Spearman permutation distances
and Shannon’s distinguishability

Luca Bortolussi\textsuperscript{1}, Liviu P. Dinu\textsuperscript{2}, Andrea Sgarro\textsuperscript{1}

\textsuperscript{1} DMI, University of Trieste, Italy
\textsuperscript{2} Facultatea de matematic\u0103, University of Bucharest, Romania

Abstract. Spearman distance is a permutation distance which might prove useful for coding, e.g. for codes on permutations. It turns out, however, that the Spearman geometry of strings is rather unruly from the point of view of error correction and error detection, and special care has to be taken to discriminate between the two notions of codeword distance and codeword distinguishability. All of this stresses the importance of rejuvenating the latter notion, extending it from Shannon’s zero-error information theory to the more general setting of metric string distances.

1 Foreword

Recent literature \cite{1} has highlighted the importance of codes on permutations, where the Hamming distance is replaced by the Kendall distance, a permutation distance which counts the minimum number of \emph{twiddles}, i.e. of transposition between two adjacent letters, which is required to transform two permuted strings into each other. Codes constructed on the basis of alternative string distances are found also in DNA word design \cite{3}, but in both cases the basic notions of minimum distance, sphere packing, error correction and error detection work as in the standard case of Hamming distances. Is this always so? The answer is no, as we show below by discussing the case of the Spearman distance (called also \emph{footrule distance}, or \emph{rank distance}), itself a permutation distance often met beside Kendall distance, which has the advantage of being computed in linear\textsuperscript{1} time $\Theta(n)$, and which might have an actual role to play in coding theory. The Spearman distance will give us a chance to highlight the importance of rejuvenating a notion due to Shannon, \emph{codeword distinguishability}, or complementarily \emph{codeword confusability}, extended from Shannon’s zero-error information theory \cite{9, 7} to the more general setting of metric string distances, cf. Sections 2 and 5. In Section 2 we give preliminaries based on available material \cite{10, 2} (proofs are omitted, but they are soon re-obtained), while in Section 3 we discuss the case of Spearman distances and show how pervasive the unruly behavior of this

---

\textsuperscript{1}A variant of the Spearman distance has been used in \cite{6} to give the example of a distance for DNA strings which is at the same time non-trivial and quick to compute.
distance is, cf. Theorem 1. In Section 4 we give two actual code constructions, one for strings without repetitions (permutations proper as in [1]), the other for strings with repetitions as in [6]; codeword lengths as short as 3 and 4, respectively, are enough to exhibit the novelties entailed by use of Spearman distances. Final remarks are to be found in Section 5; for basics on distances and coding cf. e.g. [4] and [8], respectively.

2 Distances vs. distinguishabilities

Our distances are integer metric distances $d(x, y)$ for strings $x, y$ of length $n$ over the alphabet $A$ of size $K$, $x, y \in K \subseteq A^n$. The values taken on by all our distances are consecutive integers, a fact which will expedite some of our claims, cf. however footnote 5. Shannon’s distinguishability $\delta(x, y)$ as in [9] is generalized [10, 2] to any distance $d(x, y)$ between two strings $x$ and $y$ belonging to $K$ by setting:

$$\delta(x, y) = \min_{z \in K} [d(x, z) \lor d(y, z)]$$

(Symbols $\lor$ and $\land$ to be used later denote maximum and minimum, respectively.) Any minimizing string $z$ as in (1) is a combinatorial center of the segment $[x, y]$. One can readily check that the following bounds hold true in our integer metric case:

$$\left[\frac{d(x, y)}{2}\right] \leq \delta(x, y) \leq d(x, y)$$

We find it convenient to give the following definition:

Definition 1 An integer metric distance is called Hamming-like when the lower bound in (2) is uniformly met with equality.

Hamming-like means that the string space is dense enough so that, up to an unavoidable integer correction, one always finds centers halfway between the two extremes of the segment $[x, y]$. Proving that a metric distance is Hamming-like is quite easy; it suffices to exhibit a single string $z$ halfway (up to the correction) between $x$ and $y$. This way one can readily prove that the following distances, to give a few examples relevant for coding, are all Hamming-like:

- the Hamming distance $d_H$;
- the Kendall distance $d_K$ on a composition\(^2\) class $K$;
- the DNA distance $d_{DNA}$ (cf. [12]) defined\(^3\) as follows: given a complementation $\chi$ on $A$, i.e. an involutiv e bijection, one sets $d_{DNA}(x, y) = d_H(x, y) \land d_H(x, y^*)$, $y^*$ being the reverse of $\chi(y)$, i.e. the reverse of the string obtained by complementing each letter occurring in $y$.

\(^2\)A composition class is made up of a string together with all of its permutations.

\(^3\)It is quite easy to prove that $d_{DNA}$ is actually pseudometric; the fact that there are distinct strings at zero distance brings about no problems below, cf. also footnote 8.
the Levenshtein distance (edit distance) \( d_{ST} \) where substitutions and twiddles\(^4\) have both cost 1.

When the metric distance is not Hamming-like, error correction and error detection do not work as usual, and one cannot forget about distinguishabilities. Before stating the corresponding criteria, we need a definition, cf. [2]:

**Definition 2** A couple \( x, y \) is called an even couple when all the minimizing centers \( z \) as in (1) are equidistant from \( x \) and \( y \); it is called odd when no center is equidistant; else it is a mixed couple.

The triangular inequality soon implies that, for any couple \((x, y)\) which achieves the lower bound in (2), centers are all equidistant or none equidistant according whether the corresponding distance \( d(x, y) \) is an even or an odd integer, which explains our choice of the names; this is not so in the general case, e.g. if the upper bound is achieved and if \( d(x, y) \) is positive, be it even or odd, the couple cannot be even, because the two extremes are both also centers. These observations will be used in Section 4 to quickly spot even couples.

When decoding by minimum distance, a hard decoder always takes a crisp decision to a single codeword, while a soft decoder declares a detected error if more than one codeword has the same minimum distance from the received output. One has:

**Criteria for error correction and error detection** [10, 2]: The minimum distinguishability between codewords \( \Delta \) is the smallest integer such that all the errors of weight \(< \Delta \) are corrected when using a hard decoder; it is the smallest integer such that all the errors of weight \( \leq \Delta \) are corrected or at least detected when using a soft decoder, provided however that the codebook has been constructed with the constraint that any two codewords at distinguishability equal to \( \Delta \) should form an even couple.

In the case of Hamming-like distances these criteria soon boil down to the usual statements as found in any book on coding theory (recall that even couples are those at even distance), and so one does not need to resort explicitly to distinguishabilities.

### 3 Distinguishabilities for Spearman distances

We move the Spearman distance, defining it directly for strings with possible repetitions. Say \( x = x_1 x_2 \ldots x_n \) is a string of length \( n \) in the composition class \( K \subseteq \mathcal{A}^n \). We may index any such string as follows: if a letter \( a \) occurs \( m > 0 \) times, in the indexed version of \( x \) we write those \( m \) occurrences as \( a_1, a_2, \ldots, a_m \) in the order as they appear; e.g. \( x = aabbea \) is indexed to \( a_1 a_2 b_1 b_2 c_1 a_3 \). If, after

\(^4\)Inadvertently transposing two letters is quite a common source of error when typing, but error-correcting codes based on the Hamming-distance over-estimate twiddles, which count as much as two substitutions. One might prefer to add in \( d_{ST} \) also deletions and insertions, but this requires a slight extension of our approach, cf. footnote 8.
indexing, the \( j \)-th component in \( x \) is the letter \( a_i \); we define the rank \( \pi(a_i|x) \) of \( a_i \) in \( x \) to be its position \( j \); e.g. in our example \( \pi(b_2|x) = 4 \). Unlike usual, in (3) we divide by 2, and this just to expedite some of our claims: like that Spearman distances share a common unit with Kendall distances, i.e. the twiddle, and assume values which are consecutive integers\(^5\). Below possible void summations contribute 0.

**Definition 3** *(Spearman distance, or footnote distance, or rank distance)*

\[
d_S(x, y) = \frac{1}{2} \sum_{a \in A} \sum_{1 \leq i \leq |a|} |\pi(a_i|x) - \pi(a_i|y)|, \quad x, y \in \mathcal{K}
\]  

In the binary case \( A = \{0, 1\} \) the Kendall distance and the (halved) Spearman distance coincide\(^6\), and so, in the binary case, also the Spearman distance is Hamming-like. Instead, when \( |A| \geq 3 \), the Spearman distance \( d_S \) is not Hamming-like, as the following example with a codeword length as short as 3 is enough to show.

**Example 1** Take \( x = abc \) and \( y = cba \) which are at Spearman distance 2 (and at Kendall distance 3). Bounds (2) become \( 1 \leq \delta_S(abc, cba) \leq 2 \). Whenever \( d_S(abc, z) = 1 \) (the two strings differ by a single twiddle) one has \( d_S(cba, z) = 2 \) and so also the Spearman distinguishability is equal to 2.

In this case, it is the right-hand bound in (2) which holds with equality, rather than the left one. The theorem below shows that couples which are not Hamming-like are met in all compositions classes where at least three letters occur explicitly, i.e. at least once. First we need a simple lemma:

**Lemma 1** Let \( x = x_1x_2 \) and \( y = y_1y_2 \) two strings in the composition class \( \mathcal{C} \) such that the prefixes \( x_1 \) and \( y_1 \) of length \( m < n \) are a permutation of each other; then any center as in (1) can be written in the form \( z = z_1z_2 \) where \( z_1 \) is itself a permutation of \( x_1 \).

**Proof:** We begin by an obvious observation: let \( z \) be a center for \( (x, y) \) and alter it to \( z' \) by a single twiddle between two adjacent letters; if, so doing, the contribution of one of these two letters to the summation as in (3) diminishes by 1 both in its distance from \( x \) and in its distance from \( y \), then also \( z' \) is a center for \( (x, y) \). Now, per absurdum assume \( z \) is a center which violates the lemma: then there are in the suffix \( z_2 \) indexed letters matched with indexed letters in the prefixes \( x_1 \) and \( y_1 \), and there are in the prefix \( z_1 \) indexed letters matched with indexed letters in the suffixes \( x_2 \) and \( y_2 \); we shall call them (indexed) letters of type \( \alpha \) and type \( \beta \), respectively. Take in the suffix \( z_2 \) the letter of type \( \beta \) with smaller rank; if its rank is not \( m \), transpose it with the preceding letter to get

\(^5\)Without halving nothing special changes, just the integer ceiling in (2) should be replaced by the smallest even integer greater or equal to its argument.

\(^6\)This is soon checked, e.g. use an induction on the weight of the two strings.
a new center $z'$ (recall the observation). One behaves similarly for the prefix $z_1$, moving forward letters of type $\beta$ so as to provide a new center $z''$ which has a letter of type $\beta$ with rank $m-1$ and a letter of type $\alpha$ with rank $m$. If one transposes them, one obtains yet another string $w$, which has strictly smaller Spearman distances from both $x$ and $y$, but this is a contradiction.

**Theorem 1** Take Spearman distances: in any composition class $\mathcal{C}$ with at least three distinct letters which occur explicitly one can provide two strings $x$ and $y$ which do not meet the lower bound in (2) with equality.

**Proof:** Recalling the example above, just take $x = abcx_2$ and $y = cbax_2$ with a common suffix $x_2$ of length $n - 3$; one has $d_S(abcx_2, cbax_2) = d_S(abc, cba) + d_S(x_2, x_2) = d_S(abc, cba) = 2$. The lemma implies that also the distinguishability is additive: $\delta_S(x, y) = \delta_S(x_1, y_1) + \delta_S(x_2, y_2)$, which in our case is simply $\delta_S(abc, cba) = 2$.

### 4 Two ternary code constructions

To emphasize the importance of distinguishabilities in a general theory of coding, it will be enough give two code constructions on the ternary alphabet $\mathcal{A} = \{a, b, c\}$ with lengths as short as $n = 3, 4$. Basically, computations are by exhaustive\(^7\) search. For $n = 3$ the only ternary composition class contains $abc$ as in the example above, for $n = 4$ there are 3 ternary classes, but because of symmetry we shall take into account just the class which contains $aabc$. **Distinguishability graphs** with strings as vertices are already found in Shannon’s zero-error theory; here we flank them also by **expurgated graphs**, needed for error detection.

**Definition 4** Fix the integer threshold $\Delta$: in the distinguishability graph two vertices are linked if their distinguishability is $\geq \Delta$; from this graph, the expurgated distinguishability graph is obtained by deleting all edges at distinguishability equal to the threshold $\Delta$ which correspond to odd and mixed couples.

(Maximal) cliques in the distinguishability graph correspond to (maximum-size, optimal) codes which, when using a hard decoder, allow to correct errors of weight $< \Delta$; (maximal) cliques in the expurgated graph correspond to (maximum-size, optimal) codes which, when using a soft decoder, allow to correct or at least detect errors of weight $\leq \Delta$.

For length 3, the relevant thresholds are $\Delta = 1$ limited to the expurgated graph (a single twiddle is detected) and $\Delta = 2$. We write the distance and the

---

\(^7\)A systematic building of a theory for Spearman-distance codes is likely to entail combinatorial problems which are all but easy. In this correspondence, we do not try to bound optimal transmission rates in function of pre-assigned and suitably normalized distinguishabilities, which would have to take the place of pre-assigned normalized distances, or correction rates.
distinguishability matrix, leaving empty the all-zero main diagonal to improve readability:

<table>
<thead>
<tr>
<th>$d_S, \delta_S$</th>
<th>$abc$</th>
<th>$acb$</th>
<th>$bac$</th>
<th>$bea$</th>
<th>$cab$</th>
<th>$cba$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$abc$</td>
<td>1.1</td>
<td>1.1</td>
<td>2.1</td>
<td>2.1</td>
<td>2.1</td>
<td>2.1</td>
</tr>
<tr>
<td>$acb$</td>
<td>1.1</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
</tr>
<tr>
<td>$bac$</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>2.2</td>
<td>2.1</td>
<td>2.1</td>
</tr>
<tr>
<td>$bea$</td>
<td>2.1</td>
<td>2.2</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
<tr>
<td>$cab$</td>
<td>2.1</td>
<td>1.1</td>
<td>2.2</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
<tr>
<td>$cba$</td>
<td>2.2</td>
<td>2.1</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
</tbody>
</table>

Even couples are those corresponding to the matrix entries 2, 1 (the lower bound in (2) is achieved and the distance is an even integer, while for entries 1, 1 the lower bound is achieved and the distance is odd, and for entries 2, 2 it is the upper bound which is achieved).

In the expurgated graph with $\Delta = 1$ a maximal clique is $\{abc, bea, cab\}$ with size 3 (single twiddles are detected). In the non-expurgated distinguishability graph with $\Delta = 2$ a maximal clique is $\{abc, cba\}$ with size 2 (single twiddles are corrected), while the expurgated graph with the same $\Delta$ has no edges, and so no error detection is feasible of error weight 2.

We move to length 4 and we write the distance and distinguishability matrix for the twelve permutations of $aabc$:

<table>
<thead>
<tr>
<th>$d_S, \delta_S$</th>
<th>$A$</th>
<th>$B$</th>
<th>$C$</th>
<th>$D$</th>
<th>$E$</th>
<th>$F$</th>
<th>$G$</th>
<th>$H$</th>
<th>$I$</th>
<th>$L$</th>
<th>$M$</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>1.1</td>
<td>1.1</td>
<td>2.1</td>
<td>2.1</td>
<td>2.2</td>
<td>2.2</td>
<td>3.2</td>
<td>3.2</td>
<td>3.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>$B$</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>2.2</td>
<td>2.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>3.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>$C$</td>
<td>1.1</td>
<td>2.1</td>
<td>2.2</td>
<td>1.1</td>
<td>2.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>3.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>$D$</td>
<td>2.1</td>
<td>1.1</td>
<td>2.2</td>
<td>2.1</td>
<td>1.1</td>
<td>3.2</td>
<td>1.1</td>
<td>3.2</td>
<td>3.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>$E$</td>
<td>2.1</td>
<td>2.2</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>3.2</td>
<td>1.1</td>
<td>3.2</td>
<td>3.2</td>
<td>4.2</td>
<td>4.2</td>
<td>4.2</td>
</tr>
<tr>
<td>$F$</td>
<td>2.2</td>
<td>2.1</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
<td>3.2</td>
<td>2.1</td>
<td>1.1</td>
<td>2.2</td>
<td>2.1</td>
<td>2.2</td>
<td>2.2</td>
</tr>
<tr>
<td>$G$</td>
<td>2.1</td>
<td>3.2</td>
<td>1.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>3.3</td>
<td>1.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>3.2</td>
</tr>
<tr>
<td>$H$</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>1.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.3</td>
<td>3.2</td>
<td>1.1</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
</tr>
<tr>
<td>$I$</td>
<td>3.2</td>
<td>3.2</td>
<td>2.1</td>
<td>3.2</td>
<td>1.1</td>
<td>2.1</td>
<td>1.1</td>
<td>3.2</td>
<td>2.2</td>
<td>1.1</td>
<td>2.1</td>
<td>2.1</td>
</tr>
<tr>
<td>$L$</td>
<td>3.2</td>
<td>3.2</td>
<td>2.1</td>
<td>2.1</td>
<td>2.1</td>
<td>1.1</td>
<td>3.2</td>
<td>1.1</td>
<td>2.1</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
<tr>
<td>$M$</td>
<td>4.2</td>
<td>4.2</td>
<td>3.2</td>
<td>3.2</td>
<td>2.1</td>
<td>2.2</td>
<td>2.1</td>
<td>3.2</td>
<td>1.1</td>
<td>1.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
<tr>
<td>$N$</td>
<td>4.2</td>
<td>4.2</td>
<td>3.2</td>
<td>3.2</td>
<td>2.2</td>
<td>2.1</td>
<td>2.1</td>
<td>3.2</td>
<td>2.1</td>
<td>1.1</td>
<td>1.1</td>
<td>1.1</td>
</tr>
</tbody>
</table>

with $A = abca$, $B = aacb$, $C = abac$, $D = acab$, $E = acab$, $F = acba$, $G = baac$, $H = caab$, $I = baca$, $L = caba$, $M = beaa$, $N = cbaa$. Even couples are those corresponding to entries 2, 1 or 4, 2, respectively, and are spotted as we did above. By the way, take e.g. $(A,I)$ for an example of a mixed couple: C and F are both centers, F is equidistant from A and I, but C is not; one has $d_S(A, N) > d_S(H, G)$ and yet $\delta_S(A, N) < \delta_S(H, G)$.

The relevant thresholds are $\Delta = 1$ limited to the expurgated graph, $\Delta = 2$ and $\Delta = 3$. The expurgated distinguishability graph with $\Delta = 1$ is obtained by taking all edges, save those corresponding to entry 1,1; optimal cliques have size ??, and one of them is ?? . If one wants to correct single twiddles, the adjacency matrix of the distinguishability graph with threshold $\Delta = 2$ is obtained by
writing 1 instead of 2 or 3 in the distinguishability matrix, else 0. Maximal cliques have size 4?, one of them being \{C, D, H, M\}??. From the latter graph, one has to expunge edges corresponding to distinguishability and distance both equal to 2 so as to detect errors corresponding to Spearman distances 2, i.e. two twiddles, or one exchange between two distinct letters with ranks differing by 2. Optimal (maximal) cliques have size ??, and one of them is ???. If one demands to correct errors of Spearman weight 2, threshold \(\Delta = 3\), one must use only codewords at distinguishability 3, and so one must be contented with \{G, H\}. As for the expurgated graph with threshold \(\Delta = 3\), the couple (G,H) is not even, and so no error detection is feasible for error weights 3.

5 History and a vindication

The notion of codeword distinguishability, or equivalently of codeword confusability, goes back to Cl. Shannon, and more precisely to his zero-error information theory, dated 1956, cf. [9]. In Shannon’s case the distance (dissimilarity, distortion) between input and output is 0 when the probability of the transition is positive even if “negligible”, else it is 1; the distinguishability function is 0 or 1, according whether there is, or there is not, at least one output which is accessible from both codewords with positive probability. The third author has put forward a multi-step generalisation\(^8\) of zero-error information theory, called possibilistic\(^9\) information theory, in which the dissimilarity between input and output is no longer constrained to have only two distinct values: distinguishability in the multi-step theory is precisely the notion defined in Definition 1. This correspondence aims at vindicating the essential role of this coding-theoretic notion also in contexts which are quite different from those conceived originally by Shannon.

References


\(^8\)In the general frame, the input and the output alphabets may be distinct, and one starts from a dissimilarity measure (a distortion measure) \(d(x, z)\) between input \(x\) and output \(z\); however, the arguments of the distinguishability \(d(x, y)\) both belong to the input alphabet, cf. [10, 2]. The contents of this correspondence extend more or less literally to the case when the input alphabet is strictly included in the output alphabet, and the dissimilarity measure is a metric distance on the output alphabet; this is useful in the DNA case when one has to rule out inputs with too large “self-hybridization”, and also if one wants to extend the edit distances \(d_{ST}\) to cover also deletions and insertions.

\(^9\)The name derives from the fact that the multi-step theory can be obtained from standard probabilistic information theory, whatever the allowed error probability, if one replaces probabilistic notions by the corresponding possibilistic ones. Possibility theory is a form of multi-valued logic, which can be also seen as a special case of interval-valued (imprecise) probabilities, cf. [11].


[12] new metric distance in DNA word design ??