribution in hyperbolic space³⁵, we define the prior distributions as

$$p(U|\sigma_U^2) = \prod_{i=1}^m \mathcal{G}(u_i|\mu_0, \sigma_U^2 I),$$

$$p(V|\sigma_V^2) = \prod_{j=1}^n \mathcal{G}(v_j|\mu_0, \sigma_V^2 I), \tag{10}$$

where $\mathcal{G}(\mu, \Sigma)$ is the pseudo-hyperbolic Gaussian distribution and $\mu_0 = (0, \dots, 0, 1)$ is the vertex of the hyperboloid (the origin of the hyperbolic space).

The pseudo-hyperbolic Gaussian distribution extends the notion of Gaussian distribution to the hyperbolic space (Fig. 2). In short, for $\mu \in \mathbb{H}^d$ and positive definite Σ , sampling from $\mathcal{G}(\mu, \Sigma)$ can be thought of as a three step process: (a) Sample a vector $x \in T_{\mu_0}\mathbb{H}^d$ from $\mathcal{N}(0, \Sigma)$, (b) Transport x along the geodesic joining the points $\mu_0 \in \mathbb{H}^d$ and $\mu \in \mathbb{H}^d$ to $y \in T_{\mu}\mathbb{H}^d$, and (c) Project y to $z \in \mathbb{H}^d$.

The step (b) is carried out using the parallel transport $g_{\mu_0 \rightarrow \mu} : T_{\mu_0}\mathbb{H}^d \rightarrow T_{\mu}\mathbb{H}^d$ (Fig. 3a), defined by

$$g_{\mu_0 \rightarrow \mu}(x) = x + \frac{\langle \mu + \langle \mu_0, \mu \rangle_{\mathcal{L}}, \mu_0, x \rangle_{\mathcal{L}}}{1 - \langle \mu_0, \mu \rangle_{\mathcal{L}}} (\mu_0 + \mu), \tag{11}$$

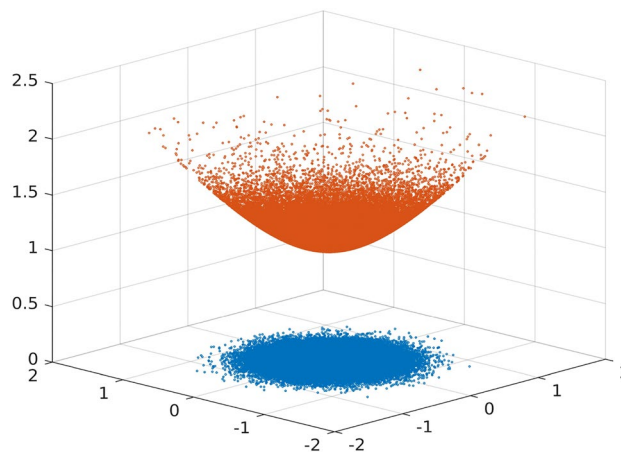


Figure 2. 100,000 samples from $\mathcal{N}(0, \Sigma)$ (blue) and the corresponding samples from $\mathcal{G}(\mu_0, \Sigma)$ (red), $\Sigma = 0.1 \cdot I_{2 \times 2}$.

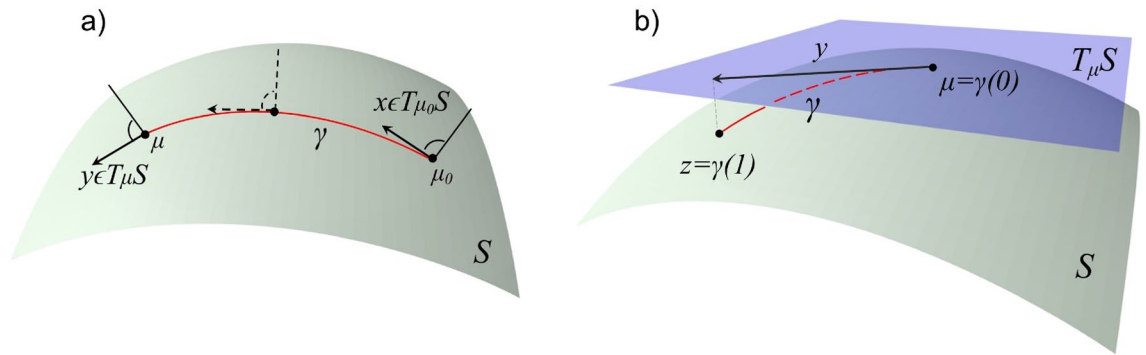


Figure 3. (a) Parallel transport of $x \in T_{\mu_0}S$ to $y \in T_{\mu}S$ along the geodesic γ , where $\mu_0 = \gamma(0)$ and $\mu = \gamma(1)$. (b) The exponential map.

while the step (c) uses the exponential map $Exp_{\mu} : T_{\mu}\mathbb{H}^d \rightarrow \mathbb{H}^d$ (Fig. 3b), defined by

$$Exp_{\mu}(y) = \cosh(\|y\|_{\mathcal{L}})\mu + \sinh(\|y\|_{\mathcal{L}})\frac{y}{\|y\|_{\mathcal{L}}}, \tag{12}$$

where $\|y\|_{\mathcal{L}} = \sqrt{\langle y, y \rangle_{\mathcal{L}}}$.

It is not difficult to show that the length of the geodesic joining μ to $Exp_{\mu}(y)$ on \mathbb{H}^d is equal to $\|y\|_{\mathcal{L}}$, i.e., $d_{\mathbb{H}^d}(\mu, Exp_{\mu}(y)) = \|y\|_{\mathcal{L}}$. The relationship between the probability densities $X \sim \mathcal{N}(0, \Sigma)$ and $Z \sim \mathcal{G}(\mu, \Sigma)$ is

$$p(x) = p(z)\det(J_f), \tag{13}$$

where $f = Exp_{\mu} \circ g_{\mu_0 \rightarrow \mu}$ and $\det(J_f)$ denotes the determinant of the Jacobian $J_f = \left| \frac{\partial f}{\partial x} \right|$ ³⁵. Finally, it can be shown that

$$\ln p(z) = \ln p(x) - (d-1)\ln \frac{\sinh(r)}{r}, \tag{14}$$

where $r = \text{arccosh}(-\langle \mu, z \rangle_{\mathcal{L}})$ ³⁵.

The loss function. With the prior placed on U and V , we return to calculating the posterior probability $p(U, V|R)$ through the Bayesian inference

$$p(U, V|R) \propto p(R|U, V)p(U|\sigma^2)p(V|\sigma^2). \tag{15}$$

Following the Euclidean matrix factorization, we take the logarithm of the posterior distribution (15) to arrive at the closed form expression for the loss function

$$\begin{aligned} L = & \sum_{i=1}^m \sum_{j=1}^n w_{ij} \left[\ln(1 + e^{-d_{\mathcal{L}}^2(u^i, v^j)}) + r_{ij} d_{\mathcal{L}}^2(u^i, v^j) \right] \\ & - \sum_{i=1}^m \left[\ln p(\bar{u}^i) - (d-1)\ln \frac{\sinh(\|\bar{u}^i\|_{\mathcal{L}})}{\|\bar{u}^i\|_{\mathcal{L}}} \right] \\ & - \sum_{j=1}^n \left[\ln p(\bar{v}^j) - (d-1)\ln \frac{\sinh(\|\bar{v}^j\|_{\mathcal{L}})}{\|\bar{v}^j\|_{\mathcal{L}}} \right]. \end{aligned} \tag{16}$$

In the expression above, p is the probability density function of the normal distribution $\mathcal{N}(0, \sigma^2 I)$ in the tangent space $T_{\mu_0}\mathbb{H}^d$ at the vertex $\mu_0 = (0, \dots, 0, 1)$ and, for $x = (x_1, \dots, x_d, x_{d+1}) \in \mathbb{H}^d$,

$$\bar{x} = Exp_{\mu_0}^{-1}x = \frac{\text{arccosh}(-\langle \mu_0, x \rangle_{\mathcal{L}})}{\sqrt{\langle \mu_0, x \rangle_{\mathcal{L}}^2 - 1}}(x + \langle \mu_0, x \rangle_{\mathcal{L}} \cdot \mu_0) = \frac{\text{arccosh}(x_{d+1})}{\sqrt{x_{d+1}^2 - 1}}(x_1, \dots, x_d, 0). \tag{17}$$

Thus,

$$\ln p(\bar{x}) = -\frac{1}{2\sigma^2} \text{arccosh}^2(x_{d+1}) + C_1, \tag{18}$$

where C_1 is a constant. Moreover, since

$$\|\bar{x}\|_{\mathcal{L}} = \operatorname{arccosh}(-\langle \mu_0, x \rangle_{\mathcal{L}}) = \operatorname{arccosh}(x_{d+1}), \tag{19}$$

It follows that

$$\frac{\sinh(\|\bar{x}\|_{\mathcal{L}})}{\|\bar{x}\|_{\mathcal{L}}} = \frac{\sqrt{x_{d+1}^2 - 1}}{\operatorname{arccosh}(x_{d+1})}. \tag{20}$$

Hence, our loss function has the following form:

$$\begin{aligned} L = & \sum_{i=1}^m \sum_{j=1}^n w_{ij} \left[\ln \left(1 + e^{-d_{\mathcal{L}}^2(u^i, v^j)} \right) + r_{ij} d_{\mathcal{L}}^2(u^i, v^j) \right] \\ & + \sum_{i=1}^m \left[\alpha_U \operatorname{arccosh}^2(u_{d+1}^i) + (d-1) \ln \frac{\sqrt{(u_{d+1}^i)^2 - 1}}{\operatorname{arccosh}(u_{d+1}^i)} \right] \\ & + \sum_{j=1}^n \left[\alpha_V \operatorname{arccosh}^2(v_{d+1}^j) + (d-1) \ln \frac{\sqrt{(v_{d+1}^j)^2 - 1}}{\operatorname{arccosh}(v_{d+1}^j)} \right] + C, \end{aligned} \tag{21}$$

where $\alpha_U = \frac{1}{2\sigma_U^2}, \alpha_V = \frac{1}{2\sigma_V^2}$ are trainable parameters and C is a constant.

Alternating gradient descent in hyperbolic space. Minimizing a real function defined in a d -dimensional Euclidean space \mathbb{R}^d is routinely accomplished using the gradient descent technique. We adopt a similar method for finding the point $u \in \mathbb{H}^d$ of a local minimum of any real valued function $f : \mathbb{H}^d \rightarrow \mathbb{R}$ ^{36,37}. For this strategy to work, the function f must be defined in the ambient space $\mathbb{R}^{d,1}$ of \mathbb{H}^d , as well as on \mathbb{H}^d . Specifically, given the initial value $u = u^{(0)}$ and a step size η , the gradient descent in hyperbolic space can be carried out by repeating the following steps:

1. Compute the gradient $\nabla_u^{\mathbb{R}^{d,1}} f$
2. Project $\nabla_u^{\mathbb{R}^{d,1}} f$ orthogonally to vector $\nabla_u^{\mathbb{H}^d} f \in T_u \mathbb{H}^d$
3. Set $u^{new} = \operatorname{Exp}_u(-\eta \nabla_u^{\mathbb{H}^d} f)$

The gradient $\nabla_u^{\mathbb{R}^{d,1}} f$ in the ambient space $\mathbb{R}^{d,1}$ is a vector of partial derivatives

$$\nabla_u^{\mathbb{R}^{d,1}} f = \left(\frac{\partial L}{\partial x_1} \Big|_u, \dots, \frac{\partial L}{\partial x_n} \Big|_u, -\frac{\partial L}{\partial x_{d+1}} \Big|_u \right) \tag{22}$$

(note the negative sign of the last vector's component).

The above representation of the gradient follows directly from its definition:

$$\forall v \in \mathbb{R}^{d,1}, \langle \nabla_u^{\mathbb{R}^{d,1}} f, v \rangle_{\mathcal{L}} = D_v f(u). \tag{23}$$

The orthogonal projection from the ambient space onto the tangent space in (step 2 above) is given by

$$\nabla_u^{\mathbb{H}^d} f = \nabla_u^{\mathbb{R}^{d,1}} f + \langle u, \nabla_u^{\mathbb{R}^{d,1}} f \rangle_{\mathcal{L}} u. \tag{24}$$

We use the ‘‘alternating gradient descent’’ method to minimize the error function $L_{A,B}$ given in (21). The partial derivatives of $L_{A,B}$ are

$$\frac{\partial L}{\partial u_k^i} = 2 \sum_{j=1}^n w_{ij} (p_{ij} - r_{ij}) v_k^j, k \leq d \tag{25}$$

$$\begin{aligned} \frac{\partial L}{\partial u_{d+1}^i} = & 2 \sum_{j=1}^n w_{ij} (r_{ij} - p_{ij}) v_{d+1}^j \\ & + 2\alpha_U \frac{\operatorname{arccosh}(u_{d+1}^i)}{\sqrt{(u_{d+1}^i)^2 - 1}} \\ & + (d-1) \frac{u_{d+1}^i \operatorname{arccosh}(u_{d+1}^i) - \sqrt{(u_{d+1}^i)^2 - 1}}{\left[(u_{d+1}^i)^2 - 1 \right] \operatorname{arccosh}(u_{d+1}^i)} \end{aligned} \tag{26}$$

Figure 4 shows the pseudocode of our algorithm.

ALGORITHM Hyperbolic Alternating Gradient Descent

```

// Input: Relationship matrix  $R$ , weight matrix  $W$ , number of epochs  $E$ ,
// learning rate  $\eta$ , parameters  $\sigma_U, \sigma_V$ , latent space dimension  $d$ 
// Output: The matrices of latent preferences  $U = (u_k^i)$  and  $V = (v_k^j)$ 
for  $i = 1$  to  $m$ 
  sample  $u^i$  from  $\mathcal{G}(\mu_0, \sigma_U^2 I)$ 
end for
for  $j = 1$  to  $n$ 
  sample  $v^j$  from  $\mathcal{G}(\mu_0, \sigma_V^2 I)$ 
end for
for  $e = 1$  to  $E$ 
  for  $i = 1$  to  $m$ 
    compute  $\nabla_{u^i}^{\mathbb{R}^{d,1}} L$ 
    project  $\nabla_{u^i}^{\mathbb{R}^{d,1}} L$  orthogonally to  $\nabla_{u^i}^{\mathbb{H}^d} L \in T_{u^i} \mathbb{H}^d$ 
     $u^i = \text{Exp}_{u^i}(-\eta \nabla_{u^i}^{\mathbb{H}^d} L)$ 
  end for
  for  $i = j$  to  $n$ 
    compute  $\nabla_{v^j}^{\mathbb{R}^{d,1}} L$ 
    project  $\nabla_{v^j}^{\mathbb{R}^{d,1}} L$  orthogonally to  $\nabla_{v^j}^{\mathbb{H}^d} L \in T_{v^j} \mathbb{H}^d$ 
     $v^j = \text{Exp}_{v^j}(-\eta \nabla_{v^j}^{\mathbb{H}^d} L)$ 
  end for
end for

```

Figure 4. Pseudocode of the hyperbolic gradient descent procedure.

Hyperbolic neighborhood regularization and cold-start. A standard way to increase the accuracy of relationship inference between the elements of two biological domains A and B is to employ the so-called *neighborhood regularization*. The goal is to ensure that similar entities from A are in relationship with similar entities from B (e.g., similar drugs interact with similar genes). To achieve this, we extend the Euclidean neighborhood regularization method^{21,38} to \mathbb{H}^d by adding the following term to the loss function $L(21)$:

$$\beta_U \sum_{i=1}^m \sum_{j=1}^m s_{i,j} d_{\mathcal{L}}^2(u^i, u^j) + \beta_V \sum_{i=1}^n \sum_{j=1}^n t_{i,j} d_{\mathcal{L}}^2(v^i, v^j), \quad (27)$$

where $s_{i,j}$ (respectively $t_{i,j}$) is the value reflecting the similarity between a^i and a^j (respectively b^i and b^j) and β_U, β_V are trainable (neighborhood regularization) parameters.

A separate procedure is needed to address the “cold-start” problem i.e., the arrival of a new node (a node with no known relationships to other nodes). In the setting of drug-target interaction prediction, this procedure is used to predict targets for new compounds (such as a chemical in pre-clinical studies) and vice versa.

For the hyperbolic cold-start, we use a hyperbolic variant of the Euclidean weighted-profile method^{21,31,33}. Specifically, the latent vector $u^i \in \mathbb{R}^d$ for a drug $a^i \in A$ that does not interact with any protein $b^j \in B$ (i.e., the i^{th} row of R is empty) is computed as the weighted combination of the rows $u^j \in U$ most similar to u^i . Specifically,

$$u^i = \frac{1}{SM} \sum_{j=1}^J s_{i,j} u^j, \quad (28)$$

where $SM = \sum_{j=1}^J s_{i,j}$ and J is a pre-defined number of nearest neighbors. The hyperbolic center of mass u^i is computed as in Law et al.³⁹.

Results

Benchmarking experiments. We benchmarked the hyperbolic matrix factorization on four drug-target interaction test sets, specifically Nr, Gpcr, Ion, and Enz⁴⁰, using four traditional classification measures, namely the area under the receiver operating characteristics curve (AUC), the area under the precision-recall curve

(AUPR), precision at top ten (PREC@10), and the average precision (AP). An extensive grid search is employed to train the parameters of each method (see the Supplementary Data).

In our first benchmark, we assessed the advantage of the basic logistic hyperbolic matrix factorization over the classical Euclidean matrix factorization (as implemented in the popular NRLMF method²¹), in absence of any side-information (i.e., the pairwise drug and the pairwise protein similarity). As described in the “Methods” section, the hyperbolic method is conceptually the same as the Euclidean method, but it uses $-d_{\mathcal{L}}^2(x, y) = 2 + 2\langle x, y \rangle_{\mathcal{L}}$ in place of $\langle x, y \rangle$ and uses the pseudo-hyperbolic Gaussian distribution (10) in place of the Gaussian prior.

We submit each method (Euclidean and hyperbolic) to ten rounds of the fivefold cross-validation (CV) test (also known as CVP test²²). In each CV round, the data set under consideration (i.e., the drug-target association matrix) is randomly split into 5 groups. Each group is used once as test data, while the remaining four groups represent training data. Hence, every (interacting and non-interacting) drug-target pair is scored once in each CV round. The final classification score (AUC, AUPR, PREC@10, AP) assigned to each DTI prediction method is computed by averaging classification scores obtained across different CV rounds.

As seen in Table 1, the bare-bone hyperbolic matrix factorization routinely outperforms the bare-bone Euclidean factorization in identifying four types of drug targets (Nr, Gpcr, Ion, and Enz) and across fundamentally different classification measures (AUC, AUPR, PREC10, AP).

Interestingly, the hyperbolic matrix factorization achieves superior accuracy at latent dimensions that are by an order of magnitude smaller compared to dimensions needed for an optimal Euclidean embedding. Specifically, optimal Euclidean factorization is most often achieved at ranks exceeding 150. In contrast, most of the time, hyperbolic factorization needs only 5 or 10 latent features to achieve the same or better classification scores (Fig. 5). We view this as additional evidence that the hyperbolic space is the native space of biological networks.

In our second test, we allow both methodologies to use drug and protein homophily information to boost the prediction accuracy. In the classical (Euclidean) setting, we incorporate side-information precisely as done in the NRLMF method²¹. The hyperbolic algorithm uses the same general formula (27), but employs the hyperbolic distances in place of the Euclidean distances. As seen in Table 2, the Euclidean factorization erases some head-start

	AUC	AUPR	PREC10	AP
Nr				
Euc	0.771 ± 0.009	0.340 ± 0.015	0.551 ± 0.010	0.461 ± 0.020
Hyp	0.850 ± 0.007	0.433 ± 0.023	0.647 ± 0.020	0.567 ± 0.028
Gpcr				
Euc	0.884 ± 0.006	0.571 ± 0.011	0.983 ± 0.021	0.987 ± 0.024
Hyp	0.904 ± 0.002	0.612 ± 0.007	1.000 ± 0.000	1.000 ± 0.000
Ion				
Euc	0.968 ± 0.002	0.842 ± 0.005	1.000 ± 0.000	1.000 ± 0.000
Hyp	0.975 ± 0.003	0.861 ± 0.001	1.000 ± 0.000	1.000 ± 0.000
Enz				
Euc	0.951 ± 0.001	0.770 ± 0.002	1.000 ± 0.000	1.000 ± 0.000
Hyp	0.966 ± 0.001	0.809 ± 0.003	1.000 ± 0.000	1.000 ± 0.000

Table 1. Comparison of the basic (no side-information or profile-weighting) Euclidean and hyperbolic logistic matrix factorization algorithm (as implemented in the NRLMF method). The results are obtained using ten rounds of fivefold cross-validation. Better results are shown in bold.

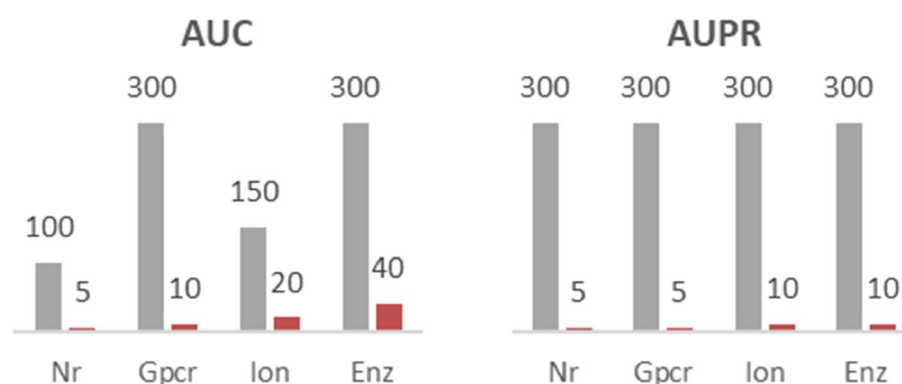


Figure 5. Optimal Euclidean rank (gray) and the hyperbolic rank (red) yielding the same or better AUC and the AUPR scores. Similar results were obtained using the PREC10 metric and the AP metric.

	AUC	AUPR	PREC10	AP
Nr				
Euc	0.954 ± 0.002	0.642 ± 0.013	0.808 ± 0.044	0.755 ± 0.006
Hyp	0.970 ± 0.004	0.697 ± 0.019	0.874 ± 0.010	0.848 ± 0.028
Gpcr				
Euc	0.962 ± 0.002	0.677 ± 0.010	0.913 ± 0.056	0.872 ± 0.078
Hyp	0.976 ± 0.002	0.710 ± 0.017	0.989 ± 0.010	0.984 ± 0.014
Ion				
Euc	0.984 ± 0.002	0.868 ± 0.005	0.948 ± 0.017	0.890 ± 0.018
Hyp	0.991 ± 0.001	0.890 ± 0.010	1.000 ± 0.000	1.000 ± 0.000
Enz				
Euc	0.983 ± 0.001	0.879 ± 0.006	1.000 ± 0.000	1.000 ± 0.000
Hyp	0.991 ± 0.001	0.899 ± 0.001	1.000 ± 0.000	1.000 ± 0.000

Table 2. Accuracy of the full-blown Euclidean and hyperbolic logistic matrix factorizations in predicting drug-target interactions (10 rounds of fivefold CV test). Both methods are allowed to take advantage of the side-information and profile weighting. Significantly better values are shown in bold.

advantage of hyperbolic factorization in the fivefold CVP test, albeit at much higher latent dimensions. This is somewhat expected, as the side information enables the Euclidean method to approach the theoretical limits on the accuracy that can be achieved on the four noisy, sparse, and biased test sets used in our study.

For a more thorough analysis, we also carried out the above benchmarks using tenfold cross validation. The results of our tenfold CV tests are shown in the Supplementary Tables 1 and 2. Depending on a test set under consideration, tenfold cross validation might be a more meaningful experiment as removing only 10% of the existing network links (as opposed to 20% in a fivefold CV) preserves important structural features of the target network^{41,42}.

While the first two benchmarks help gain insight into the value added by different components of the loss-function, our final benchmark compares the two techniques in the most important and the most difficult cold-start setting. In this experiment, known as Leave-One-Out Cross-Validation (LOOCV), we hide (zero out) and then try to recover all interactions of every drug under consideration. Specifically, for each drug d , we hide (zero out) and then try to recover all interactions (known and unknown) of d with all proteins in the data set. Thus, LOOCV can be viewed as a (non-stochastic) variant of a (single round) m -cross validation procedure, where m is the number of drugs.

To better assess the performance of hyperbolic embedding, we include in the LOOCV benchmark two additional state-of-the-arts methods, namely, DNILMF²², and NGN²⁴. The DNILMF method is like NRLMF, but it incorporates drug and protein homophily directly into the formula for $p_{i,j}$ (6). Moreover, it employs a nonlinear diffusion technique to construct pairwise drug and protein similarity matrices²². The NGN method is also similar in spirit to NRLMF, but it builds a neighborhood-based global network model instead of learning drug and target features separately²⁴.

We constructed a hyperbolic variant of each technique by simply replacing the Euclidean dot product with the negative Lorentzian distance and by replacing the Gaussian prior by the wrapped normal distribution in the hyperbolic space (as discussed in the “Methods” section).

As seen in Table 3, the hyperbolic matrix factorization improves the accuracy of current techniques in predicting protein targets for new compounds, such as the chemicals in preclinical studies or clinical trials. In addition, the Supplementary Table 3 shows that our method improves DTI predictions on isolated samples, namely drug-target pairs (d, t) , where d does not have interacting targets (other than t) and t does not have interacting drugs (other than d).

	AUC	AUPR	PREC10	AP
NRLMF				
Euc	0.905	0.243	0.495	0.549
Hyp	0.926	0.258	0.557	0.612
DNILMF				
Euc	0.895	0.239	0.489	0.546
Hyp	0.923	0.262	0.562	0.622
NGN				
Euc	0.895	0.243	0.506	0.554
Hyp	0.925	0.263	0.558	0.614

Table 3. The accuracy of the Euclidean vs. hyperbolic variants of different matrix factorization methods in the Leave-One-Out Cross-Validation (LOOCV) benchmark. Significantly better values are shown in bold.

Additional tests. Recent years have seen the developments of machine learning algorithms for different biological relationship inference tasks^{43–46}. While many of those methods can be tailored to provide predictions of drug–target interactions, it would be unrealistic to benchmark them all against the methodology presented in this article. Supplementary Table 4 provides the comparison of our technique against the SVM-based algorithm BLM⁴⁷ and the GRGMF—a matrix factorization algorithm⁴⁸.

We were also interested in how our method fares against the Cannistraci's methods⁴⁹ based on the local-community-paradigm (LCP). These methods are simple to interpret as they use a combination of node similarity metrics (directly observable in a bipartite drug–target network), such as the number of common neighbors (CN) and the number of links between those neighbors (LCL). Aside from exhibiting the accuracy superior to that of other unsupervised drug–target link prediction algorithms (and comparable to accuracies of supervised algorithms), Cannistraci's methods are extremely fast (Supplementary Fig. 1) and thus ideally suited for the task of link prediction in large networks. The results of our comparison with the LCP-based methods are shown in the Supplementary Tables 5 and 6.

While our project was, in part, inspired by the recent studies on hyperbolic network embedding, most of those methods, such as Coalescent Embedding (CE)¹⁴, were not specifically tailored for the DTI prediction task. To make a meaningful comparison with CE, we had to first place the two algorithms on the same ground. More precisely, in our tests the inference by CE was conducted based upon the hyperbolic distances between drugs and targets (closer objects are more likely to interact) computed from the coalescent embedding of the drug–target interaction network in the Poincaré disk. We also restricted the embedding dimension in our method to 2 since CE preferably uses the Poincaré disk as the latent space. The classification scores achieved by the two techniques are presented in the Supplementary Tables 7 and 8. We emphasize that, due to the methods' modifications mentioned above, the benchmarking results shown in the supplementary material should be interpreted with caution.

In a quest for high accuracy, some algorithms for DTI prediction utilize biomedical knowledge beyond the protein amino-acid sequences and drug chemical structures, including the information on adverse drug reactions, drug–disease and protein–disease associations, drug-induced gene expression profiles, protein–protein interactions, etc. Such a rich input often leads to information leak, presenting a challenge in evaluating these methods in a classical drug discovery setting where (typically) only a chemical structure of the drug and the primary sequence of the gene is known upfront.

Recent years have also seen the development of methods for drug–target affinity (DTA) prediction^{50–53}. In contrast to DTI prediction methods, DTA algorithms utilize drug–target binding affinity scores and treat DTI as a regression (rather than a binary classification) problem. Moreover, unlike DTI methods, DTA algorithms are typically evaluated on Davis⁵⁴ and KIBA⁵⁵ datasets using Concordance Index (CI), Mean Squared Error (MSE), and similar metrics for regression classification tasks. In fact, aside from KronRLS⁵⁶, very few DTA methods have been assessed in standard DTI benchmarks. While the direct comparison with DTA methods is beyond the scope of this paper, a quick look at the AUPR values in a cross-validation test published by KronRLS authors (Nr: 0.528, Gpcr: 0.602, Ion: 0.765, Enz: 0.829) and the corresponding values computed in our benchmark (Nr: 0.697, Gpcr: 0.710, Ion: 0.890, Enz: 0.899) provide some insight (albeit indirect) into potential benefits of utilizing hyperbolic space to predict drug–target binding affinities.

Discussion and conclusion

Matrix factorization is one of the main techniques used in computational systems biology to uncover relationships between the elements from a pair of biological domains. The technique works by representing the biological objects as points in a low dimensional (latent) space in a way that best explains the input set of known interactions. More precisely, the input matrix of known associations is completed by approximating it as a product of two lower dimensional matrices.

Past research in computational systems biology, including matrix factorization techniques, has taken the Euclidean geometry of the biological space for granted. This has been convenient due to the availability of advanced analytic, numerical, statistical and machine learning procedures in the Euclidean space. However, recent theoretical studies suggest that the hyperbolic geometry, rather than Euclidean, underpins all complex networks in general and the biological networks in particular. Therefore, a radical shift in data representation is necessary to obtain an undistorted view of the biological space and, in turn, ensure further progress in systems biology and related fields.

We have developed and benchmarked a technique for a probabilistic hyperbolic matrix factorization and applied it to predict drug–target interactions. We demonstrate that the Lorentzian model of hyperbolic space allows for a closed form expression of the key transformations and techniques required for latent space dimensionality reduction. Our method builds upon recent advances in the development of probabilistic models and numerical optimization in hyperbolic space to learn an optimal embedding and to compute the probabilities of drug–target interactions. Our benchmarking tests demonstrate a significant increase in accuracy and a drastic reduction in latent space dimensionality of hyperbolic embedding compared to Euclidean embedding. These findings reaffirm the negative curvature of the native biological space.

Although a (bipartite drug–target) hyperbolic network embedding arises as a byproduct of hyperbolic matrix factorization, our focus is on prioritizing targets for a given drug (and vice versa). To better assist structure-based drug discovery, DTI prediction methods focus more on identifying a handful of targets with strong binding affinities and much less on prioritizing many remaining targets with weak interactions (this also explains why the AUPR-like metrics are preferred in computational systems biology). To achieve this goal, DTI prediction methods are willing to distort the network structure away from the immediate neighbors of each drug in order to better model the network in the vicinities of drugs. In our methods, the distortion occurs each time a weighted profile is constructed to address the cold-start problem.

There are several aspects of hyperbolic matrix factorization that this study has not explored in detail, including optimal procedure for gradient descent in hyperbolic space. In contrast to decades of research on Euclidean numerical analysis techniques, the methods for numerical optimization in the hyperbolic space are few and far between. The main difficulty is the numerical instability of the hyperbolic gradient descent in vicinity of cliffs⁵⁷. In this study, we applied a simple heuristic intervention to combat the explosion in the magnitude of the hyperbolic gradient. For our optimization method to converge to a local minimum, we carried out three iterations of the gradient descent procedure, lowering the learning rate on the fly and clipping the gradient if necessary. We believe that further research in this area will add significant value to hyperbolic embedding and inference methods.

Our model uses the same hyperbolic space to represent both drugs and proteins. This widely used approach^{58–60} is applied in our study due to simplicity of algorithm design and the fact that heterogeneous networks are shown to have a metric structure with an effective hyperbolic geometry underneath⁶¹. However, alternative approaches are also worthwhile considering. Viewing biomedical entities (in our case drugs and proteins) as objects residing in spaces of different dimension and curvature, the bipartite graph of their relationships can be realized in the hyperbolic product space⁶². Finding the proper dimension and the curvature of the space that underlines each biological domain is expected to result in a more accurate latent representation and, in turn, more accurate relationship prediction.

Data availability

The code and test sets are available in the github repository https://github.com/poleksic/Hyperbolic_MF.

Received: 24 July 2022; Accepted: 11 January 2023

Published online: 18 January 2023

References

- Ding, J. & Regev, A. Deep generative model embedding of single-cellRNA-Seq profiles on hyperspheres and hyperbolic spaces. *BioRxiv* 853457 (2019).
- Scheiber, J. *et al.* Mapping adverse drug reactions in chemical space. *J. Med. Chem.* **52**(9), 3103–3107 (2009).
- Mizutani, S., Pauwels, E., Stoven, V., Goto, S. & Yamanishi, Y. Relating drug–protein interaction network with drug side effects. *Bioinformatics* **28**(18), i522–i528 (2012).
- Krioukov, D., Papadopoulos, F., Vahdat, A. & Boguná, M. Curvature and temperature of complex networks. *Phys. Rev. E* **80**(3), 035101 (2009).
- Papadopoulos, F., Psomas, C. & Krioukov, D. Network mapping by replaying hyperbolic growth. *IEEE/ACM Trans. Netw.* **23**(1), 198–211 (2014).
- Nickel, M. & Kiela, D. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems*, 6338–6347 (2017).
- Albert, R., DasGupta, B. & Mobasher, N. Topological implications of negative curvature for biological and social networks. *Phys. Rev. E* **89**(3), 032811 (2014).
- Alanis-Lobato, G., Mier, P. & Andrade-Navarro, M. A. Efficient embedding of complex networks to hyperbolic space via their Laplacian. *Sci. Rep.* **6**, 30108 (2016).
- De Sa, C., Gu, A., Ré, C. & Sala, F. Representation tradeoffs for hyperbolic embeddings. *Proc. Mach. Learn. Res.* **80**, 4460 (2018).
- Bose, A. J., Smofsky, A., Liao, R., Panangaden, P. & Hamilton, W. L. Latent variable modelling with hyperbolic normalizing flows. *arXiv preprint arXiv:2002.06336* (2020).
- Dhingra, B., Shalloe, C. J., Norouzi, M., Dai, A. M. & Dahl, G. E. Embedding text in hyperbolic spaces. *arXiv preprint arXiv:1806.04313* (2018).
- Chamberlain, B. P., Clough, J. & Deisenroth, M. P. Neural embeddings of graphs in hyperbolic space. *arXiv preprint arXiv:1705.10359* (2017).
- Leimeister, M., & Wilson, B. J. Skip-gram word embeddings in hyperbolic space. *arXiv preprint arXiv:1809.01498* (2018).
- Muscoloni, A., Thomas, J. M., Ciucci, S., Bianconi, G. & Cannistraci, C. V. Machine learning meets complex networks via coalescent embedding in the hyperbolic space. *Nat. Commun.* **8**(1), 1–19 (2017).
- Monath, N., Zaheer, M., Silva, D., McCallum, A. & Ahmed, A. Gradient-based hierarchical clustering using continuous representations of trees in hyperbolic space. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 714–722 (2019).
- Mirvakhabova, L., Frolov, E., Khrulkov, V., Oseledets, I. & Tuzhilin, A. Performance of hyperbolic geometry models on top-N recommendation tasks. In *Fourteenth ACM Conference on Recommender Systems*, 527–532 (2020).
- Tran, L. V., Tay, Y., Zhang, S., Cong, G., & Li, X. HyperML: a boosting metric learning approach in hyperbolic space for recommender systems. In *WSDM*, 609–617 (2020).
- Schmeier, T., Chisari, J., Garrett, S. & Vintch, B. Music recommendations in hyperbolic space: an application of empirical bayes and hierarchical poincaré embeddings. In *Proceedings of the 13th ACM Conference on Recommender Systems*, 437–441 (2019).
- Yu, K., Visweswaran, S. & Batmanghelich, K. Semi-supervised hierarchical drug embedding in hyperbolic space. *arXiv preprint arXiv:2006.00986* (2020).
- Rendle, S., Krichene, W., Zhang, L. & Anderson, J. Neural collaborative filtering vs. matrix factorization revisited. In *Fourteenth ACM Conference on Recommender Systems*, 240–248 (2020).
- Liu, Y., Wu, M., Miao, C., Zhao, P. & Li, X. Neighborhood regularized logistic matrix factorization for drug–target interaction prediction. *PLoS Comput. Biol.* **12**, e1004760 (2016).
- Hao, M., Bryant, S. H. & Wang, Y. Predicting drug–target interactions by dual-network integrated logistic matrix factorization. *Sci. Rep.* **7**(1), 1–11 (2017).
- Li, Y., Li, J. & Bian, N. DNILMF-LDA: Prediction of lncRNA–disease associations by dual-network integrated logistic matrix factorization and Bayesian optimization. *Genes* **10**(8), 608 (2019).
- Wang, S., Li, J., Wang, Y. & Juan, L. A neighborhood-based global network model to predict drug–target interactions. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **19**(4), 2017–2020 (2021).
- Hao, M., Bryant, S. H. & Wang, Y. Open-source chemogenomic data-driven algorithms for predicting drug–target interactions. *Brief. Bioinform.* **20**(4), 1465–1474 (2019).
- Zhao, Q. *et al.* IRWNLPI: Integrating random walk and neighborhood regularized logistic matrix factorization for lncRNA–protein interaction prediction. *Front. Genet.* **9**, 239 (2018).
- Ban, T., Ohue, M. & Akiyama, Y. NRLMFβ: Beta-distribution-rescored neighborhood regularized logistic matrix factorization for improving the performance of drug–target interaction prediction. *Biochem. Biophys. Rep.* **18**, 100615 (2019).

28. Yan, C. *et al.* DNRLMF-MDA: Predicting microRNA-disease associations based on similarities of microRNAs and diseases. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **16**(1), 233–243 (2017).
29. Steck, H. Training and testing of recommender systems on data missing not at random. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 713–722 (2010).
30. Johnson, C. C. Logistic matrix factorization for implicit feedback data. In *Advances in Neural Information Processing Systems 27: Distributed Machine Learning and Matrix Computations Workshop* (2014).
31. Lim, H., Gray, P., Xie, L. & Poleksic, A. Improved genome-scale multi-target virtual screening via a novel collaborative filtering approach to cold-start problem. *Sci. Rep.* **6**(1), 1–11 (2016).
32. Lim, H. *et al.* Large-scale off-target identification using fast and accurate dual regularized one-class collaborative filtering and its application to drug repurposing. *PLoS Comput. Biol.* **12**(10), e1005135 (2016).
33. Poleksic, A. & Xie, L. Predicting serious rare adverse reactions of novel chemicals. *Bioinformatics* **34**(16), 2835–2842 (2018).
34. Ratcliffe, J. G., Axler, S. & Ribet, K. A. *Foundations of Hyperbolic Manifolds* Vol. 149 (Springer, 2006).
35. Nagano, Y., Yamaguchi, S., Fujita, Y., & Koyama, M. A wrapped normal distribution on hyperbolic space for gradient-based learning. *arXiv preprint arXiv:1902.02992* (2019).
36. Nickel, M. & Kiela, D. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, 3779–3788 (PMLR, 2018).
37. Wilson, B. & Leimeister, M. Gradient descent in hyperbolic space. *arXiv preprint arXiv:1805.08207* (2018).
38. Yao, Y., Tong, H., Yan, G., Xu, F., Zhang, X., Szymanski, B. K. & Lu, J. Dual-regularized one-class collaborative filtering. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, 759–768 (2014).
39. Law, M., Liao, R., Snell, J. & Zemel, R. Lorentzian distance learning for hyperbolic representations. In *International Conference on Machine Learning*, 3672–3681 (2019).
40. Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W. & Kanehisa, M. Prediction of drug–target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics* **24**(13), i232–i240 (2008).
41. Zhou, T. Progresses and challenges in link prediction. *Iscience* **24**(11), 103217 (2021).
42. Mussolini, A. & Cannistraci, C. V. “Stealing fire or stacking knowledge” by machine intelligence to model link prediction in complex networks. *Iscience* **26**(1), 105697 (2023).
43. Zhang, W. *et al.* SFPEL-LPI: Sequence-based feature projection ensemble learning for predicting lncRNA-protein interactions. *PLoS Comput. Biol.* **14**(12), e1006616 (2018).
44. Ma, Y. & Ma, Y. Hypergraph-based logistic matrix factorization for metabolite–disease interaction prediction. *Bioinformatics* **38**(2), 435–443 (2022).
45. Ma, Y., He, T. & Jiang, X. Projection-based neighborhood non-negative matrix factorization for lncRNA-protein interaction prediction. *Front. Genet.* **10**, 1148 (2019).
46. Ma, Y. DeepMNE: Deep multi-network embedding for lncRNA-disease association prediction. *IEEE J. Biomed. Health Inform.* **26**(7), 3539–3549 (2022).
47. Bleakley, K. & Yamanishi, Y. Supervised prediction of drug–target interactions using bipartite local models. *Bioinformatics* **25**(18), 2397–2403 (2009).
48. Zhang, Z. C. *et al.* A graph regularized generalized matrix factorization model for predicting links in biomedical bipartite networks. *Bioinformatics* **36**(11), 3474–3481 (2020).
49. Durán, C. *et al.* Pioneering topological methods for network-based drug–target prediction by exploiting a brain-network self-organization theory. *Brief. Bioinform.* **19**(6), 1183–1202 (2018).
50. Ru, X., Ye, X., Sakurai, T. & Zou, Q. NerLTR-DTA: Drug–target binding affinity prediction based on neighbor relationship and learning to rank. *Bioinformatics* **38**(7), 1964–1971 (2022).
51. Huang, K. *et al.* DeepPurpose: A deep learning library for drug–target interaction prediction. *Bioinformatics* **36**(22–23), 5545–5547 (2020).
52. Öztürk, H., Özgür, A. & Ozkirimli, E. DeepDTA: Deep drug–target binding affinity prediction. *Bioinformatics* **34**(17), i821–i829 (2018).
53. Nguyen, T. *et al.* GraphDTA: Predicting drug–target binding affinity with graph neural networks. *Bioinformatics* **37**(8), 1140–1147 (2021).
54. Davis, M. I. *et al.* Comprehensive analysis of kinase inhibitor selectivity. *Nat. Biotechnol.* **29**(11), 1046–1051 (2011).
55. Tang, J. *et al.* Making sense of large-scale kinase inhibitor bioactivity data sets: A comparative and integrative analysis. *J. Chem. Inf. Model.* **54**(3), 735–743 (2014).
56. Pahikkala, T. *et al.* Toward more realistic drug–target interaction predictions. *Brief. Bioinform.* **16**(2), 325–337 (2015).
57. Papadopoulos, F., Aldecoa, R. & Krioukov, D. Network geometry inference using common neighbors. *Phys. Rev. E* **92**(2), 022807 (2015).
58. Wang, X., Zhang, Y. & Shi, C. Hyperbolic heterogeneous information network embedding. In *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 5337–5344 (2019).
59. Chamberlain, B. P., Hardwick, S. R., Wardrope, D. R., Dzogang, F., Daolio, F. & Vargas, S. (2019). Scalable hyperbolic recommender systems. *arXiv preprint arXiv:1902.08648*.
60. Wang, L., Gao, C., Huang, C., Liu, R., Ma, W. & Vosoughi, S. Embedding heterogeneous networks into hyperbolic space without meta-path. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 10147–10155, (2021).
61. Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A. & Boguná, M. Hyperbolic geometry of complex networks. *Phys. Rev. E* **82**(3), 036106 (2010).
62. Kitsak, M., Papadopoulos, F. & Krioukov, D. Latent geometry of bipartite networks. *Phys. Rev. E* **95**(3), 032309 (2017).

Author contributions

A.P. conceived and designed the method, implemented and tested the algorithms, analyzed the data and wrote the paper.

Competing interests

The author declares no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-023-27995-5>.

Correspondence and requests for materials should be addressed to A.P.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023