HLoOP - Hyperbolic 2-space Local Outlier Probabilities

Clémence Allietta, Jean-Philippe Condomines, Jean-Yves Tourneret, Emmanuel Lochin

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

HLoOP – Hyperbolic 2-space Local Outlier Probabilities

CLÉMENCE ALLIETTA[1], JEAN-PHILIPPE CONDOMINES[1], JEAN-YVES TOURNERET[2], EMMANUEL LOCHIN[1],

[1]Fédération ENAC ISAE-SUPAERO ONERA, Université de Toulouse, France
[2]IRIT/ENSEEIHT/Tésa, université de Toulouse, France

Corresponding author: Emmanuel Lochin (e-mail: emmanuel.lochin@enac.fr).

ABSTRACT Hyperbolic geometry has recently garnered considerable attention in machine learning due to its capacity to embed hierarchical graph structures with low distortions for further downstream processing. This paper introduces a simple framework to detect local outliers for datasets grounded in hyperbolic 2-space referred to as Hyperbolic Local Outlier Probability (HLoOP). Within a Euclidean space, well-known techniques for local outlier detection are based on the Local Outlier Factor (LOF) and its variant, the LoOP (Local Outlier Probability), which incorporates probabilistic concepts to model the outlier level of a data vector. The developed HLoOP combines the idea of finding nearest neighbors, density-based outlier scoring with a probabilistic, statistically oriented approach. Therefore, the method consists in computing the Riemmanian distance of a data point to its nearest neighbors following a Gaussian probability density function expressed in a hyperbolic space. This is achieved by defining a Gaussian cumulative distribution in this space. The HLoOP algorithm is tested on the WordNet dataset yielding promising results. The code and data will be made available on request for reproducibility.

INDEX TERMS Outlier Detection, Hyperbolic Embedding, LoOP, HLoOP

I. INTRODUCTION AND PRIOR WORK

From social interaction analysis in social sciences to sensor networks in communication, machine learning has gained in importance in the last few years for analyzing large and complex datasets. Applying machine learning algorithms in an Euclidean space is efficient when data have an underlying Euclidean structure. However, in many applications such as computer graphics or computer vision, data cannot be embedded in a Euclidean space, which prevents the use of conventional algorithms [1]. As an example, in datasets having a hierarchical structure, the number of relevant features can grow exponentially with the depth of the hierarchy; thus, these features cannot be embedded without distortions in an Euclidean space. In the quest for a more appropriate geometry of hierarchies, hyperbolic spaces and their models (Poincaré disk or upper-half plane conformal models, Klein non-conformal model, Beltrami hemisphere model and Lorentz hyperboloid model among others [2]) provide attractive properties that can lead to substantial performance and efficiency benefits for learning representations of hierarchical and graph data. Among several potential advantages, we can highlight [3] 1) a better generalization capability of the model, with less overfitting, computational complexity, and requirement of training data; 2) a reduction in the number of model parameters and embedding dimensions; 3) a better model understanding and interpretation. Empowered by these geometric properties, hierarchical embeddings have recently been investigated for complex trees with low distortions [4], [5], [6], [7]. This has led to rapid advances in machine learning and data science across many disciplines and research areas, including but not limited to graph networks [8], [9], [10], [11], computer vision [12], [13], [14], [15], [16], network topology analysis [17], [18], [19], [20], quantum science [21], [22]. Finally, it is interesting to mention the recent boom in hyperbolic neural networks and hyperbolic computer vision, which has been reported in recent reviews [23], [3].

Motivated by these recent advances, identifying and dealing with outliers is crucial for generating trustworthy insights and making data-driven decisions in hyperbolic spaces, e.g., providing information about which nodes are highly connected (and hence more central) or which nodes correspond to outliers such that embedding methods can realistically be used to model real complex patterns. In this study, we focus on *local* outlier detection, which describes *local* properties of data, which is relevant in many applications involving Euclidean spaces. An overview of *local* anomaly detection

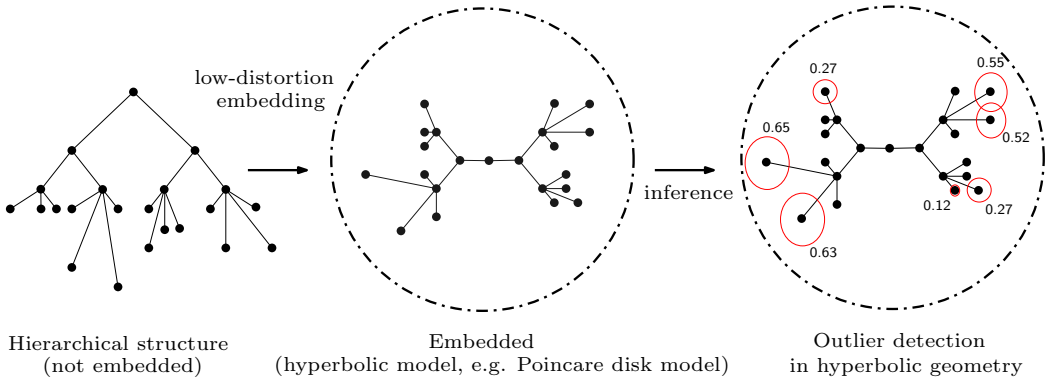


FIGURE 1: Illustration inspired from [24] of local outlier probabilities in a hyperbolic model: A hierarchical structure (left) is embedded in a hyperbolic space with low-distortion (middle). Note the middle figure illustrates that the metrics used in the Euclidean space and a hyperbolic space are different, which results in a modification of the distance between each point. The right figure illustrates the way the local outlier probabilities can be displayed in this hyperbolic space.

methods can be found in the literature from many surveys or books [25], [26], [27], [28]. Initially, research related to local outlier detection was focused on intrusion detection [29], [30], [31], fraud detection [32], [33], [34], [35] and medical applications [36], [37]. Intrusion detection consists of detecting abnormal traffic in networks due to suspicious data or violations of network management policies. Fraud detection detects unexpected activities in banking or insurance data, such as fraudulent online payments by credit card or inconsistent insurance claims. In the wake of other disciplines, local outlier detection algorithms have been used for medical data, e.g., to detect abnormal QRS complexes in electrocardiograms due to certain diseases (such as premature ventricular contraction).

A well-known technique for local outlier detection is the Local Outlier Factor (LOF) [38][39] and its variant the LoOP (Local Outlier Probability)[40] with probabilistic concepts allowing the outlier level of a data point to be defined. The properties of these methods make the detection of historical data attractive, particularly because they provide local outlier scores based on the degree of isolation of each vector from the neighborhood. While the LOF detects outlier data points using the score of an outlier, the LoOP detects them by providing for each data point p an outlier score (belonging to the interval $]0, 1[$) corresponding to the probability that p is an anomaly. Because the distances have positive values, the LoOP algorithm assumes a *half-Gaussian* distribution for these distances. Based on Bayesian inference, the outlier score is directly interpretable as an outlier probability.

Probabilistic inference for data embedding in hyperbolic spaces is a young research area, in which the first main contributions can be dated from the beginning of 2020 (see [24], [41], [42], [43] and the references therein). These insights led, for instance, to define the so-called Souriau Gibbs in the Poincaré disk with its Fisher information metric coinciding with the Poincaré Riemannian metric [44]. A novel parametrization for the density of Gaussian on hyperbolic spaces is presented in [41]. This density can be analytically calculated and differentiated using a simple random variate generation algorithm. An alternative is to use a simple Gaussian distribution in hyperbolic spaces, e.g., [45], [46] introduced Riemannian normal distributions for the *univariate normal model*, with an application to the classification of univariate normal populations. Along with the wrapped normal generalization used in [41], [47] studies a thorough treatment of the maximum entropy normal generalisation. Meanwhile, many applications combining hyperbolic geometry and Variational Auto-Encoders (VAEs) were investigated in [48], [47], [49], [43] based on the fact that VAE latent space components embedded in hyperbolic space help to represent and discover hierarchies. This work introduces an original framework to detect local outliers for datasets grounded in hyperbolic 2-space referred to as HLoOP (Hyperbolic Local Outlier Probability). To the best of our knowledge, this is the first existing algorithm allowing to detect outliers belonging to data embedded within an hyperbolic space.

II. CONTRIBUTIONS AND PAPER ORGANIZATION

The key contributions of this paper are:

(1) We extend the *Local Outlier Probabilities* (LoOP) algorithm to make it applicable to *hyperbolic models*, e.g, the Poincaré disk model, leading to Hyberbolic 2-space Local Outlier probabilities (HLoOP). Figure 1 illustrates the pipeline to obtain *local* outlier probability distributions in hyperbolic geometry from hierarchical structures.

(2) We derive an expression of a *Gaussian cumulative distribution in hyperbolic spaces* which ensures that the Probabilistic Local Outlier Factor (PLOF) is performed by fully exploiting the information geometry of the observed data.

The rest of this paper is structured as follows: section IV briefly outlines some concepts from Riemannian geometry for univariate models. In Section V we introduce local outlier probability detection in hyperbolic spaces and discuss how this probability can be computed. Section VI evaluates the

proposed approach on the benchmark dataset "taxonomy embedding from WordNet".

III. MOTIVATION OF THIS WORK

The first question that may arise when reading this article is *Why perform outlier detection in hyperbolic space rather than Euclidean space?*

Simply because certain data are better analyzed within a hyperbolic space (e.g., Google maps [50]) and often, there is no rationale to convert this representation into Euclidian space. For instance and without loss of generality, many recent applications in computer networking are represented in hyperbolic space, including :

- **network topology :** hyperbolic geometry can be used to model complex networks with a hierarchical structure, such as the internet or social networks. By representing these networks as hyperbolic spaces, it is possible to capture their underlying geometry and study their properties and dynamics. In [51], the authors illustrate how heterogeneous degree distributions and strong clustering naturally arise from the negative curvature of the underlying hyperbolic geometry. The authors show that if a network has a metric structure and a heterogeneous degree distribution, then it has an effective hyperbolic geometry. This allows the authors to establish a mapping between the geometric framework and statistical mechanics of complex networks;
- **routing algorithms :** hyperbolic geometry can also be used to design efficient routing algorithms for large-scale networks. In hyperbolic space, the distance between two points grows exponentially as they move away from each other, which can be exploited to design routing algorithms that minimize the number of hops needed to transmit data between two nodes. In [17], the author proposes a reliable routing algorithm for wireless networks and sensor-nets, able to assign virtual coordinates to each node in the hyperbolic plane, allowing for successful and consistent routing of packets to a destination point. Similarly, in [18], the authors designed an algorithm for online greedy graph embedding in the context of dynamic multihop communication networks. Several other proposals exist in the context of overlay networks [19] or satellites networking [20];
- **network visualization :** hyperbolic geometry can be used to visualize complex networks in two or three dimensions, allowing researchers to explore the structure and properties of these networks in a more intuitive way. This can be especially useful for large-scale networks that are difficult to visualize using traditional methods. As shown in [50], the authors demonstrate that Google maps on a cell phone is an example of hyperbolic geometry;
- **distributed systems :** finally and as shown in [52], [53], hyperbolic geometry can be used to design distributed systems that are more fault-tolerant and scalable than traditional systems. By representing the system as a hyperbolic space, it is possible to distribute the load across the network in a more efficient manner, thereby reducing the risk of overload or failure.

These examples motivate the present study, which aims to introduce a well-known and efficient outlier detection tool : LoOP [40]), to the hyperbolic space.

IV. A UNIVARIATE NORMAL MODEL FOR HYPERBOLIC SPACES

This section briefly reminds some concepts of Riemannian geometry [46], [54], [47] for the univariate normal model, which are necessary to formally extend the LoOP detection algorithm. Note that similarly to [45], the main assumptions is that the data used, must belong to a hyperbolic space with a well-defined hyperbolic geometry.

A. RIEMANNIAN GEOMETRY AND RAO DISTANCE

A Riemannian manifold is a real and smooth manifold denoted as $\mathcal{M}$ equipped with a positive definite quadratic form $g_x : \mathcal{T}_x\mathcal{M} \times \mathcal{T}_x\mathcal{M} \mapsto \mathbb{R}$ at each point $x \in \mathcal{M}$, where $\mathcal{T}_x\mathcal{M}$ is the tangent space defined at the local coordinates $x = (x_1, ..., x_n)^T$. Intuitively, it contains all the possible directions in which one can tangentially pass through x. A norm is induced by the inner product on $\mathcal{T}_x\mathcal{M} : \|\cdot\|_x = \sqrt{<\cdot,\cdot>_x}$. An infinitesimal volume element is induced on each tangent space $\mathcal{T}_x\mathcal{M}$. The quadratic form g_x is called a Riemannian metric and allows us to define the geometric properties of spaces, such as the angles and lengths of a curve. The Riemannian metric g_x is an n-by-n positive definite matrix such that an infinitesimal element of length ds^2 is defined as:

$$ds^2 = \begin{pmatrix} dx_1 & \cdots & dx_n \end{pmatrix} g_x \begin{pmatrix} dx_1 \\ \vdots \\ dx_n \end{pmatrix}. \tag{1}$$

The Riemannian metric is a well-known object in differential geometry. For instance, the Poincaré disk with a unitary constant negative curvature corresponds to the Riemannian manifold in the hyperbolic space $(\mathbb{H}, g_x^{\mathbb{H}})$, where $\mathbb{H} = \{x \in \mathbb{R}^n : \|x\| < 1\}$ is the open unit disk[1]. Its metric tensor can be written from the Euclidean metric $g^E = I_n$ and the Riemannian metric such that $g_x^{\mathbb{H}} = \lambda_x^2 g^E$, where $\lambda_x = \frac{2}{1-\|x\|^2}$ is the conformal factor. The Rao distance between two points $z_1 = (x_1, y_1)^T$ and $z_2 = (x_2, y_2)^T$ in $\mathbb{H}$ is given as:

$$d_H(z_1, z_2) = \operatorname{arcosh}\left[1 + 2\frac{||z_1 - z_2||^2}{(1 - ||z_1||^2)(1 - ||z_2||^2)}\right], \tag{2}$$

where arcosh denotes the inverse hyperbolic cosine and $\|\cdot\|$ is the usual Euclidean norm. Unlike the Euclidean distance, the hyperbolic distance grows exponentially fast as we move the points toward the boundary of the open unit disk. There exists many models of hyperbolic geometry including the Klein non-conformal model, the Beltrami hemisphere model and the Lorentz hyperboloid model among others. One model

[1]A d-dimensional hyperbolic space, denoted $\mathbb{H}^d$, is a complete, simply connected, d-dimensional Riemannian manifold with constant negative curvature c.

of hyperbolic geometry can be transformed into another one by using a one-to-one mapping, which yields an isometric embedding [55].

B. RIEMANNIAN PRIOR ON THE UNIVARIATE NORMAL MODEL

A Gaussian distribution in $\mathbb{H}$, denoted as $\mathcal{N}_H(\mu,\sigma)$, depends on two parameters, the Fréchet mean $\mu \in \mathbb{H}$ (i.e., the center of mass) and the dispersion parameter $\sigma > 0$, similar to the Gaussian density in the Euclidean space. The Gaussian probability density function (pdf) in the hyperbolic space, denoted as $p_H(x|\mu,\sigma)$ is defined as follows [45]:

$$p_H(x|\mu,\sigma) = \frac{1}{Z(\sigma)} \exp\left[-\frac{d_H^2(x,\mu)}{2\sigma^2}\right]. \tag{3}$$

Several remarks can be made from (3): (i.) the main difference between the hyperbolic density $p_H(\cdot)$ and the Gaussian density in the Euclidean space is the use of the squared distance $d_H^2(x,\mu)$ in the exponential (referred to as Rao distance) and a different dispersion-dependent normalization constant $Z(\sigma)$ which reduces to $\sqrt{2\pi\sigma^2}$ in the Euclidean case. Note that the constant $Z(\sigma)$ is linked to the underlying geometry of the hyperbolic space (ii.). To define a Gaussian distribution $\mathcal{N}_H(\mu,\sigma)$, through its pdf, it is necessary to have an exact expression of the normalizing constant $Z(\sigma)$. This constant can be determined using hyperbolic polar coordinates $r = d_H(x,\mu)$ (i.e, a pulling-back) to calculate $Z(\sigma)$ using an integral depending on the Riemannian volume element include a reference here (iii.). By introducing the parametrization $z = (x,y)^T$ where $x = \mu/\sqrt{2}$, $y = \sigma$ and the Riemannian metric for the univariate Gaussian model $ds^2(z) = (dx^2 + dy^2)/y^2$, since $\mathbb{H}$ is of dimension 2, the Riemannian area is $dA(z) = dxdy/y^2$ or $dA(z) = \sinh(r)drd\varphi$ in polar coordinates. For a two-dimensional parameter space, the normalization constant $Z(\sigma)$ was computed in [45]:

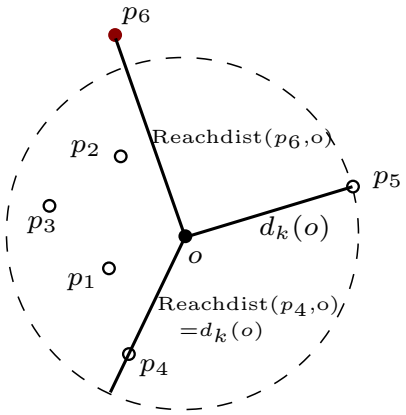
$$\begin{aligned} Z(\sigma) &= \int_{\mathbb{H}} \exp\Big(-\frac{r^2}{2\sigma^2}\Big) dA(z) \\ &= 2\pi\sigma\sqrt{\frac{\pi}{2}} \exp\left(\frac{\sigma^2}{2}\right) \operatorname{erf}\left(\frac{\sigma}{\sqrt{2}}\right), \end{aligned} \tag{4}$$

where erf is the error function. This expression of $Z(\sigma)$ completes the definition of the Gaussian distribution $\mathcal{N}_H(\mu,\sigma)$. The authors of [47] have shown that when σ get smaller (resp. bigger), the Riemannian normal pdf is closer (resp. further) to the wrapped normal pdf [47].

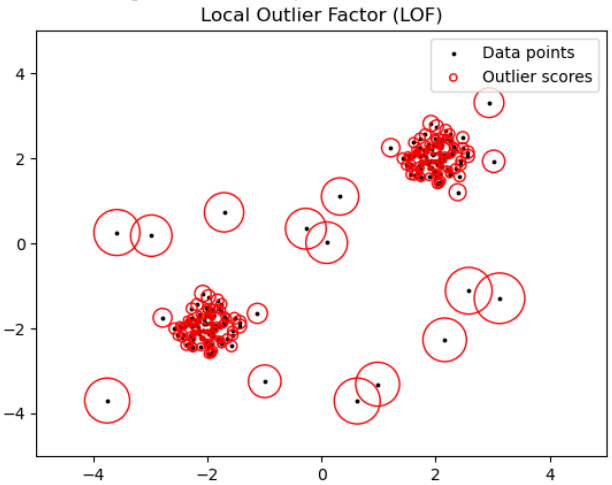
V. HYPERBOLIC 2-SPACE LOCAL OUTLIER PROBABILITY

A. DENSITY-BASED OUTLIER SCORE USING A PROBABILISTIC APPROACH

This subsection briefly presents the main theoretical principles of some research studies dealing with local outlier probability concepts. A local outlier is a data point that is different or far from most elements of the entire dataset compared to its local neighborhood, which is measured by the k-Nearest Neighbors (kNN) algorithm [56]. Therefore, local outlier detection covers a small subset of data points at a given time (Figure 2a). To



(a) Example of reachability distance.



(b) Example of LOF algorithm.

FIGURE 2: (a) Examples of reachability distances for different data points p with respect to o, when $k = 5$. (b) Examples of detected anomalies using local density deviations of a given dataset, where the white and red points indicate the normal points and the anomalies.

compute the degree of outlier of a point p in a dataset $\mathcal{D}$, several distances must be introduced [57]. The k-distance of a point $p \in \mathcal{D}$ denoted as $d_k(p)$ is the distance between $p \in \mathcal{D}$ and its k-nearest neighbor. The notion of k-distance must be used to delimit a neighborhood that contains the k-nearest neighborhood of p. This neighborhood denoted as $N_k(p)$ is defined as $N_k(p) = \{q \in \mathcal{D}\backslash\{p\} | d(q,p) \leq d_k(p)\}$. The reachability distance denoted as $\text{reachdist}_k(p,o)$ of a point $p \in \mathcal{D}$ with respect to a point o is defined as $\text{reachdist}_k(p,o) = \max\{d_k(p), d(p,o)\}$. Based on these definitions, the LOF (Local Outlier Factor) algorithm has been introduced in [38] to improve the kNN approach in the scenario where, e.g., in a two-dimensional dataset, the density of one cluster is significantly higher (resp. lower) than another cluster. To do this, it calculates the local reachable density of

the data, and calculates the local outlier factor score according to the local reachable density. Figure 2b illustrates the local density-based outlier scoring for a given dataset. LOF values are converted into circle radius so that it is possible to set a threshold (e.g., 0.1) to detect outlier samples. In this work, we introduce the HLOF (Hyperbolic Local Outlier Factor) algorithm by replacing d(q, p) with the Rao distance (2).

While the LOF (Local Outlier Factor) algorithm detects outlier data points using the reachability distance, the Local Outlier Probability (LoOP) algorithm introduces the probabilistic distance of $o \in \mathcal{D}$ to a context set $\mathcal{S} \subseteq \mathcal{D}$, referred to as $\mathrm{pdist}(o, \mathcal{S})$ with the following property:

$$\forall s \in \mathcal{S} : \mathcal{P}[d(o,s) \leq \mathrm{pdist}(o,\mathcal{S})] \geq \varphi. \tag{5}$$

This probabilistic distance corresponds to the radius of a disk that contains a data point of $\mathcal{S}$, obtained from the kNN algorithm, with a certain probability, denoted as φ. The reciprocal of the probabilistic distance can be considered as an estimation of the density of $\mathcal{S}$, i.e, $\mathrm{pden}(\mathcal{S}) = \frac{1}{\mathrm{pdist}(o,\mathcal{S})}$. Assuming that o is the center of $\mathcal{S}$ and the local density is approximately a *half-Gaussian* distribution, the probabilistic set distance of o to $\mathcal{S}$ can be defined as

$$\mathrm{pdist}(o,\mathcal{S}) = \lambda\sigma(o,\mathcal{S}), \tag{6}$$

where $\sigma(o,\mathcal{S}) = (\sum_{s\in\mathcal{S}} d^2(o,s)/\,|\mathcal{S}|)^{1/2}$ is the standard Euclidean distance of o in $\mathcal{S}$. The parameter λ is linked to the selectivity of detection through the *quantile function* of the normal distribution via the relation $\lambda = \sqrt{2}\mathrm{erfinv}(\varphi)$, where erfinv is the inverse error function.

To be detected as an anomaly for the set $\mathcal{S}$, a data point should deviate from the center of $\mathcal{S}$ for more than λ times the standard distance. For instance, $\lambda = 3$ means that a circle of radius $\mathrm{pdist}(o,\mathcal{S})$ and center o contains any data point of $\mathcal{S}$ with a probability $\varphi \approx 99.7\%$. The resulting probability is the Local Outlier Probability (LoOP) given by

$$\mathrm{LoOP}_{\mathcal{S}}(o) = \max\left\{0, \mathrm{erf}\left(\frac{\mathrm{PLOF}_{\lambda,\mathcal{S}}(o)}{\mathrm{nPLOF}\sqrt{2}}\right)\right\}, \tag{7}$$

where the Probabilisitic Local Outlier Factor (PLOF) is defined as $\mathrm{PLOF}_{\lambda,\mathcal{S}}(o) = (\mathrm{pdist}(\lambda,o,\mathcal{S}))/(\mathrm{E}_{s\in\mathcal{S}}\left[\mathrm{pdist}(\lambda,s,\mathcal{S})\right]) - 1$ and a normalization factor nPLOF is such that $\mathrm{nPLOF} = \lambda(\mathrm{E}[\mathrm{PLOF}^2])^{1/2}$. The LoOP value is directly interpretable as the probability of o being an outlier, i.e, close to 0 for points within dense regions and close to 1 for density-based outliers.

B. HLOOP ALGORITHM

This subsection presents the main contribution of this study, which is an adaptation of the LoOP algorithm to data lying in a hyperbolic 2-space. As mentioned above, the LoOP algorithm in an Euclidean space exploits a probabilistic set distance, called $\mathrm{pdist}(o,\mathcal{S})$ (see 6), to pick the density around o in the context set $\mathcal{S}$ with a probability of φ. To define a local outlier probability adapted to hyperbolic geometry, it is necessary to calculate a new parameter λ_H, which ensures that $\mathrm{pdist}(o,\mathcal{S})$ is performed considering the hyperbolic geometry. To come up with such a solution, the key idea is to derive a new *quantile function* through an expression of a Gaussian cumulative distribution function (cdf) that can be obtained by integrating the pdf (3) in $\mathbb{H}$. Using polar coordinates (see subsection IV-B, remark iii.), it is possible to calculate this Gaussian cdf explicitly. To find the parameter λ_H, we consider the probabilistic distance of $o \in \mathcal{D}$ to a context set $\mathcal{S} \subseteq \mathcal{D}$ using a Riemannian distance $d_H(o,s)$ and the following statistical property:

$$\forall s \in \mathcal{S} : \varphi = \mathcal{P}[0 < d_H(o,s) \leq \lambda_H\sigma_r] = \mathcal{G}_H(\lambda_H\sigma_r), \tag{8}$$

where $\mathcal{G}_H$ is the cdf of $d_H(o,s)$. Assuming that o is the center of $\mathcal{S}$ and the set of distances of $s \in \mathcal{S}$ to o is approximately *half-Gaussian* in a hyperbolic space, one can compute the standard deviation σ_r using the Riemannian distance $r = d_H(o,s)$ with a mean $d_H(o,o) = 0$. Note that the standard deviation of r denoted as σ_r and its pdf can be determined from the function $\mathcal{G}_H$, e.g., $p_H(r,\sigma_r) = \mathcal{G}'_H(r,\sigma_r)$. Theorem 1 presents the main result of this paper.

Theorem 1. *Given $r \in \mathbb{H}$, $\sigma_r > 0$, the Riemannian geometry of the Gaussian cumulative model associated with the distribution defined in* (3) *is given by*

$$\begin{aligned}\mathcal{G}_H(r,\sigma_r) &= \frac{\pi\sqrt{2\pi}\sigma_r e^{\frac{\sigma_r^2}{2}}}{2Z(\sigma_r)} \\ &\times \left[2\mathrm{erf}\Big(\frac{\sigma_r}{\sqrt{2}}\Big) + \mathrm{erf}\Big(\frac{r-\sigma_r^2}{\sigma_r\sqrt{2}}\Big) - \mathrm{erf}\Big(\frac{r+\sigma_r^2}{\sigma_r\sqrt{2}}\Big)\right].\end{aligned} \tag{9}$$

Proof. Let $\mathcal{P}[0 < r \leq R]$ and $dA(z) = \sinh(r)drd\varphi$ such that:

$$\mathcal{G}_H(R) = \int_0^{2\pi}\int_0^R \frac{1}{Z(\sigma_r)} \exp\Big(-\frac{r^2}{2\sigma_r^2}\Big)\sinh(r)drd\varphi.$$

The pdf $p_H(\cdot)$ satisfies the following condition:

$$\int_{\mathbb{H}} p_H(x|\mu,\sigma)d(\mu,\sigma) = 1, \tag{10}$$

where $d(\mu,\sigma)$ is the Lebesgue measure. The cumulative distribution function of the univariate Gaussian distribution of pdf $p_H(\cdot)$ can be computed using (10) as follows:

$$\begin{aligned}\mathcal{G}_H(R) &= \frac{2\pi}{Z(\sigma_r)} \\ &\times \int_0^R \frac{e^{\frac{\sigma_r^2}{2}}}{2}\left(e^{-\frac{(r-\sigma_r^2)^2}{2\sigma_r^2}} - e^{-\frac{(r+\sigma_r^2)^2}{2\sigma_r^2}}\right)dr, \\ &= \frac{\pi\sqrt{2\pi}\sigma_r e^{\frac{\sigma_r^2}{2}}}{2Z(\sigma_r)} \\ &\times \left(\frac{2}{\sqrt{\pi}}\int_{\frac{-\sigma_r}{\sqrt{2}}}^{\frac{R-\sigma_r^2}{\sqrt{2}\sigma_r}} e^{-u_1^2}du_1 - \frac{2}{\sqrt{\pi}}\int_{\frac{\sigma_r}{\sqrt{2}}}^{\frac{R+\sigma_r^2}{\sqrt{2}\sigma_r}} e^{-u_2^2}du_2\right), \\ &= \frac{\pi\sqrt{2\pi}\sigma_r e^{\frac{\sigma_r^2}{2}}}{2Z(\sigma_r)} \\ &\times \left(2\mathrm{erf}\Big(\frac{\sigma_r}{\sqrt{2}}\Big) + \mathrm{erf}\Big(\frac{R-\sigma_r^2}{\sigma_r\sqrt{2}}\Big) - \mathrm{erf}\Big(\frac{R+\sigma_r^2}{\sigma_r\sqrt{2}}\Big)\right).\end{aligned} \tag{11}$$

Taking the limit $\mathcal{G}_H(R) \underset{R\to 1}{\longrightarrow} 1$ in (11) yields:

$$Z(\sigma_r) = (2\pi\sigma_r)\sqrt{\frac{\pi}{2}}\exp\left(\frac{\sigma_r^2}{2}\right)\operatorname{erf}\left(\frac{\sigma_r}{\sqrt{2}}\right),$$

which allows the expression given in [45] to be recovered and completes the proof. □

Combining all these results, the parameter $\lambda_H(\sigma_r)$ is determined by the inverse of $\mathcal{G}_H(\lambda_H\sigma_r)$ (see 8) such that

$$\lambda_H(\sigma_r) = \frac{1}{\sigma_r}\mathcal{G}_H^{-1}(\varphi). \tag{12}$$

It is interesting to note that, while the traditional *quantile function* is independent of the standard deviation, we have obtained means to directly derive the parameter λ_H that exploits the underlying geometry of the hyperbolic space (see subsection IV-B, remark ii.). The HLoOP algorithm is summarized in Algorithm 1.

Algorithm 1 HLoOP algorithm

Input: Dataset $\mathcal{X} = \{x^i\}_{i=1}^m$ where $x^i = (x_1^i, \cdots, x_n^i) \in \mathbb{R}^n$. Pre-determined threshold φ, parameter k, and hyperbolic distance $d_H(p,q)$;

(1) **Determine** the context set $\mathcal{S}$ of the data point x^i from the kNN algorithm;
(2) **Compute** the standard distance σ_r of the context set $\mathcal{S}$;
(3) **Determine** $\mathcal{G}_H^{-1}(\varphi)$ to derive the parameter λ_H by (12);
(4) **Calculate** the probabilistic set distance $\mathrm{pdist}_k(x^i)$ of the data point x^i by (6);
(5) **Compute** the local outlier probability $\mathrm{LoOP}_k(x^i)$ of the data point x^i by (7);

Output: Anomaly scores for the elements of the dataset $\mathcal{X}$.

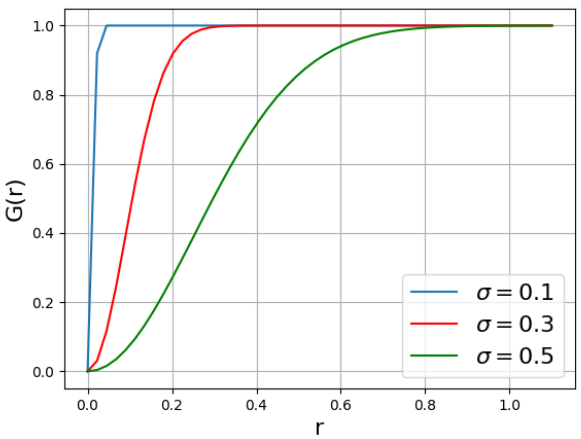
It is worth noting that the main difference between the LoOP (computational complexity of $O(n*\log(n))$ and HLoOP algorithms is the addition of a Newton algorithm to estimate the threshold given by $\lambda = G_H^{-1}$. Knowing that the time complexity of Newton's method is $O(\log(1/\epsilon))$, where ϵ is the desired precision of the root; we can assess that the complexity of the HLoOP algorithm is equivalent to that of the LoOP algorithm. Similarly, the same applies to HLOF, where LOF is the computational complexity of $O(n^2)$.

C. IMPLEMENTATION DETAILS

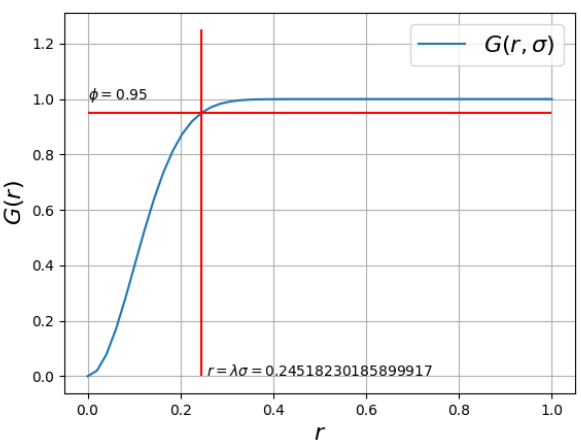
Before presenting the performance of the algorithm described above, it is interesting to discuss some aspects related to the implementation of the HLoOP algorithm. Most of the steps used in the implementation of the HLoOP algorithm are directly related to the Euclidean LoOP, except that the distances are no longer Euclidean but hyperbolic. However, the computation of the significance λ cannot be computed as in the Euclidean LoOP. While an analytic expression of the Gaussian quantile function is known in Euclidean space, the derivation of the cumulative distribution in the Poincare disk, illustrated in Figure 3a, does not lead to an analytic formulation of its inverse $\mathcal{G}_H^{-1}$. Actually, an analytic expression of $\mathcal{G}_H^{-1}$ is not needed to compute λ providing the value of $r = \lambda\sigma$ for which $\mathcal{G}_H(r,\sigma) = \varphi$ can be determined. Solving

$$\mathcal{G}_H(r,\sigma) = \varphi,$$

for a given pair (σ,φ) can be performed using the Newton method. Once we have obtained r, as shown in Figure 3b, the significance can be determined using the relation $r = \lambda\sigma$, yielding $\lambda = r/\sigma$. All the elements required to implement the HLoOP algorithm are now available. The next section is dedicated to the assessment of the performance of this algorithm.



(a) Univariate cdf associated with the density (3).



(b) Newton's method to determine $r = \mathcal{G}^{-1}(\varphi)$.

FIGURE 3: Cumulative distribution $\mathcal{G}_H$ in the Poincare disk and resolution of $\mathcal{G}_H(r,\sigma) = \varphi$.

VI. RESULTS

A. PERFORMANCE OF HLOOP ON A TOY DATASET

The HLoOP method is first used to detect outliers in a toy dataset, with a reduced number of points. The dataset was gen-

erated as follows: first, some vectors were generated uniformly in two circular areas located in the Poincaré Disk (clusters **A** and **B**). Then, each area was filled with 40 points whose positions are pulled from the normal distribution $\mathcal{N}(\cdot, RI_2)$ where I_2 is the 2×2 identity matrix. Five points located outside these areas (cluster **C**) constitute the outliers of the toy dataset, which is finally composed of $2 \times 40 + 5 = 85$ points. The HLoOP algorithm was applied to this dataset, and its performance was compared to that of HLOF. As a first test, we compute the HLOF and HLoOP values of each point of the embedding for $k = 15$ and a threshold $\varphi = 95\%$ for HLoOP. Figures 4 and 5 show the different points that are surrounded by a circle whose radius is proportional to the HLoOP or HLOF values. We observe that for both methods (HLoOP or HLOF), the outliers (cluster **C**) have a higher score than the inliers (clusters **A** and **B**). For the HLoOP, this corresponds to the probability of a point being an outlier, whereas for the HLOF, the interpretation of the score is less straightforward. It is also interesting to note that cluster **A** highlights a weakness of HLOF, which is designed for clusters of uniform density as LOF. It is interesting to mention here that HLOF assigns relatively high outlier scores to the points of cluster **A** compared to HLoOP, which shows that HLoOP performs better for this example.

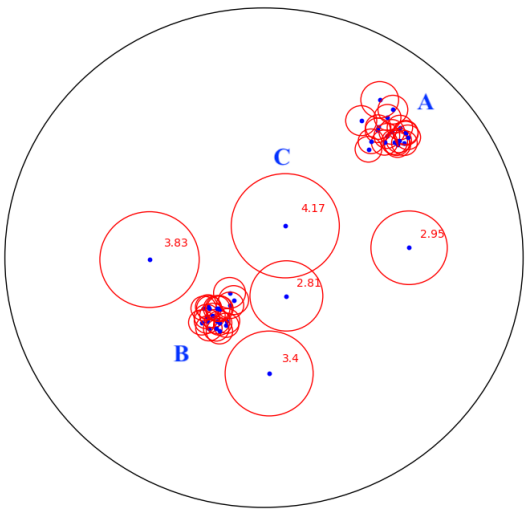


FIGURE 4: Embedding of the toy dataset in the Poincaré disk: HLOF values.

A classical metric used to quantify the quality of the outlier detection is the Area Under the Receiver Operating Curve (AUC-ROC). We recall that a value of AUC-ROC close to 0 corresponds to very poor detection performance (around 0% of the decisions made by the algorithm are correct), while an AUC ROC close to 1 means that the algorithm is making very few errors. As observed in Figure 6, the HLoOP algorithm provides good anomaly detections: for $k > 2$ (number of neighbors considered to evaluate the density of the context set $\mathcal{S}$), the number of true positives (actual outliers detected as outliers) is between 95 and 100 %, which is a very good result. Meanwhile, the performance of HLOF is more contrasted and

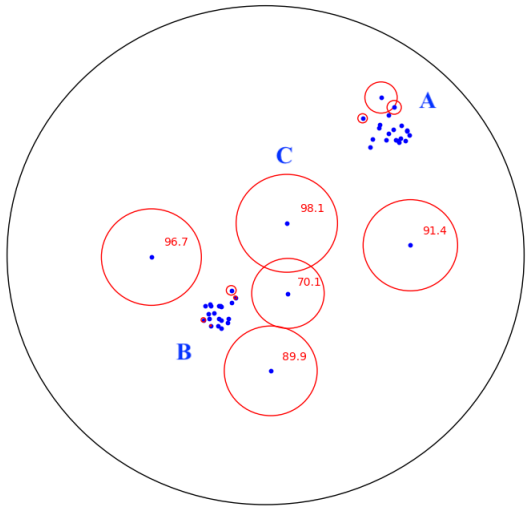


FIGURE 5: Embedding of the toy dataset in the Poincaré disk: HLoOP values. Some points have very small HLoOP values and their associated circles do not appear in the figure.

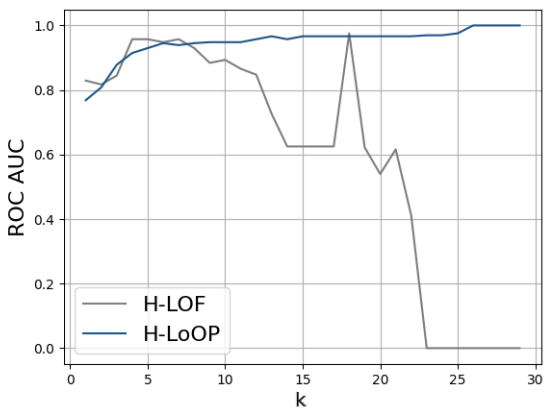


FIGURE 6: AUC ROC - Outlier detection in a toy dataset

strongly dependent on the value of k. In particular, for higher values of k, the HLOF performance dramatically decreases. The next section assesses the performance of HLoOP on a larger dataset containing up to 1000 points.

B. EVALUATING THE PERFORMANCES ON THE WORDNET/MAMMALS SUBGRAPH

This section evaluates the performance of HLoOP on a subgraph of the WORDNET database. WORDNET is a lexical dataset composed of 117000 synsets[2], which correspond to nouns, adjectives, or verbs that are linked by conceptual relations. Several subgraphs are known to exist in this dataset. Among them, we chose to apply HLOF and HLoOP to a group of 1180 synsets from the subgraph ‘‘Mammals’’. The dataset was corrupted by 11 outliers corresponding to nouns of animals that are not mammals (i.e., fishes, reptiles or birds) and was

[2]data elements that are considered semantically equivalent

embedded in the Poincaré Disk using the algorithm of Nickel et al. (2017). The values of HLoOP and HLOF were calculated for the points of this embedding. The Area Under the Receiving Operator Curve (AUC ROC) was finally calculated for both HLOF and HLoOP for several values of k. As shown in Figure 7, the performance of HLoOP is better than HLOF for all values of k, with a ROC AUC larger than 0.98, while HLOF leads to a ROC AUC less than 68%. In addition to its good performance, the HLoOP algorithm yields AUC values that are quite independent of k, which is outstanding.

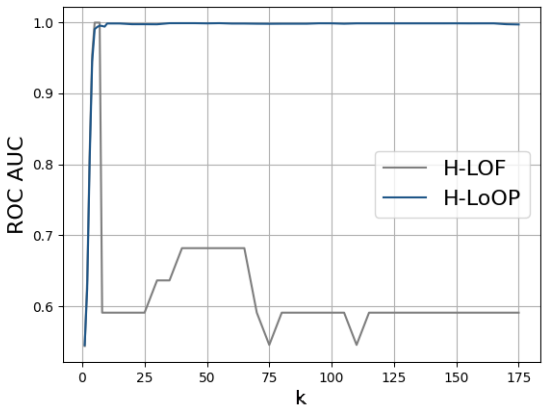


FIGURE 7: AUC ROC - Outlier detection in the corrupted subgraph of Wordnet/Mammals

As shown in Figures 8 and 9 respectively for HLOF and HLoOP, the detected anomalies are surrounded by a circle whose radius is proportional to the HLoOP or HLOF values. The number of neighbors considered in the density computations of HLoOP and HLOF is 20. What is interesting is that for the same number of neighbors considered in the density calculation, the figures clearly show that the performance of HLoOP better discriminates outliers than HLOF. Particularly, we observe that one of the anomalies is not detected by the HLOF, and points that are very far from the group in the center had a fairly low score.

VII. CONCLUSION AND PERSPECTIVES

This paper presents extensions of the Local Outlier Factor (LOF) and Local Outlier Probability (LoOP) algorithms, which are referred to as Hyperbolic LOF (HLOF) and Hyperbolic LoOP (HLoOP). Rather than working in the Euclidean space, these extensions work in a specific model of hyperbolic space, namely the Poincaré Disk. Both algorithms are density based and compare the density of a point with the density of its neighbors. On the one hand, HLOF computes the density based on a deterministic distance (called reachability distance), while HLoOP introduces the notion of probabilistic distance and returns its probability of being an outlier for each point. The simulations conducted on a toy dataset have shown that the HLoOP algorithm allows a better distinction of outliers and inliers than HLOF. While HLoOP directly provides the

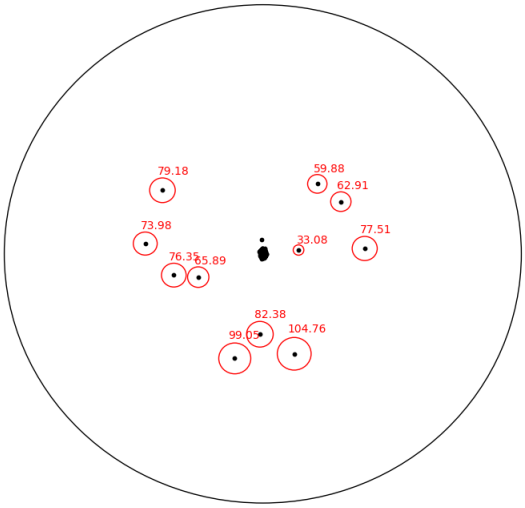


FIGURE 8: Wordnet/Mammals HLOF values.

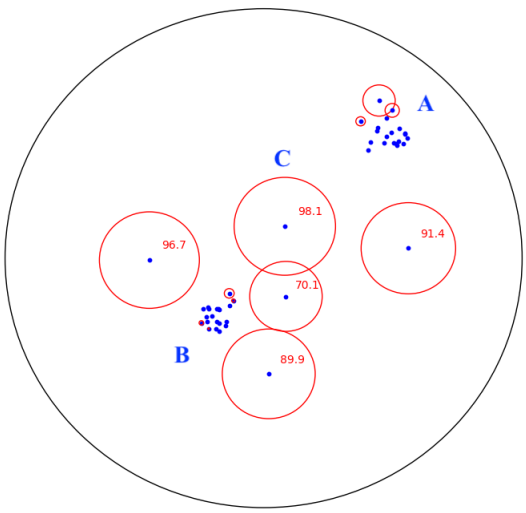


FIGURE 9: Wordnet/Mammals HLoOP values.

probability of each point being an outlier, HLOF returns a score whose interpretation is not straightforward and depends on the dataset under study. Evaluations of the areas under the receiver operating characteristics of data in the Poincaré disk have confirmed that HLoOP has a better detection performance than HLOF. The results obtained with this dataset also show that the HLoOP performance is less sensitive to the number of neighbors considered in the computation of the density, than HLOF. Given these promising results, we have embedded the *mammals* subset of the Wordnet dataset in the Poincaré disk after introducing artificial outliers. The HLOF and HLoOP values and the areas under the receiver operating characteristics of HLOF and HLoOP confirm the results obtained with the previous dataset. At last but not least, HLoOP adopts the same assumptions as in the LoOP algorithm. These assumptions are based on the application of the central limit theorem, which suggests that the distances follow a normal distribution. By

doing so, we prevent our method from being restricted to a particular type of distribution. It is worth noting that other types of distributions can also be considered in the hyperbolic space, as discussed in [24].

Future work includes the extension of HLOF and HLoOP to the Lorentz disk, i.e., to another model of the hyperbolic space. Indeed, it has been shown that the Poincaré disk presents numerical instabilities that are not observed in the Lorentz model. Moreover, it would be interesting to apply the HLoOP and HLOF algorithms to more complex datasets, with more points and attributes. For instance, given the growing interest for hyperbolic geometry in the computer vision domain, it could be worthwhile to try using HLoOP and HLOF to detect outliers in a set of images. Finally, hyperbolic geometry could be used to derive new outlier detection algorithms based on isolation forest or one-class support vector machines.

REFERENCES

[1] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond Euclidean data. *CoRR*, abs/1611.08097, 2016.

[2] James W Anderson. *Hyperbolic geometry*. Springer Science & Business Media, 2006.

[3] Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10023–10044, 2022.

[4] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Proc. Advances in Neural Information Processing Systems*, Long Beach, CA, USA, Dec. 2017.

[5] Rik Sarkar. Low distortion delaunay embedding of trees in hyperbolic plane. In *Proc. Int. Symp. Graph Drawing*, pages 355–366, Eindhoven, NL, Sep. 2011.

[6] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *Proc. Int. Conf. Machine Learning*, pages 4460–4469, Stockholm, Sweden, 10–15 Jul. 2018.

[7] Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *Proc. Int. Conf. Machine Learning*, pages 1646–1655, Stockholm, Sweden, Jul. 2018.

[8] Kiela D Liu Q, Nickel M. Hyperbolic graph neural networks. In *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019.

[9] Ines Chami, Rex Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. In *Proc. Int. Conf. on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019.

[10] Jindou Dai, Yuwei Wu, Zhi Gao, and Yunde Jia. A hyperbolic-to-hyperbolic graph convolutional network. In *Int. Conf. Computer Vision and Pattern Recognition*, June 2021.

[11] Edoardo Cetin, Benjamin Chamberlain, Michael Bronstein, and Jonathan J Hunt. Hyperbolic deep reinforcement learning. *arXiv*, 2022.

[12] Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne van Noord, and Pascal Mettes. Hyperbolic image segmentation. In *Proc. Int. Conf. Comp. Vis. and Pattern Recognition*, pages 4453–4462, New-Orleans, LO, USA, June 2022.

[13] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Curvature generation in curved spaces for few-shot learning. In *Proc. Int. Conf. on Computer Vision*, pages 8691–8700, Montreal, Canada, Oct. 2021.

[14] Vondrick C Suris D, Liu R. Learning the predictability of the future. In *Proc. Int. Conf. Comp. Vis. and Pattern Recognition*, pages 12602–12612, June 2021.

[15] Kong Y Dengxiong X. Generalized open set recognition via hyperbolic side information learning. In *Proc. Winter Conf. on Applications of Computer Vision*, pages 3992–4001, Waikoloa, HI, USA, Jan. 2023.

[16] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proc. Int. Conf. Computer Vision*, pages 9650–9660, Montreal, Canada, Oct. 2021.

[17] R. Kleinberg. Geographic routing using hyperbolic space. In *Proc. Int. Conf. on Computer Communications*, pages 1902–1909, Anchorage, AK, USA, May 2007.

[18] A. Cvetkovski and M. Crovella. Hyperbolic embedding and routing for dynamic graphs. In *Proc. Int. Conf. on Computer Communications*, pages 1647–1655, Rio de Janeiro, Brasil, April 2009.

[19] Cyril Cassagnes, Telesphore Tiendrebeogo, David Bromberg, and Damien Magoni. Overlay addressing and routing system based on hyperbolic geometry. In *Proc. Symposium on Computers and Communications (ISCC)*, pages 294–301, Kerkyra, Corfu, Greece, June 28-July 1 2011.

[20] Sai Lv, Hui Li, Jiangxing Wu, He Bai, Xi Chen, Yufei Shen, Jun Zheng, Rui Ding, Huajun Ma, and Wenjun Li. Routing strategy of integrated satellite-terrestrial network based on hyperbolic geometry. *IEEE Access*, 8:113003–113010, 2020.

[21] Oscar Higgott and Nikolas P. Breuckmann. Subsystem codes with high thresholds by gauge fixing and reduced qubit overhead. *Physical Review X*, 11(3), aug 2021.

[22] Oscar Higgott and Nikolas P. Breuckmann. Constructions and performance of hyperbolic and semi-hyperbolic Floquet codes, 2023.

[23] Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. Hyperbolic deep learning in computer vision: A survey. *arXiv 2305.06611*, 2023.

[24] Frank Nielsen and Kazuki Okamura. Information measures and geometry of the hyperbolic exponential families of Poincaré and hyperboloid distributions. 2022.

[25] Shubin Su, Limin Xiao, Li Ruan, Fei Gu, Shupan Li, Zhaokai Wang, and Rongbin Xu. An efficient density-based local outlier detection approach for scattered data. *IEEE Access*, 7:1006–1020, 2019.

[26] Omar Alghushairy, Raed Alsini, Terence Soule, and Xiaogang Ma. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cogn. Comput.*, 5, 2020.

[27] Imen Souiden, Zaki Brahmi, and Hajer Toumi. A survey on outlier detection in the context of stream mining: Review of existing approaches and recommandations. In *Proc. Int. on Intelligent Systems Design and Applications*, Porto, Portugal, Dec. 2016.

[28] Guilherme O. Campos, Arthur Zimek, Jörg Sander, Ricardo J. G. B. Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E. Houle. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, 30(4):891–927, 2016.

[29] Leonid Portnoy. Intrusion detection with unlabeled data using clustering. Nov. 2001.

[30] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, 28(1):18–28, 2009.

[31] Dit-Yan Yeung and Yuxin Ding. Host-based intrusion detection using dynamic and static behavioral models. *Pattern Recognition*, 36(1):229–243, 2003.

[32] Shiguo Wang. A comprehensive survey of data mining-based accounting-fraud detection research. In *Proc. Int. Conf. Intelligent Computation Technology and Automation*, volume 1, pages 50–53, Changsha, China, May 2010.

[33] Clifton Phua, Vincent Cheng-Siong Lee, Kate Smith-Miles, and Ross W. Gayler. A comprehensive survey of data mining-based fraud detection research. *ArXiv*, abs/1009.6119, 2010.

[34] Sutapat Thiprungsri and Miklos A. Vasarhelyi. Cluster analysis for anomaly detection in accounting data: An audit approach 1. *The International Journal of Digital Accounting Research*, 11:69–84, 2011.

[35] Richard J. Bolton and David J. Hand. Unsupervised profiling methods for fraud detection. 2002.

[36] J. Lin, E. Keogh, Ada Fu, and H. Van Herle. Approximations to magic: finding unusual medical time series. In *Proc. Symp. Computer-Based Medical Systems (CBMS'05)*, pages 329–334, Dublin, Ireland, June 2005.

[37] Rashi Bansal, Nishant Gaur, and Shailendra Narayan Singh. Outlier detection: Applications and techniques in data mining. In *Proc. Int. Conf. Cloud System and Big Data Engineering (Confluence)*, pages 373–377, Noida, India, Jan. 2016.

[38] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. LOF: identifying density-based local outliers. In *Proc. Int. Conf. on Management of data*, pages 93–104, Dallas, Texas, USA, May 2000.

[39] Mingchao Guo, Shijie Pan, Wenmin Li, Fei Gao, Sujuan Qin, XiaoLing Yu, Xuanwen Zhang, and Qiaoyan Wen. Quantum algorithm for unsupervised anomaly detection. *Physica A: Statistical Mechanics and its Applications*, 625:129018, September 2023.

[40] Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. LoOP: local outlier probabilities. *Proc. Conf. on Information and Knowledge Management*, pages 1649–1652, Nov. 2009.
[41] Seunghyuk Cho, Juyong Lee, Jaesik Park, and Dongwoo Kim. A rotated hyperbolic wrapped normal distribution for hierarchical representation learning. *ArXiv*, abs/2205.13371, 2022.
[42] Maximilian Nickel and Douwe Kiela. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. *CoRR*, abs/1806.03417, 2018.
[43] Yoshihiro Nagano, Shoichiro Yamaguchi, Yasuhiro Fujita, and Masanori Koyama. A wrapped normal distribution on hyperbolic space for gradient-based learning. In *Int. Conf. on Machine Learning*, Long Beach, CA, USA, June 2019.
[44] Frédéric Barbaresco. Lie group machine learning and Gibbs density on poincaré unit disk from Souriau Lie groups thermodynamics and SU(1,1) coadjoint orbits. In Frank Nielsen and Frédéric Barbaresco, editors, *Geometric Science of Information*, pages 157–170, Cham, 2019. Springer International Publishing.
[45] Salem Said, Lionel Bombrun, and Yannick Berthoumieu. New Riemannian priors on the univariate normal model. *Entropy*, 16(7):4015–4031, 2014.
[46] Xavier Pennec. Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Vision*, 25:127–154, 2006.
[47] Emile Mathieu, Charline Le Lan, Chris J. Maddison, Ryota Tomioka, and Yee Whye Teh. Continuous hierarchical representations with Poincaré variational auto-encoders. In *Proc. Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019.
[48] Ivan Ovinnikov. Poincaré Wasserstein autoencoder. *CoRR*, abs/1901.01427, 2019.
[49] Seunghyuk Cho, Juyong Lee, and Dongwoo Kim. GM-VAE: Representation learning with VAE on Gaussian manifold. *arXiv preprint arXiv:2209.15217*, 2022.
[50] Y. Baryshnikov and R. Ghrist. Navigating the negative curvature of google maps. Technical report, Technical report of the University of Illinois Urbana-Champaign, February 2023.
[51] Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguñá. Hyperbolic geometry of complex networks. *Phys. Rev. E*, 82:036106, Sep 2010.
[52] Mark McCann and Nicholas Pippenger. Fault tolerance in cellular automata at low fault rates. *Journal of Computer and System Sciences*, 79(7):1126–1143, 2013.
[53] Telesphore Tiendrebeogo and Damien Magoni. Virtual and Consistent Hyperbolic Tree: A New Structure for Distributed Database Management. In *3rd International Conference on Networked Systems*, volume 9466 of *Lecture Notes in Computer Science*, pages 411–425, Agadir, Morocco, May 2015.
[54] P. Petersen. *Riemannian Geometry*. Graduate Texts in Mathematics. Springer, New York, 2006.
[55] Frank Nielsen and Richard Nock. Hyperbolic Voronoi diagrams made easy. In *Proc. Int. conf. on Computational Science and Its Applications*, pages 74–80, Amsterdam, NL, May 31-June 2 2010.
[56] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.
[57] Li-Yu Hu, Min-Wei Huang, Shih-Wen Ke, and Chih-Fong Tsai. The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus*, 5(1):1304, 2016.

CLÉMENCE ALLIETA received her engineering degree in aerospace radiofrequency engineering from ENAC in 2023. She is currently a PhD student at ENAC, under the supervision of Rémi Douvenot and Sonia Cafieri, working on new inversion methods for the study of planetary atmospheres by radio occultation.

JEAN-PHILIPPE CONDOMINES received the Ph.D degree in Automatic Control from Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO) in 2015. He joined the UAV Team of ENAC (French National Civil Aviation University) in 2016 as an Assistant Professor. He obtained the Habilitation degree (HDR) in Automatic Control from University of Toulouse in 2021. He moved to the Dynamic Systems and Controls Group (SYS-DYN) of ENAC in 2022 where he is currently professor. His research interests focus on both theory and application of geometrical methods to solve nonlinear estimation/control problems for dynamical systems.

JEAN-YVES TOURNERET (SM'08, F'19) received the ingénieur degree in electrical engineering from the Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et des Télécommunications (ENSEEIHT) de Toulouse in 1989 and the Ph.D. degree from the National Polytechnic Institute from Toulouse in 1992. He is currently a professor in the university of Toulouse (ENSEEIHT) and a member of the IRIT laboratory (UMR 5505 of the CNRS). His research activities are centered around statistical signal and image processing with a particular interest to Bayesian and Markov chain Monte Carlo (MCMC) methods. He has been the general chair of the International Workshop on Computational Advances in Multi-Sensor Adaptive Processing CAMSAP in 2015 (with P. Djuric) and in 2019 (with D. Brie). He has been a member of different technical committees including the Signal Processing Theory and Methods (SPTM) committee of the IEEE Signal Processing Society (2001-2007, 2010-2015, 2019-2021) and the EURASIP SAT committee on Theoretical and Methodological Trends in Signal Processing (2015-2019). He has been serving as an associate editor for the IEEE Transactions on Signal Processing (2008-2011, 2015-2019) and for the EURASIP journal on Signal Processing (2013-2019). In 2019, He becomes president of the French Association GRETSI. He was a member of the board of directors of EURASIP from 2019 to 2021. From January 2022 to December 2024, He was the president of the EURASIP association.

EMMANUEL LOCHIN received his PhD from the LIP6 laboratory of Pierre and Marie Curie University - Paris VI in December 2004 and the Habilitation Thesis (Habilitation à Diriger des Recherches) in October 2011 from Institut National Polytechnique de Toulouse (INPT). From July 2005 to August 2007, he held a researcher position in the Networks and Pervasive Computing research program at National ICT Australia, Sydney. He then held a full professor position at ISAE-SUPAERO from September 2007 to March 2020 and has co-founded SPEERYT in July 2018 to stimulate the development and diffusion of an on-the-fly coding scheme named Tetrys. Before SPEERYT, this technology was transferred by TTT to a world leader in Internet content distribution. He is now full professor at ENAC since April 2020 and is also member of TéSA laboratory and computer networking expert in the TeSA scientific committee.