SOMETIMES THEY COME BACK: TESTING TWO SIMPLE HYPOTHESES
(IN THE REALM OF UNLABELED DATA)

Stefano Marano
DIEM
University of Salerno, Fisciano (SA), Italy
marano@unisa.it

Peter Willett*
P. Willett was supported by NPS via ONR contract N00244-16-1-0017.
ECE Department
University of Connecticut, Storrs, CT, USA
peter.willett@uconn.edu

ABSTRACT
Consider a binary hypothesis where data are independent and identically distributed under the null hypothesis, and known only to be independent under the alternative. The statistician observes an \( n \)-vector \( X^n \) \((n \gg 1)\) and makes a decision using the optimal likelihood ratio test. This seems a widely-known detection problem, but: What if only the set of samples of \( X^n \) are made available to the statistician, while the positions of the individual samples inside the vector are not? Does there exist an optimal test in that case? What is the fundamental performance limit? Are there nicely-performing practical detectors with affordable computational complexity?

Answers to these questions are in large part unknown, despite the fact that the problem – which is becoming known under the name of unlabeled detection – is very relevant in modern sensor network applications where the sample positions can be lost due to their means of delivery from the remote units, or because of network attacks.

Index Terms— Unlabeled detection, fundamental limits of detection, auction algorithm.

1. INTRODUCTION
In many implementations of sensor networks, the data delivered from the remote nodes to the fusion center (FC) carry timing information, which is crucial for certain inference tasks such as binary detection. This is certainly the case when the system is faced with a binary detection problem with independent and identically distributed (i.i.d.) data under the null hypothesis, and independent but not identically distributed data under the alternative. Let \( X^n \) be the vector of observations collected at the FC. If the alternative hypothesis involves a pattern, the exact position (labels) of the samples inside the vector is key for solving the test. However, sensor networks are vulnerable to all sorts of attacks, and a particularly insidious one consists of altering the timing information such that the data arrive at the FC permuted. Even in absence of an attack, there is recent evidence that equipping the observations with time/space labels can be too costly or inefficient in many sensor network implementations, especially when the data are drawn from a finite alphabet of small cardinality, or when the finite cardinality is the consequence of quantization of originally continuous observations. In severely quantized systems, the additional bits necessary to encode timing could instead improve quantization precision, and it is by no mean obvious which use is more important.

This motivates our study that is a revisitation of the classical two-simple-hypotheses statistical test with i.i.d. data under the null hypothesis and independent data under the alternative. The difference with the classical setup is the assumption that the labels of the data have been lost, namely we assume that the FC does not observe \( X^n \), but only an unlabeled version thereof, denoted by \( X^n_u \). That is, the values of the entries of \( X^n \) are observed, but their positions inside \( X^n \) are not. Signal processing with permuted data is of mounting interest, and might refer to observations obtained by broadcast from a sensor network. In any case, we cite [1–3] as examples and access points to potential applications.

As a consequence of the lack of labels, the FC cannot implement the optimal log-likelihood test. One typical alternative is to resort to the generalized likelihood ratio test (GLRT), in which the decision statistic is the largest among all the \( n! \) statistics that correspond to the possible assignments of the \( n \) values available to the FC to the set of \( n \) labels \( \{1, 2, \ldots, n\} \). At first glance, the problem seems combinatorial and therefore impractical. However, we shall see that there exist algorithms able to compute the GLRT statistic in a polynomial time. Intriguingly, even in the case of Gaussian shift-in-mean detection problem (the only case studied so far in the literature, to the best of our knowledge) the maximization needed for computing the GLRT does not require exhaustive search, but simply to order the vector of the data and the vector of the mean values [4] – quite easy computationally.

In this paper we address the unlabeled binary detection problem (i.i.d. under the null, independent under the alternative) in the case where data come from a discrete alphabet. One contribution of the present work is to provide a limiting result that assesses the ultimate performance – the fundamental limit – of the unlabeled detection problem. We also proffer practical algorithms to solve the test with affordable computational complexity in the non-asymptotic regime.
2. PROBLEM FORMULATION

Consider the following standard hypothesis test

\[ H_1 : X^n \sim p_{1:n}(x^n) = \prod_{i=1}^n p_i(x_i), \]
\[ H_0 : X^n \sim q_{1:n}(x^n) = \prod_{i=1}^n q(x_i), \]

where \( X^n = (X_1, \ldots, X_n) \) represents the vector of the observations (lowercase letter \( x_i \) will denote realizations), and where \( q \) and the elements of the sequence \( \{p_i\}_{i=1}^n \) are strictly positive probability mass functions (PMFs) over a common finite alphabet \( \mathcal{X} \). The error probabilities of test (1) are \( P_0(X^n \notin \mathcal{A}) \) (type I error), and \( P_1(X^n \in \mathcal{A}) \) (type II error), where \( P_1 \) and \( P_0 \) are the probability operators under \( H_1 \) and \( H_0 \), respectively, and where \( \mathcal{A} \subseteq \mathcal{X}^n \) is some decision region in favor of \( H_0 \).

We are interested in a modification of the above test that amounts to replacing the observed vector \( X^n \) with its unlabeled version \( X^n_u \), which is defined as the set of the \( n \) values appearing in \( X^n \), without any ordering. This is equivalent to assume that the observed vector has undergone an unknown permutation, or that we observe the sorted (e.g., increasing value) version \( (x_1, \ldots, x_{(n)}) \) of \( X^n \), or that we observe \( n \) and the type vector \( t_{X^n} \), whose \( x \)-th entry \( t_{X^n}(x) \) is the relative frequency of occurrence of the symbol \( x \) in \( X^n \). This leads us to consider the following test:

\[ H_1 : X^n_u \text{ with } X^n \sim p_{1:n}(x^n) = \prod_{i=1}^n p_i(x_i), \]
\[ H_0 : X^n_u \text{ with } X^n \sim q_{1:n}(x^n) = \prod_{i=1}^n q(x_i), \]

for which the decision region \( T_n \) in favor of \( H_0 \) is some subset of \( \mathcal{P}_n \) — the set of all \( n \)-types. The error probabilities for test (2) are \( P_0(t_{X^n} \notin T_n) = P_0(\text{choose } H_1) \) and \( P_1(t_{X^n} \in T_n) = P_1(\text{choose } H_0) \).

For later use, let us introduce the arithmetic mean of the PMFs under \( H_1 \), that is \( \bar{p} = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n p_i \), and let us denote by \( D(q||p) \) the divergence from \( q(x) \) to \( p(x) \) [6].

3. FUNDAMENTAL LIMIT OF UNLABELED DETECTION

\textbf{Theorem.} Consider test (2). Let \( 0 < \alpha < \infty \), and define \( \Psi(\alpha) \triangleq \inf_{q' \in \mathcal{P}(X)} D(q'||\bar{p}) < \alpha \). For any sequence \( \{T_n\} \) of acceptance regions for \( H_0 \):

\begin{equation}
\liminf_{n \to \infty} \frac{1}{n} \log P_0(t_{X^n} \notin T_n) \geq \alpha
\end{equation}

\begin{equation}
\Rightarrow \limsup_{n \to \infty} \frac{1}{n} \log P_1(t_{X^n} \in T_n) \leq \psi(\alpha).
\end{equation}

(b) By using the sequence \( \{U_{n,\alpha}\} \), where \( U_{n,\alpha} = \{t_{X^n} : D(t_{X^n}||q') < \alpha\} \), one gets

\begin{equation}
\liminf_{n \to \infty} \frac{1}{n} \log P_0(t_{X^n} \notin U_{n,\alpha}) \geq \alpha.
\end{equation}

\begin{equation}
\lim_{n \to \infty} \frac{1}{n} \log P_1(t_{X^n} \in U_{n,\alpha}) = \psi(\alpha).
\end{equation}

\[ \Psi(\alpha) \triangleq \inf_{q' \in \mathcal{P}(X)} D(q'||\bar{p}) < \alpha \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]

\[ \Psi(\alpha) \triangleq \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^n D(q_i||p_i). \]
4. PRACTICAL ALGORITHMS

While the result of the previous section gives the complete characterization of the test in the asymptotic regime of unboundedly large number of samples $n$, and provides a satisfying answer to the theoretical questions about the unlabeled detection, the question remains whether there exist practical algorithms that for finite values of $n$ can be implemented with affordable computational complexity.

To address the issue consider first a statistical test in the form (1) and suppose that the labeled vector $\mathbf{x}^n = (x_1, \ldots, x_n)$ is observed. Without loss of generality, let $\mathcal{X} = \{1, 2, \ldots, m\}$ be the alphabet of the observations. The log-likelihood can be written in matrix form:

$$
\begin{bmatrix}
\ell_{11} & \ell_{12} & \cdots & \ell_{1(n-1)} & \ell_{1n} \\
\ell_{21} & \ell_{22} & \cdots & \ell_{2(n-1)} & \ell_{2n} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\ell_{m1} & \ell_{m2} & \cdots & \ell_{m(n-1)} & \ell_{mn}
\end{bmatrix}
$$

(8)

where $\ell_{ki} = \log \frac{p_i(x_k)}{q(k)}$ is the marginal log-likelihood of the $i$-th observed sample $x_i$, when $x_i = k$. The optimal log-likelihood statistic is $\sum_{i=1}^n \ell_{ji}$, where $j_i$ is the value taken by the $i$-th observation $x_i$. Such statistic can be visualized as the sum of $n$ entries of matrix (8), one entry for each column and $n$ $t_{xn}(k)$ entries for the $k$-th row. Regarding matrix (8) as a trellis, this identifies a path over the trellis. As an example, the path emphasized in bold in (8) corresponds to the vector of observations $\mathbf{x}^n = (3, 1, m, \ldots, 2, 3)$.

When the statistician does not observe $\mathbf{x}^n$ but only the unlabeled version thereof, the path over the trellis associated to $\mathbf{x}^n$ cannot be identified, and the optimal decision statistic (the sum of the entries of the optimal path) cannot be computed. What is available, however, is the type $t_{xn}$, which tells us how many entries should be selected on each row. In this situation, a convenient non-optimal but usually nicely-performing alternative approach, is to resort to the GLRT [8]. For the case at hand, the GLRT approach consists of replacing the log-likelihood that contains unknown parameters with its maximum over all the possible choices of these parameters. In other words, among all the possible paths over the trellis (8), which are compatible with the observed $t_{xn}$, the GLRT decision statistic is the one yielding the largest sum. The compatible paths are those with exactly one entry per row, and exactly $n$ $t_{xn}(k)$ entries over the $k$-row, $k = 1, \ldots, m$. Computing the GLRT statistic by means of an exhaustive search over all the compatible paths is combinatorially complex and may be unaffordable even for moderate values of $n$. Fortunately, there exist valid alternatives to exhaustive search. The problem of finding the GLRT path over the trellis (8) is an instance of a transportation problem – a special case of the assignment problem – for which efficient algorithms have been developed [9]. Consider in fact an augmented version of the matrix (8) where each row appearing in (8) is copied a number of times equal to the number of occurrences of the corresponding symbol in the unlabeled vector. Namely, the $k$-th row is copied $n$ $t_{xn}(k)$ times, and we obtain an augmented version of (8) which is an $n$ by $n$ square matrix.

With this trick, it is easily seen that the problem of finding the GLRT path over (8) becomes a standard assignment problem for the square augmented matrix, namely it becomes the classical optimization problem that consists of selecting exactly one entry for each row and one entry for each column, in such a way that the sum of the $n$ selected entries is maximum over all the possible choices. This optimization problem is usually visualized by thinking of each row of the augmented matrix as a person and each column as an object. The augmented matrix is a benefit matrix whose $(k, i)$-th entry is the benefit for person $k$ if it obtains object $i$. To each person must be assigned exactly one object, and to each object must be assigned exactly one person. This is known as the (linear, symmetric) assignment problem.

Many algorithms have been developed for the assignment problem. The Hungarian algorithm solves exactly the problem in $O(n^3)$ operations [10], and is sometimes known as the Munkres or Munkres-Kuhn algorithm [11]. The auction method has lower complexity and is amenable to parallel implementation. There exist many variants of the auction method, and one of the most popular is the $\epsilon$-scaled implementation, which achieves a solution of the assignment problem $\epsilon$-close to the actual maximum (when the benefit matrix is transformed into a matrix of integers, this means that selecting $\epsilon < 1/n$ achieves exactly the maximum) [12]. The computational complexity of the auction algorithm depends on the data structure. When the assignment problem involves similar persons (i.e., equal rows), it can be highly inefficient [13]. A variation of the auction algorithm specifically tailored to address assignment problems with similar persons and similar objects (in fact, an instance of the transportation problem) has been proposed in [13, 14]. The auction algorithm used in this paper is a special form of that proposed in [13], accounting for the presence of similar persons but not of similar objects. The details of the algorithm – here referred to as “auction-sp” – can be found in [13], and a nice overview of auction and its application to data association can be found in [15]. The version we use is $\epsilon$-scaled: there is no guarantee of achieving exactly the maximum over the trellis, but after several trials and errors we find that $\epsilon = 10^{-3}/m$ practically achieves the same total benefit as the Hungarian algorithm. In the numerical experiments that follow we set $\epsilon = 10^{-3}/m$.

As an alternative to the auction-sp algorithm, we also consider two greedy procedures. The first, called “greedy-1”, proceeds sequentially over the observed sorted (e.g., increasing ordered) version $(x_{(1)}, \ldots, x_{(n)})$ of vector $\mathbf{x}^n$ as follows.

---

Set $\Gamma = \text{matrix (8)}$

For $k = 1, \ldots, n$, do: add to the path the maximum entry over the $x_{(k)}$-th row of $\Gamma$, then remove from $\Gamma$ the column where the maximum has been found

end do

---
The second algorithm, referred to as “greedy-2” is now described. First of all, regardless of the observed number of symbol occurrences $n t_{x_{m}}$, a candidate path is selected over the trellis (8) by choosing the maximum entry over each column. To this candidate path it corresponds some type $t_{x_{m}}$: $n t_{x_{m}} (k)$ equals the number of times that an entry over the $k$-th row is included in the candidate path. If $t_{x_{m}} = t_{x_{m}}$ we are done. Otherwise there are a number of symbol modifications, say $h$, needed to convert $t_{x_{m}}$ to $t_{x_{m}}$. Let us say that symbols $x_{1}, \ldots, x_{h}$ must be replaced with $x_{1}, \ldots, x_{h}$. The algorithm greedy-2 proceeds sequentially over the entries of these two sequences in the following way.

Set $\Gamma =$ matrix (8)

For $k = 1, \ldots, h$, do:
- consider all the columns of $\Gamma$ for which the entries on the $x_{k}$-th row have been included in the candidate path, denoted by $\gamma_{x_{k}1}, \ldots, \gamma_{x_{k}j}$;
- consider also the entries over the same columns, but on the $x_{k}$-th row, namely $\gamma_{x_{k}1}, \ldots, \gamma_{x_{k}j}$;
- $\delta' = \arg \min_{\delta_{1}, \ldots, \delta_{m}} \{ \gamma_{x_{k}1} - \gamma_{x_{k}1}, \ldots, \gamma_{x_{k}j} - \gamma_{x_{k}j} \}$;
- modify the candidate path by replacing the current entry $\gamma_{x_{k}\delta'}$ with $\gamma_{x_{k}\delta'}$; remove column $\delta'$ from $\Gamma$

end do

The rationale is to modify the candidate path in such a way that (i) each modification yields the minimum reduction of the total sum of the path entries, and (ii) the final path complies with the observed $t_{x_{m}}$.

4.1. Computer Experiments

To illustrate by computer experiments the performance of the three detectors (auction-sp, greedy-1, greedy-2), we consider two scenarios. Scenario S1: Suppose that under $\mathcal{H}_{0}$ the data are uniformly distributed. Suppose also that we organize the entries of the $n$ PMFs under $\mathcal{H}_{1}$ in a matrix $P = \{p_{ki}\}$, just as we did for the log-likelihoods $\ell_{ki}$ in (8). Then we set the entries of the first column $p_{11}, \ldots, p_{1n}$ as linearly increasing values from $p_{11} = 0$ to $p_{1n} = 2/m$, and we set the entries over the last column $p_{m1}, \ldots, p_{mn}$ all equal to $1/m$.

The entries on the generic $i$-th row of $P$ vary linearly (in increasing or decreasing way) from the first entry $p_{i1}$ to the last entry $p_{in}$. Scenario S2: Each PMF under $\mathcal{H}_{1}$ is randomly and independently generated, by drawing $m$ numbers from a uniform $(0,1)$ distribution and then normalizing to their sum; under $\mathcal{H}_{0}$ we set $q = [\Delta, 2\Delta, \ldots, m\Delta]^{T}$, where $\Delta = \frac{2}{M(M+1)}$.

The results of computer experiments for the two scenarios are shown in Fig. 2.

5. DISCUSSION & CONCLUSIONS

As a benchmark for the detection performance of the detectors introduced in the previous section, we consider the optimal likelihood ratio decision statistic for the detection problem shown in (7), in which data are i.i.d. under both hypotheses. Its performance represents a benchmark for the unlabeled detection because, as shown in Sect. 3, in the limit $n \rightarrow \infty$ the performance of the optimal test for (7) is an upper bound to the performance achievable for the unlabeled case (2).

Our computer experiments, instances of which are shown in Fig. 2, show that the detector performance improve with $n$, while the dependence on $m$ depends on the detection scenario (i.e., data structure). The auction-sp detector performs quite close to the benchmark, especially for large $n$, while the best between greedy-1 and greedy-2 depends on the scenario. Scenario S1 greedy-2 performs exactly as auction-sp.

In terms of computational complexity it turns out that the algorithm greedy-2 is the less time consuming. The execution time is, of course, highly dependent by a number of factors. Just to give an order of magnitude, we mention that in our experiments greedy-2 finishes about $3 \times 10^{-2}$ seconds with $n = 10^{3}$, and about 3 seconds with $n = 10^{4}$, with little variation for $m = 5, 20$. For $m = 5$ and large $n$ (in the order of $10^{3}$ or $10^{4}$), the algorithm greedy-1 may be $\approx 1.5$ times more expensive, and auction-sp requires something like 10 times the execution time of greedy-2. This factor grows with $m$ (for $m = 20$, can reach $\approx 60$ for scenario S1 and more than 20 for scenario S2). The loss factor of greedy-1 does not vary significantly with $m$. All in all, greedy-2 seems to represent a valid detector in many practical cases. For this reason, characterizing its performance with respect to the data structure represents an important open problem.

Other possible topics for future investigations include the characterization of the detection performance for finite $n$, and the study of the rate of convergence to the asymptotic results. It would be also interesting to address the generalization of the theorem in Sec. 3 to the case where observations are drawn from a continuous alphabet.
6. REFERENCES


