Distance correlation and distance covariance for binary data

Martin Gebert and Miru Lee

Abstract. We present known formulas for distance correlation and distance variance specifically for binary vectors. We derive that the distance correlation reduces to the absolute value of the Pearson correlation or the Matthews correlation coefficient (MCC) in the case of two binary vectors. We also derive a formula to compute the distance correlation quickly in the case of one binary vector. Our motivation is to enable fast computational algorithms when working with binary vectors. These notes accompany the Rust project dist_corr (https://github.com/mg-gebert/dist_corr) where fast algorithms for binary vectors are implemented. Some timing benchmarks are also added.

1. Preliminaries

The primary motivation for studying quantities such as distance correlation is to find a computationally convenient test for determining whether two vectors originate from independent probability distributions. In probability theory courses, discussions of independence criterions often begin with the characteristic function of a random variable. This is the Fourier transform of a random variable X, i.e. for $t \in \mathbb{R}$

$$\varphi_X(t) = \mathbb{E}[e^{itX}] \tag{1}$$

and also the starting point here. The main relationship with independence of two random variables X,Y on the same probability space is the following

$$X, Y \text{ independent } \iff \forall s, t \in \mathbb{R}: \quad \varphi_{X,Y}(s,t) = \varphi_X(t)\varphi_Y(s). \tag{2}$$

Integrating the right hand side we see that

$$X, Y \text{ independent } \iff \int_{\mathbb{R}} dt \int_{\mathbb{R}} ds |\varphi_{X,Y}(s,t) - \varphi_X(t)\varphi_Y(s)|^2 \omega(t,s) = 0 \tag{3}$$

for any appropriate weight function $\omega > 0$ ensuring finiteness of the above integral. The idea in [SRBN07] is to use the weight function

$$\omega(t,s) = \frac{1}{4t^2s^2}. \tag{4}$$

The reason for this choice is simple and very clever as its Fourier transform is

$$\int_{\mathbb{R}} dt\, e^{-ixt} \frac{1}{|t|^2} = -2|x|. \tag{5}$$

By replacing $\mathbb{E}$ with the sample average, all of the previous results apply to vectors of samples as well. Hence, given two non-constant vectors $v, w \in \mathbb{R}^n$ we define the sample characteristic functions

$$\varphi_{v,w}(t,s) := \frac{1}{n}\sum_{j=1}^{n} e^{-isv_j}e^{-itw_j} \text{ and } \varphi_v(t) := \frac{1}{n}\sum_{j=1}^{n} e^{-itv_j}. \tag{6}$$

Then a computation using (5) shows that

$$\int_{\mathbb{R}} dt \int_{\mathbb{R}} ds \frac{|\varphi_{v,w}(s,t) - \varphi_v(t)\varphi_w(s)|^2}{4t^2s^2} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}B_{ij} \tag{7}$$

where for $i, j = 1, .., n$

$$A_{ij} = |v_i - v_j| - \frac{1}{n}\sum_{i=1}^{n}|v_i - v_j| - \frac{1}{n}\sum_{j=1}^{n}|v_i - v_j| + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}|v_i - v_j| \tag{8}$$

and B_{ij} is defined similarly using w.

Recalling (3), we have now identified a quantity that allows us to infer independence. This leads to the following definition of distance covariance and distance correlation.

Definition 1.1 (Distance correlation). *Let $v, w \in \mathbb{R}^n$ be two non-constant vectors. The distance covariance is defined by*

$$dCov^2(v,w) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}B_{ij} \tag{9}$$

and the distance correlation by

$$dCorr(v,w) = \frac{dCov(v,w)}{dCov(v,v)^{1/2}dCov(w,w)^{1/2}} \tag{10}$$

where A_{ij} and B_{ij} are defined in (8) and dCov is the non-negative square root of $dCov^2$.

Remark 1.1. *From the derivation of (9) and (10) one sees that*

(i) $0 \leq dCorr(v,w) \leq 1$.
(ii) $dCorr(v,w) = 0 \iff v, w$ *independent.*
(iii) $dCorr(v,w) = 1 \iff \exists\, a, b \in \mathbb{R} : v = aw + b$, *i.e. v, w linearly dependent.*

So far, we have defined the distance correlation for two vectors in $\mathbb{R}^n$, meaning vectors of one-dimensional samples. However, distance correlation can also be applied to vectors of higher-dimensional samples by using the Euclidean norm in place of the absolute value in equation (8).

There are various ways to represent the distance covariance. While the formulation (8) in terms of means and grand means of distance matrices is common, it may not always be the most convenient. For instance, it can also be expressed as the trace of a product of appropriate matrices.

Remark 1.2. *Let $v, w \in \mathbb{R}^n$ be two non-constant vectors and $D_v := (|v_i - v_j|)_{i,j}$ and $D_w := (|w_i - w_j|)_{i,j}$ be the corresponding distance matrices. Let $\varphi = (1/\sqrt{n}, ..., 1/\sqrt{n})^T \in \mathbb{R}^n$ then one can rewrite A defined in (7) according to*

$$A = D_v - |\varphi\rangle\langle D_v\varphi| - |D_v\varphi\rangle\langle\varphi| + |\varphi\rangle\langle\varphi|\langle\varphi, D_v\varphi\rangle \tag{11}$$

and B along the same lines in terms of D_w. Using this, we obtain

$$\begin{aligned} dCov^2(v,w) &= \frac{1}{n^2} Tr(P^\perp D_v P^\perp D_w P^\perp) \\ &= \frac{1}{n^2} Tr(D_v D_w) - \frac{2}{n^2}\langle D_v\varphi, D_w\varphi\rangle + \frac{1}{n^2}\langle\varphi, D_v\varphi\rangle\langle\varphi, D_w\varphi\rangle. \end{aligned} \tag{12}$$

where $P^\perp$ is the orthogonal projection onto the orthogonal complement of the subspace spanned by the vector φ, i.e. in bar-ket notation $P^\perp = id - |\varphi\rangle\langle\varphi|$ with id being the identity matrix and Tr denotes the trace of a matrix.

A key fact for what follows is that if $v \in \mathbb{R}^n$ is a binary vector then the distance matrix D_v can be simplified.

Lemma 1.2. *Let $v \in \mathbb{R}^n$ be a 0-1-valued binary vector, i.e. $v_i \in \{0,1\}$. Define $\tilde{v} \in \mathbb{R}^n$ by $\tilde{v}_i = 2v_i - 1 \in \{-1,1\}$ and $\tilde{\varphi} = (1,...,1)^T \in \mathbb{R}^n$. Then*

$$D_v = \frac{1}{2}\Big(|\tilde{\varphi}\rangle\langle\tilde{\varphi}| - |\tilde{v}\rangle\langle\tilde{v}|\Big), \tag{13}$$

i.e. the distance matrix can be represented as a rank-2 matrix.

2. Distance Covariance for two binary vectors

Let $v, w \in \mathbb{R}^d$ be two binary vectors, i.e. vectors which take on only two different values, for simplicity we assume $v_i, w_i \in \{0,1\}$ for all $i = 1,...,n$. In this case, the formula for the distance correlation can be simplified, and it turns out that in this case the distance correlation reduces to the Pearson correlation. More precisely, one obtains the following formula.

Theorem 2.1. *Let $v, w \in \mathbb{R}^n$ such that $v_i, w_i \in \{0,1\}$ for all $i = 1,...,n$. We define the confusion matrix corresponding to the piar v, w*

	$w_i = 1$	$w_i = 0$	Σ
$v_i = 1$	n_{11}	n_{10}	n_{1-}
$v_i = 0$	n_{01}	n_{00}	n_{0-}
Σ	n_{-1}	n_{-0}	n

Then

$$dCov^2(v,w) = \frac{4}{n^4}\big(n_{11}n_{00} - n_{10}n_{01}\big)^2 \tag{14}$$

and

$$dCorr(v,w) = \frac{|n_{11}n_{00} - n_{10}n_{01}|}{\sqrt{n_{1-}n_{-1}n_{0-}n_{-0}}}. \tag{15}$$

Remark 2.1. *The above formula for the distance correlation is known and turns out to be similar to the absolute value of the Pearson correlation which is also called Phi coefficient or the Matthews correlation coefficient (MCC) in the case of two binary vectors, see wikipedia: Phi coefficient.*

For completeness we prove the above formula.

Proof of Theorem 2.1. Let $\varphi := \frac{1}{\sqrt{n}}(1, ..., 1)^T \in \mathbb{R}^n$ and $\tilde{v}, \tilde{w} \in \mathbb{R}^n$ are defined by $\tilde{v}_i := 2v_i - 1$, $\tilde{w}_i := 2w_i - 1$. Using the representation of the distance covariance (12) and (13), we obtain

$$\begin{aligned} \mathrm{dCov}^2(v, w) &= \frac{1}{n^2}\mathrm{Tr}(P^\perp D_v P^\perp D_w P^\perp) \\ &= \frac{1}{4n^2}\langle \tilde{v}, P^\perp \tilde{w}\rangle\langle \tilde{w}, P^\perp \tilde{v}\rangle \\ &= \frac{1}{4n^2}\big(\langle \tilde{v}, \tilde{w}\rangle - \langle \tilde{v}, \varphi\rangle\langle \varphi, \tilde{w}\rangle\big)^2 \end{aligned} \tag{16}$$

where we used $P^\perp \tilde{\varphi} = 0$. We write the latter scalar products in terms of $n_{11}, n_{10}, ...$ and get

$$\begin{aligned} (16) &= \frac{1}{4n^2}\Big(n_{11} + n_{00} - n_{10} - n_{01} - \frac{1}{n}(n_{1-} - n_{0-})(n_{-1} - n_{-0})\Big)^2 \\ &= \frac{1}{4n^4}\Big(((n_{11} + n_{00})^2 - (n_{10} + n_{01})^2) \\ &\qquad - (n_{11} - n_{00} + n_{10} - n_{01})(n_{11} - n_{00} + n_{01} - n_{10})\Big)^2 \end{aligned} \tag{17}$$

where we wrote out $n = n_{11} + n_{10} + n_{01} + n_{00}$, $n_{1-} = n_{11} + n_{10}$ and n_{0-}, n_{-1}, n_{-0} accordingly. Rewriting the latter further we end up with

$$\begin{aligned} (17) &= \frac{1}{4n^4}\Big(((n_{11} + n_{00})^2 - (n_{10} + n_{01})^2) - (n_{11} - n_{00})^2 + (n_{10} - n_{01})^2)\Big)^2 \\ &= \frac{1}{4n^4}\Big(4n_{11}n_{00} - 4n_{10}n_{01}\Big)^2. \end{aligned} \tag{18}$$

The formula for $dCorr$ follows from applying the above formula to (10). □

3. Distance Covariance involving one binary vector

Let $v \in \mathbb{R}^d$ be a binary vector, i.e. $v_i \in \{0, 1\}$ for all $i = 1, ..., n$ and some arbitrary $w \in \mathbb{R}^d$. In this case the formula for the distance correlation can also be simplified and one obtains the following.

Theorem 3.1. *Let $v \in \mathbb{R}^n$ be such that $v_i \in \{0, 1\}$ for all $i = 1, ..., n$ and $w \in \mathbb{R}^d$. Then*

$$dCov^2(v, w) = -\frac{1}{2n^2}\langle P^\perp \tilde{v}, D_w P^\perp \tilde{v}\rangle \tag{19}$$

where $\tilde{v} \in \mathbb{R}^n$ is defined by $\tilde{v}_i := 2v_i - 1$, $D_w := (|w_i - w_j|)_{i,j}$ is the distance matrix corresponding to w and $P^\perp = id - |\varphi\rangle\langle\varphi|$ with $\varphi := \frac{1}{\sqrt{n}}(1, ..., 1)^T \in \mathbb{R}^n$.

Proof. Using the representation (13) of D_v for binary v and $P^\perp \tilde{\varphi} = 0$, we obtain

$$\begin{aligned} \mathrm{dCov}^2(v, w) &= \frac{1}{2n^2}\mathrm{Tr}(P^\perp D_v P^\perp D_w P^\perp) \\ &= -\frac{1}{2n^2}\langle P^\perp \tilde{v}, D_w P^\perp \tilde{v}\rangle. \end{aligned} \tag{20}$$

□

Remark 3.1. *The vector $P^\perp \tilde{v} = \tilde{v} - \langle \tilde{v}, \varphi\rangle\varphi$ can clearly be computed in $O(n)$ and also $\langle P^\perp \tilde{v}, D_w P^\perp \tilde{v}\rangle$ can be calculated in $O(n)$ for an increasingly sorted w. This effectively reduces the complexity of computing equation (19) to that of sorting, that is $O(n \log n)$. The algorithm described above is simpler and faster than the general $O(n \log n)$-algorithm* *for two arbitrary vectors, see* [CH19]. *In the next section, we underline this with numerical evidence.*

4. Performance speed

In this section we evaluate the performance of the $O(n \log n)$ algorithm from [CH19] against improved implementations that take advantage of cases in which one or both input vectors are binary-valued. Benchmarking was performed on a Windows system equipped with an AMD Ryzen 7 PRO 6850U processor and 32 GB of RAM and using the implementation from our Rust project dist_corr (https://github.com/mg-gebert/dist_corr).

4.1. One binary vector. We generate pairs (v_1, v_2) of vectors of length n where the first vector v_1 is randomly chosen within $[-10, 10]$ and the companion vector v_2 is defined by

$$v_2(j) := \begin{cases} 1.0, & \text{if } v_1(j) < 0.0, \\ 0.0, & \text{otherwise}, \end{cases}$$

for $j = 1, ..., n$. This means we compare a general float vector with a binary vector. We compute the distance correlation for various lengths $n = 2^m$ for $m = 6, 8, 10, 12, 14, 16, 18, 20, 22$ for

1. the standard $O(n \log(n))$ algorithm outlined in [CH19].
2. the semi-binary $O(n \log(n))$ algorithm for one binary vector outlined in Section 3.

Table 1. Benchmark median running times (seconds) for general float vs binary

n	standard (s)	semi-binary (s)
2^6	3.8514×10^{-6}	1.2830×10^{-6}
2^8	1.93810×10^{-5}	0.57068×10^{-5}
2^{10}	9.70580×10^{-5}	2.79640×10^{-5}
2^{12}	4.486000×10^{-4}	1.313100×10^{-4}
2^{14}	1.652700×10^{-3}	5.121000×10^{-4}
2^{16}	9.675100×10^{-3}	2.068600×10^{-3}
2^{18}	4.708400×10^{-2}	1.064600×10^{-2}
2^{20}	2.961400×10^{-1}	0.4855400×10^{-1}
2^{22}	1.504900×10^{0}	0.2533500×10^{0}

4.2. Two binary vectors. We generate pairs of vectors (v_1, v_2) of length n. The first vector v_1 is a random binary vector, i.e. 0-1-valued, where the probability of 0 and 1 are 0.5. The companion vector v_2 is defined by

$$v_2(j) := \begin{cases} v_1(j), & \text{if } 2 \mid v_1(j), \\ 0.0, & \text{otherwise}, \end{cases}$$

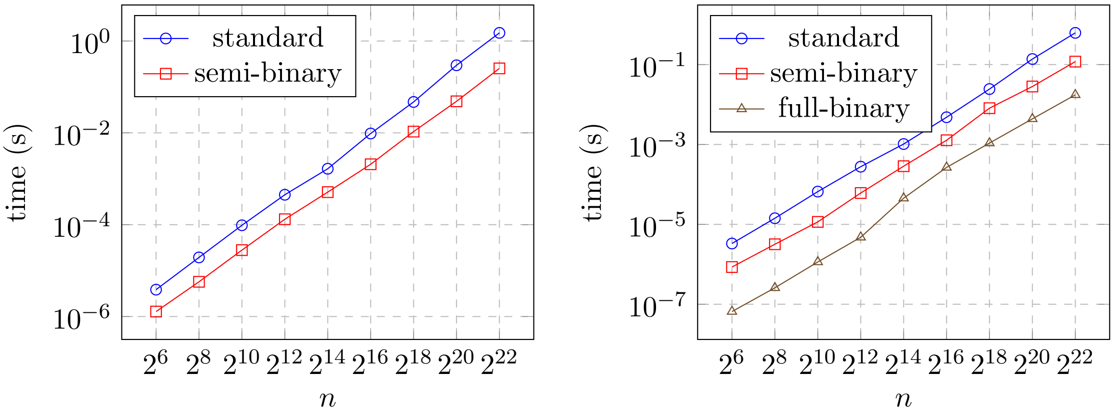
for $j = 1, ..., n$. This means we compare two binary vectors. We compute the distance correlation for various lengths $n = 2^m$ for $m = 6, 8, 10, 12, 14, 16, 18, 20, 22$ for

1. the standard $O(n \log(n))$ algorithm outlined in [CH19].
2. the semi-binary $O(n \log(n))$ algorithm for one binary vector outlined in Section 3. We call this semi-binary.
3. the full-binary $O(n)$ algorithm for two binary vectors outlined in Section 2.

TABLE 2. Benchmark median running times (seconds) for binary vs binary.

n	standard (s)	semi-binary (s)	full-binary (s)
2^6	3.3123×10^{-6}	0.8548×10^{-6}	0.0656×10^{-6}
2^8	1.4228×10^{-5}	0.3175×10^{-5}	0.0258×10^{-5}
2^{10}	6.6114×10^{-5}	1.1531×10^{-5}	0.1144×10^{-5}
2^{12}	2.7786×10^{-4}	0.6089×10^{-4}	0.0475×10^{-4}
2^{14}	1.0255×10^{-3}	0.2861×10^{-3}	0.0452×10^{-3}
2^{16}	4.7995×10^{-3}	1.2748×10^{-3}	0.2663×10^{-3}
2^{18}	2.4360×10^{-2}	0.8102×10^{-2}	0.1084×10^{-2}
2^{20}	1.3763×10^{-1}	0.2827×10^{-1}	0.0438×10^{-1}
2^{22}	6.2279×10^{-1}	1.1863×10^{-1}	0.1749×10^{-1}

FIGURE 1. Comparison of benchmark median running times (seconds)



(A) General vs. semi-binary

(B) General vs semi-binary vs. full-binary

References

[CH19] A. Chaudhuri and W. Hu, A fast algorithm for computing distance correlation, *Computational Statistics & Data Analysis* **135**, 15-24 (2019).

[SRBN07] G. J. Székely, M. L. and Rizzo, and N. K. Bakirov, Measuring and testing dependence by correlation of distances, *The annals of statistics* **6**, 2769–2794 (2007).

VERNAIO GMBH, BOSCHETSRIEDERSTR. 71, 81379 MÜNCHEN, GERMANY

Email address: martin.gebert@vernaio.com

VERNAIO GMBH, BOSCHETSRIEDERSTR. 71, 81379 MÜNCHEN, GERMANY

Email address: miru.lee@vernaio.com