10 Distances for Distributions

So far we have mostly talked about distances (and similarities) between two abstract data types: sets (e.g., Jaccard) and vectors (e.g., L_p, cosine). However there is one additional and commonly considered class of abstract representations that is becoming increasing important: probability distributions.

One source of these representations is from the fact that we use randomized algorithms within data mining, or we assume that data is drawn iid from some unknown distribution. In this sense, while there may be an underlying continuous probability distribution representing things, we often have access through repeated trials or observations, and so the representation is a set of observations. What makes this different from set distances is that those do not somehow account for either the probabilistic nature of things, or the geometric notions where each observation is encoded as a vector. How do we compare these probabilistic outputs?

The second source, is in learned representations of objects in a data set. For instance, word vector embeddings take say $n = 100{,}000$ objects and embed each in say $d = 300$ dimensions. The words are embedded each as a vector, and the distance between vectors is important. How do we compare such distributions to each other (e.g., embeddings of words from different languages)?

We have already seen a couple distances, lets revisit them:

10.0.1 Warm Up 1: Discrete distributions

In some cases, we can define a finite collections of states an object can take. For instance, if we assume there are $m = 100{,}000$ words in the English languages. Or there are $m = 29$ counties in Utah, so each event in Utah (e.g., a vote or a lightening strike), occurs in one of those regions. We can then store our observations of events as counts in an m-dimensional vector $v \in \mathbb{R}^m$. If there are n events, we transform this into a discrete probability distribution by L_1-normalizing, that is we divide each entry $v(j)$ by n (the number of observations), this results in a vector v'.

What space does v' lie in? Its more restrictive than $\mathbb{R}^m$, but it is not quite $\mathbb{S}^{m-1}$ (the $(m-1)$-dimensional hypersphere, the results of L_2-normalizing. We label this space

$$\triangle^{m-1} = \{x \in \mathbb{R}^m \mid x_i > 0 \text{ and } \|x\|_1 = 1\}.$$

This spaces is sometimes called the $(m-1)$-simplex, and defines a higher-dimensional equilateral triangle (e.g., $\triangle_3$ is a tetrahedron). It is the convex hull of the vectors $\{e_1 = (1,0,0,0,\ldots), e_2 = (0,1,0,0,0,\ldots), e_3 = (0,0,1,0,0,\ldots),\ldots, e_m = (0,0,0,\ldots,0,1)\}$. And as a result, unlike $\mathbb{S}^{m-1}$, it is a linear space, and straight Euclidean line segments between two elements are contained within the space. As a results, in some sense for $v', u' \in \triangle_m$ then $\mathbf{d}_{\text{Euc}}(v', u') = \|v' - u'\|$ makes sense.

Another common distance is the *Kullback-Leibler Divergence* defined for $a, b \in \triangle^{m-1}$ as:

$$\mathbf{d}_{KL}(a,b) = \sum_{j=1}^{m} a_j \ln(a_j/b_j)$$

It can be derived from information theory, and is given the understanding of a distribution a, how much information is needed to convey (relative to a) to describe a distribution b. Note that this is not a metric, since it is not symmetric. That is we do not always (and usually do not) have $\mathbf{d}_{KL}(a,b) = \mathbf{d}_{KL}(b,a)$.

Another common one is the *Hellinger distance* defined for $a, b \in \triangle^{m-1}$ as:

$$\mathbf{d}_H(a,b) = \frac{1}{\sqrt{2}}\sqrt{\sum_{j=1}^{d}(\sqrt{a_j} - \sqrt{b_j})^2}.$$

It is a metric $\triangle^{m-1}$. It can be interpreted as "lifting" to a wedge of $\mathbb{S}^{m-1}$ and then using the Euclidean distance among those representations.

Or the *Total Variation Distance* defined for $a, b \in \triangle^{m-1}$ is

$$\mathbf{d}_{\text{TV}}(a, b) = \max_{S \subset [m]} \sum_{j \in S} (a_j - b_j) = \frac{1}{2}\|a - b\|_1$$

It inherits the metric properties from L_1, and is in many senses the most sensitive metric on distributions.

10.0.2 Warm Up 2: Kolmogorov-Smirnov in 1 Dimension

The discrete distribution measures restrict comparisons to, well, discrete sets. What if the distribution is inherently continuous, like completion time, rainfall, or height? These have two complications: First, it could be no two observations among distributions μ and ν are at the exact same value (so $\mathbf{d}_{\text{TV}}(\mu, \nu)$ is always 1). Second, these formulations lose the information that 1.0012 inches of rain is very similar to 1.0014 inches of rain.

The Kolmogorov-Smirnov distance provides an elegant and powerful approach for this for distributions over $\mathbb{R}$ (so 1 dimension).

Recall the cumulative density function (CDF) of a distribution μ is defined

$$\text{CDF}_\mu(z) = \int_{y=-\infty}^{z} \mu(z) \mathsf{d}y$$

and its range is always in $[0, 1]$ for a probability distribution μ.

Then the Kolmogorov-Smirnov (KS) distance is defined for two probability distributions μ, ν defined over $\mathbb{R}$ as

$$\mathbf{d}_{\text{KS}}(\mu, \nu) = \max_{z \in \mathbb{R}} |\text{CDF}_\mu(z) - \text{CDF}_\nu(z)|$$

It is a metric. For discrete distributions (e.g, where μ and ν are represented as a sample from potentially continuous distributions), then it can be computed efficiently (like with change-point anomalies) by sorting these points, and scanning them in sorted order while maintaining the CDFs. Note because of this, it is valid to define over any domain $\mathcal{X}$ which has a total sorted order, and is efficient to work with if there is a fast comparator operator.

We next discuss the most common approaches that scale naturally and well to multi-dimensional settings for μ, ν defined over $\mathbb{R}^d$.

10.1 Wasserstein Distances

This is a powerful family of metric distances for distributions μ, ν defined over a metric space $(\mathcal{X}, \mathbf{d})$. That is, it works for some domain $\mathcal{X}$ and a metric *base distance* $\mathbf{d}$ defined on that domain. For this discussion we will restrict to the case where $\mathcal{X} = \mathbb{R}^d$ and for $a, b \in \mathbb{R}^d$ that $\mathbf{d}(a, b) = \mathbf{d}_{\text{Euc}}(a, b) = \|a - b\|$.

Also, for simplicity, lets assume that the two distributions μ_P, ν_Q are actually represented by a set of discrete observations $P \subset \mathbb{R}^d$ for μ_P and $Q \subset \mathbb{R}^d$ for ν_Q. That is, we assume a uniform measure on the points; these may arise via sampling $P \sim \mu_P$ and $Q \sim \nu_Q$. We will also assume both have n points, so $|P| = |Q| = n$. These restrictions are not needed, but will simplify exposition.

Now this metric is to show the cost of transforming from one distribution P to another Q. It accounts for both the probability in P and Q, and the distance between elements p and q. It wants to find a *transportation plan* π that moves each $p \in P$ to some $q \in Q$ that has the minimum cost. However, it cannot move two p_i, p_j to the same q; it must keep them balanced.

A common analogy (which describes the W_1 variant) is the *Earth Movers Distance (EMD)*. It imagines both P as piles of dirt, and Q as a set of holes in the ground, and the goal is to make it flat (it assumes here the amount of dirt and amount of hole space is the same). If the distance $\mathbf{d}(p, q)$ measures the cost of moving dirt from one location p to another q, then EMD measures the total cost of filling all holes under Q with the dirt from P.

In this context, we will describe a transportation plan γ from P to Q as a set of edges from P to Q (a bipartite graph on $P \cup Q$), so $(p_i, q_i) \in \gamma$ if the plan routes from p_i to q_i. Note that if $(p_i, q_i) \in \gamma$, then (p_i, q_j) cannot be in γ if $i \neq j$. And similarly, nor can (p_j, q_i). Let $\Gamma(P, Q)$ be the set of all valid transportation plans between P and Q. (Note this is more complicated if P and Q have different sizes or the weight is not uniform, etc – but the main idea is the same).

Also, note that in $\Gamma(P, Q)$ we do not pre-assign an ordering label $\{p_1, p_2, \ldots, p_i, \ldots, p_n\}$ and $\{q_1, q_2, \ldots, q_i, \ldots, q_n\}$ so we would not always know that $(p_i, q_i) \in \gamma$. We can consider any permutation over Q (e.g., could be $(p_6, q_3) \in \gamma$ is a valid $\gamma \in \Gamma(P, Q)$.

Now finally we can define the Wasserstein W_s distance (for $s \in [1, \infty), \infty$) between two equal-sized point sets $P, Q \subset \mathbb{R}^d$ as

$$W_s(P, Q) = \min_{\gamma \in \Gamma(P,Q)} \left(\frac{1}{|P|} \sum_{(p,q) \in \gamma} \|p - q\|^s \right)^{1/s}$$

This is a metric, with the $s = \infty$ case being defined with a $\max$ operator. Its more general form for distributions μ, ν over metric space $(\mathcal{X}, \mathbf{d})$ with appropriately defined space of transportation plans $\Gamma(\mu, \nu)$ is

$$W_s(\mu, \nu) = \inf_{\gamma \in \Gamma(P,Q)} (\mathbf{E}_{(p,q) \in \gamma} \mathbf{d}(p, q)^s)^{1/s}.$$

As long as $\mathbf{d}$ is a metric over $\mathcal{X}$, then $W_s(\mu, \nu)$ is a metric over probability distributions defined on that domain $\mathcal{X}$. They are sometimes called the *optimal transport* (OT) distance.

The most common forms are the $W_1(P, Q)$, which corresponds with the Earth Movers Distance (EMD), and the $W_2(P, Q)$ which has some nice computational properties and approximation.

Computation. Note that like in the KS distance, we need to optimize over some method of measurement. Unlike KS where we found the maximum difference, here we need to find the best (the minimum cost) transportation plan. This is not as easy as sorting and scanning as in KS (unless $d = 1$). For the $W_2(P, Q)$ and $W_1(P, Q)$, this is a combinatorial optimization problem that has been well-studied, and given the $O(n^2)$ pairs of distances as input, can generically solved in about $O(n^3 \log n)$ time.

For the $W_2(P, Q)$, a fast approximate variant is called the Sinkhorn distance. It "regularizes" the distance with "entropy," but can also be thought of as an interpolation between W_2 and the MMD/kernel distance we discuss next. This makes the problem convex, so the transportation plan $\min_{\gamma \in \Gamma(P,Q)}$ can be solved for efficiently with gradient descent sort of approaches. For an ε-approximation to W_2 it takes roughly $O(\frac{1}{\varepsilon} n^2 \log n)$ time; so if ε is not too small, the the cost is not much more than computing all pairs of n^2 distances – but still expensive if n is large.

This is a really powerful and useful metric, but gets a bit computational painful to work with for n large. It is an active research area to improve the runtime, even for approximate versions, even for special cases.

10.2 Maximum Mean Discrepancy (MMD) / Kernel Distance

Another common "base" way to measure similarity / distance between objects is via a kernel, such as a Gaussian kernel, defined for two points $p, q \in \mathbb{R}^d$ as:

$$K(p, q) = \exp(-\|x - p\|^2).$$

Many other notions of kernels are possible, e.g.,

- Laplace $K(p,q) = \exp(-\|x-p\|)$
- Triangle $K(p,q) = \max\{0, 1 - \|x-p\|\}$
- general Gaussian $K(p,q) = \exp(-\mathsf{d}(x,p)^2)$ for p,q elements of metric space $(\mathcal{X}, \mathsf{d})$.

It is hard to limit what is a "kernel" as there are many useful exceptions to any rule. But here a couple of common traits are

- $K(p,q) \in [0,1]$
- K is *positive definite*. This is equivalent to: for *any* data set X, a "gram" matrix $G \in \mathbb{R}^{n \times n}$ for $|X| = n$ so that $G_{i,j} = K(x_i, x_j)$, and G is positive definite (i.e., all of its n eigenvalues are real and positive).

Note that all examples above satisfy the $[0,1]$ property, and all except Triangle are positive definite.

Now we can define a distance between point sets using $K(p,q)$ as a generalized inner product. Recall that with a standard dot product $\langle p, q\rangle = \sum_{j=1}^d p_j q_j$, then we can write

$$\begin{aligned}\|p-q\|^2 &= \|p\|^2 + \|q\|^2 - 2\langle p,q\rangle \\ &= \langle p,p\rangle + \langle q,q\rangle - 2\langle p,q\rangle\end{aligned}$$

Similarly, we can define the *kernel distance* between two points as

$$\mathsf{d}_K(p,q) = \sqrt{K(p,p) + K(q,q) - 2K(p,q)}.$$

Whenever K is positive definite (minus a few exceptions, mostly for unusual metric spaces), then d_K is a metric. Its value lies in the range $[0, \sqrt{2}]$.

We can then generalize this to over distributions P and Q by defining a generalized notion of similarity between P and Q using a kernel. For that we use the all-pairs average similarity defined as

$$K(P,Q) = \frac{1}{|P|}\frac{1}{|Q|}\sum_{p \in P}\sum_{q \in Q} K(p,q).$$

Following this, *kernel distance* between point sets (aka, *maximum mean discrepancy, MMD*) is defined

$$\mathsf{d}_K(P,Q) = \sqrt{K(P,P) + K(Q,Q) - 2K(P,Q)}.$$

Again (except for a few exotic exceptions), if K is positive definite, then d_K is a metric on point sets.

It can also be extended to general probability distributions μ, ν using $K(\mu,\nu) = \mathbf{E}_{p\sim\mu}\mathbf{E}_{q\sim\nu} K(p,q)$, and $\mathsf{d}_K(\mu,\nu) = \sqrt{K(\mu,\mu) + K(\nu,\nu) - 2K(\mu,\nu)}$, where it is a metric under the same conditions.

d_K vs. W_s: Unlike W_s, these kernel distances do not require solving for an optimal transportation plan $\gamma \in \Gamma(P,Q)$, so are more efficient. Although still (aside approximate methods – not covered here) still require all-pairs computation. They will also lead to a nice Euclidean approximate representation. They are also resistant to outliers in the distributions, since far away points all get the same minimal similarity, and the distance effect (of them in the average) is bounded (e.g., by $\sqrt{2}$).

However, W_2 distances seem to be a bit more refined, as they ensure each point is matched to a point in the other distribution. This gives better alignment and does not miss "modes" of the distributions. The optimal transportation plan can also be useful in understanding what the distances mean, and for deeper structural understanding.

Euclidean-like properties of MMD: Another advantage of the Kernel distance d_k is that there is an interpretation of it as acting Euclidean (with all the good things we know how to do with that).

The Gram matrix G so $G_{i,j} = K(x_i, x_j)$ is a measure of the covariance of the point set under this distance. As such (see PCA / Eigenstructure part of notes) the eigenvectors of this matrix $u_1, u_2, \ldots, u_n$ provide an orthonormal bases. And the Euclidean distance in this basis is equivalent to the kernel distance d_K. That is, after $O(n^3)$ time (for the Eigendecomposition), we perform any Euclidean operations on the point set X (or P, Q). However, this approach needs to know the data X being considered ahead of time.

Another direct method just replaces each instance of a dot product $\langle p, q \rangle$ with the kernel $K(p, q)$. Many L_2 formulations can do this (although sometimes it requires at least an n^2 or n^3 step), and is called the *kernel trick*.

A broader perspective considers the *Reproducing Kernel Hilbert Space, (RKHS)*, $\mathcal{H}_K$. It is a function space, so it contains functions, in this case including $K(x, \cdot) \in \mathcal{H}_K$ for fixed K, but any choice of x.

Importantly, $\mathcal{H}_K$ is larger than this, and also contains linear combinations of these $K(x, \cdot)$. For instance, the *kernel density estimate*

$$\textsc{kde}_X(\cdot) = \frac{1}{|X|} \sum_{x \in X} K(x, \cdot) = \mathbf{E}_{x \in X} K(x, \cdot).$$

These are linear combinations, so for any X then $\textsc{kde}_X \in \mathcal{H}_K$. By considering infinite X we can get arbitrary positive weights, but $\mathcal{H}_K$ also allows negative weights.

Again, through a Eigen-decomposition of the Gram matrix G of any finite set of points we can find a linear, even Euclidean subspace. And Euclidean distance in this subspace is d_K, generally for two elements $f, f' \in \mathcal{H}_K$ as $\|f - f'\|_{\mathcal{H}_K}$. In this framing,

$$\mathsf{d}_K(P, Q) = \|\textsc{kde}_P - \textsc{kde}_Q\|_{\mathcal{H}_K} = \|\mathbf{E}_{p \sim P} K(p, \cdot) - \mathbf{E}_{q \sim Q} K(q, \cdot)\|_{\mathcal{H}_K}.$$

The second formulation gives a hint at where the name *maximum mean discrepancy* comes from (the expected value $\mathbf{E}_{p \sim P} K(p, \cdot)$ is a *mean*), although I like to call it the *kernel distance*.