Chapter 6

Recognition Problem of Biometrics: Nonparametric Dependence Measures and Aggregated Algorithms

Andrew L. Rukhin

6.1 Introduction

This chapter explores the possibility of using nonparametric dependence characteristics to evaluate biometric systems or algorithms that play an important role in homeland security for the purpose of law enforcement, sensitive areas access, borders and airport control, etc. These systems, which are designed to either detect or verify a person's identity, are based on the fact that all members of the population possess unique characteristics (biometric signatures) such as facial features, eye irises, fingerprints, and gait, which cannot be easily stolen or forgotten. A variety of commercially available biometric systems are now in existence; however, in many instances there is no universally accepted optimal algorithm. For this reason it is of interest to investigate possible aggregations of two or several different algorithms. Kittler, Hatef, Duin, and Matas [220] and Jain, Duin, and Mao ([193], Sec. 6) review different schemes for combining multiple matchers.

We discuss here the mathematical aspects of a fusion for algorithms in the *recognition* or identification problem, where a biometric signature of an unknown person, also known as *probe*, is presented to a system. This probe is compared with a database of, say, N signatures of known individuals called the *gallery*. On the basis of this comparison, an algorithm produces the similarity scores of the probe to the signatures in the gallery, whose elements are then ranked accordingly. The top matches with the highest similarity scores are expected to contain the true identity.

A common feature of many recognition algorithms is representation of a biometric signature as a point in a multidimensional vector space. The similarity scores are based on the distance between the gallery and the query (probe) signatures in that space (or their projections onto a subspace of a smaller dimension). Because of inherent commonality of the systems, the similarity scores and their resulting

orderings of the gallery can be dependent for two different algorithms. For this reason, traditional methods of combining different procedures, like classifiers in pattern recognition, are not appropriate. Another reason for failures of popular methods like bagging and boosting [55] [358] is that the gallery size is much larger than the number of algorithms involved. Indeed the majority voting methods used by these techniques (as well as in analysis of multicandidate elections and social choice theory [381] are based on aggregated combined ranking of a fairly small number of candidates obtained from a large number of voters, judges, or classifiers. The axiomatic approach [267] to this fusion leads to the combinations of classical weighted means (or random dictatorship).

As the exact nature of the similarity scores derivation is typically unknown, the use of nonparametric measures of association is appropriate. The utility of rank correlation statistics, like Spearman's rho or Kendall's tau, for measuring the relationship between different face recognition algorithms, was investigated by Rukhin and Osmoukhina [342]. In Sec. 6.2 the natural extensions of two classical rank correlation coefficients solely based on a given number of top matches are given. We demonstrate difficulties with using these correlation coefficients for estimation of the correlation over the whole gallery. A version of a scan statistic, which measures co-occurrence of rankings for two arbitrary algorithms across the gallery, is employed as an alternative characteristic. The exact covariance structure of this statistic is found for a pair of independent algorithms; its asymptotic normality is derived in the general case.

An important methodological tool in nonparametric dependence characteristics studies is provided by the concept of copula [195]. Special tail-dependence properties of copulas arising in the biometric algorithms analysis are established in Sec. 6.3. For common image recognition algorithms, the strongest (positive) correlation between algorithms similarity scores is shown to hold for both large and small rankings. Thus, in all observed cases the algorithms behave somewhat similarly, not only by assigning the closest images in the gallery but also by deciding which gallery objects are most dissimilar to the given image. This finding is useful for the construction of new procedures designed to combine several algorithms and also underlines the difficulty with a direct application of boosting techniques.

As different recognition algorithms generally fail on different subjects, two or more, methods could be fused to get improved performance. Several such methods for aggregating algorithms are discussed in Sec. 6.4. These methods are based on different metrics on the permutation group and include a simple version of linear fusion suggested by Rukhin and Malioutov [341].

Notice that the methods of averaging or combining ranks can be applied to several biometric algorithms, one of which, say, is a face recognition algorithm, and another is a fingerprint (or gait, or ear) recognition device. Jain, Bolle and Pankanti [192] discuss experimental studies of multimodal biometrics, in particular, fusion techniques for face and fingerprint classifiers. Methods discussed in Sec. 6.4 can be useful in a *verification* problem when a person presents a set of biometric signatures and claims that a particular identity belongs to the provided signatures.

The continued example considered in this chapter comes from the FERET (face recognition technology) program [312]) in which four recognition algorithms

each produced rankings from galleries in three FERET datasets of facial images. It is discussed in detail in Sec. 6.5.

6.2 Correlation Coefficients, Partially Ranked Data, and the Scan Statistic

One of the main performance characteristics of a biometric algorithm is the percentage of queries in which the correct answer can be found in the top few, say, K, matches. To start quantifying dependence between two algorithms for a large gallery size N, it seems sensible to focus only at the images in the gallery receiving the best K ranks. The corresponding metrics for the so-called partial rankings were suggested by Diaconis [100] and studied by Critchlow [95]. A survey of these methods is given in Chap. 11 of [266].

Let X_i and Y_i, $i = 1, \cdots, N$, be similarity scores given to the gallery elements by two distinct algorithms on the basis of a given probe. We assume that the similarity scores can be thought of as continuous random variables, so that the probabilities of ties within the original scores are negligible.

In image analysis it is common to write similarity scores of each algorithm in decreasing order, $X_{(1)} \geq \cdots \geq X_{(N)}$, $Y_{(1)} \geq \cdots \geq Y_{(N)}$, and rank them, so that $X_i = X_{(R(i))}$, and $Y_i = Y_{(S(i))}$. Thus, $X_{(1)}$ is the largest and $X_{(N)}$ is the smallest similarity score while the rank of the largest similarity score is 1, and that of the smallest score is N. We use the notation $\mathbf{R}$ and $\mathbf{S}$ for the vectors of ranks $\mathbf{R} = (R(1), \cdots, R(N))$ and $\mathbf{S} = (S(1) \cdots S(N))$, which can be interpreted as elements of the permutation group $\mathcal{S}_N$. Given a ranking $\mathbf{R}$, introduce the new ranking $\tilde{\mathbf{R}}$ by giving the rank $(N + K + 1)/2$ to all images not belonging to the subset of the best K images (which maintain their ranks). More specifically, new ranks $\tilde{R}_i$ are obtained from the formula

$$\tilde{R}(i) = \begin{array}{ll} R(i) & \text{if } R(i) \leq K \\ \frac{N+K+1}{2} & \text{otherwise.} \end{array}$$

This assignment preserves the average of the largest $N-K$ ranks, so that $\sum_{i=1}^{N} \tilde{R}(i) = \sum_{i=1}^{N} R(i) = N(N+1)/2$. Define the analogue of the Spearman rho coefficient for partial rankings of two algorithms producing rankings $\mathbf{R}$ and $\mathbf{S}$ as the classical rho coefficient for the rankings $\tilde{R}(i)$ and $\tilde{S}(i)$,

$$\tilde{\rho}_S = \frac{\sum_{i=1}^{N} \left(\tilde{R}(i) - \frac{N+1}{2}\right)\left(\tilde{S}(i) - \frac{N+1}{2}\right)}{\sqrt{\sum_{i=1}^{N} \left(\tilde{R}(i) - \frac{N+1}{2}\right)^2 \sum_{i=1}^{N} \left(\tilde{S}(i) - \frac{N+1}{2}\right)^2}}.$$

The advantage of this definition is that by using the central limit theorem for linear rank statistics one can establish, for example, asymptotic normality of the Spearman coefficient when $N \to \infty$. A general result (Theorem 3.2) is formulated later.

The analogue of the Kendall tau coefficient for partial rankings is similarly

defined. Namely, for the rankings $\tilde{R}(i)$ and $\tilde{S}(i)$

$$\tilde{\rho}_K = \frac{\sum_{i,j} sign\left((\tilde{R}(i) - \tilde{R}(j))(\tilde{S}(i) - \tilde{S}(j))\right)}{K(2N - K - 1)}.$$

The denominator, $K(2N-K-1) = K(N-1)+(N-K)K$, can be interpreted as the total number of different pairs formed by the ranks $\tilde{R}(i)$ and $\tilde{S}(i)$. Unfortunately, both of these partial correlation coefficients exhibit the same problem of drastically underestimating the true correlation for small and moderate K.

In accordance with the FERET protocol, four algorithms (I:MIT, March 96; II:USC, March 97; III:MIT, Sept 96; IV:UMD, March 97) have produced similarity scores of items from a gallery consisting of $N = 1196$ images with 234 probe images. The rank correlation matrix based on Spearman rho coefficients is

$$S = \begin{pmatrix} 1 & 0.189 & 0.592 & 0.340 \\ & 1 & 0.205 & 0.324 \\ & & 1 & 0.314 \\ & & & 1 \end{pmatrix}$$

Disappointingly, both coefficients $\tilde{\rho}_S$ and $\tilde{\rho}_K$ have very small values for small and moderate K (see Fig. 6.1). Although they have the tendency to increase as K increases, the largest value of $\tilde{\rho}_S$ (for two most correlated MIT methods I and III) was only 0.29 for $K = 50$.

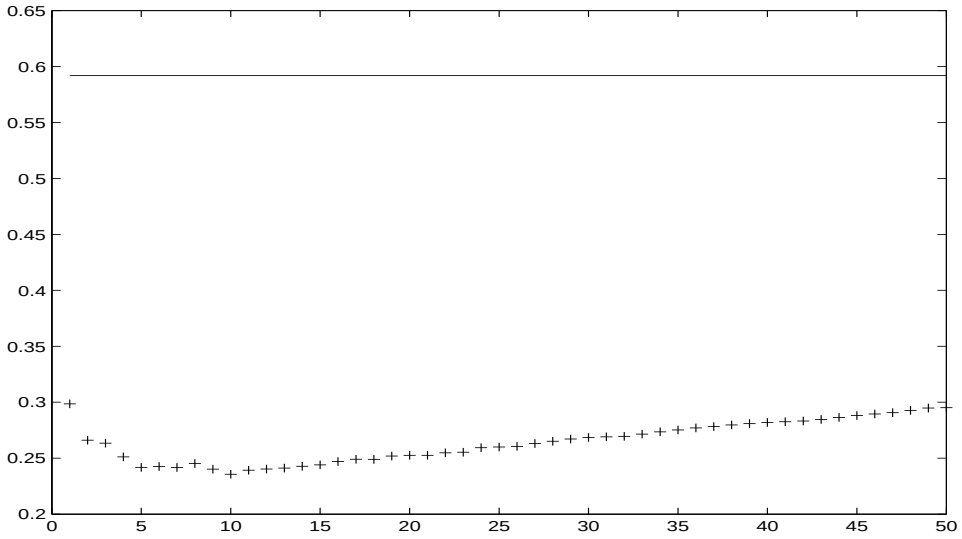

Figure 6.1. *The plot of the partial Spearman rho coefficient for algorithms I and III as a function of K. The solid line represents the limiting value* 0.592.

Another definition of the correlation coefficient for partially ranked data can be obtained from a distance $d(\mathbf{R}, \mathbf{S})$ on the coset space $\mathcal{S}_N/\mathcal{S}_{N-K}$ of partial rankings. The list of the most popular metrics [100] includes Hamming's metric d_H,

Spearman's L_2, Footrule L_1, Kendall's distance, Ulam's distance, and Cayley's distance. With $\bar{d} = \max_{\mathbf{R},\mathbf{S}} d(\mathbf{R}, \mathbf{S})$, let

$$\rho_d = 1 - 2\frac{d(\mathbf{R}, \mathbf{S})}{\bar{d}}$$

be such a correlation coefficient. One can show that, as $N \to \infty$, $\rho_d \to -1$ even for independent $\mathbf{R}$, $\mathbf{S}$, when d is the Kendall metric or the Spearman metric (including L_1 Footrule). For moderate N, $d(\mathbf{R}, \mathbf{S})$ has the expected value too close to $\bar{d}$ for ρ_d to be of practical use. Indeed small variability of ρ_d makes it similar in this regard to the coefficient based on Cayley's distance [101].

To understand the reasons for failure of partial rank correlation characteristics the following *scan* (or co-occurrence) statistic was employed. For two algorithms producing similarity scores X_i and Y_i with rankings $\mathbf{R}$ and $\mathbf{S}$, put for a fixed M and $u = 1, \ldots, N - M + 1$,

$$T(u) = \text{card}\,\{i : u \le R(i)\ S(i) \le u + M - 1\}. \tag{6.1}$$

For independent X_i and Y_i both $\mathbf{R}$ and $\mathbf{S}$ are uniformly distributed over the permutation group $\mathcal{S}_N$. In this case one only needs to consider $W_r = S\left(R^{-1}(r)\right)$. Let $Y_{[i]}$ be the similarity score of the second algorithm corresponding to $X_{(i)}$. These statistics are called *concomitants* of order statistics $X_{(1)}, \ldots, X_{(n)}$. Thus, W_r is the rank of $X_{(i)}$, whose concomitant $Y_{[i]}$ has the rank r, and

$$T(u) = \sum_{u \le r, s \le u+M-1} I(W_r = s),$$

where $I(\cdot)$ is the indicator function. The random variable $T(u)$ counts the common ranks between u and $u + M - 1$. Therefore, in the uniform case it follows a hypergeometric distribution with parameters (N, M, M),

$$P(T = t) = \frac{\left(\begin{array}{c} M \\ t \end{array}\right)\left(\begin{array}{c} N - M \\ M - t \end{array}\right)}{\left(\begin{array}{c} N \\ M \end{array}\right)}, \quad t = 0, 1, \cdots, M.$$

The behavior of the scan statistic for biometric data is very different from that for independent $\mathbf{R}$ and $\mathbf{S}$. Indeed, for all datasets in FERET, the scan statistic exhibits a "bathtub" effect, i.e. its typical plot looks bowl-shaped (see Fig. 6.2). The readings of the scan statistic $T(u)$ for the correlated scores are much larger than the corresponding values based on independent scores for both small and large u. These values for independent scores would oscillate around the mean $ET = M^2/N$. As the variables $T(u)$ must be positively correlated, the covariance function is of interest.

Theorem 6.1. *If the random scores X_i and $Y_i, i = 1, \ldots, N$, are independent, then the covariance function of $T(u)$, for $0 \le h \le N - 1$, $1 \le u \le N - M - h + 1$, has*

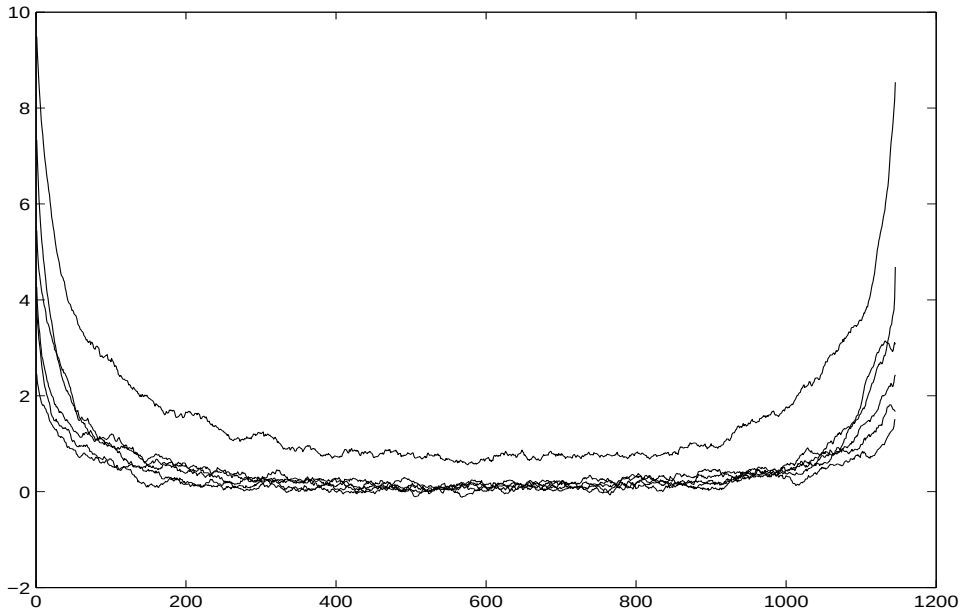

Figure 6.2. *The plots of the scan statistic for algorithms in the FERET study.*

the form

$$Cov(T(u), T(u+h)) = \begin{cases} \dfrac{[(M-h)N - M^2]^2}{N^2(N-1)}, & h < M, \\ \dfrac{M^4}{N^2(N-1)}, & h \geq M. \end{cases}$$

For independent scores neither the covariance between $T(u)$ and $T(u+h)$, nor the mean of $T(u)$ depend on u; $T(u)$ is then a stationary process, and the bathtub effect cannot take place.

6.3 Copulas and Asymptotic Normality

To study the structure of dependence of a pair of algorithms, one can employ the concept of *copula* defined for two random variables X and Y with cumulative distributions functions F_X and F_Y, respectively. In our context X and Y represent random similarity scores of the algorithms. Copula is a function $C(u, v), 0 < u, v < 1$, such that

$$P(F_X(X) \leq u,\ F_Y(Y) \leq v) = C_{X,Y}(u, v) = C(u, v).$$

Copulas are invariant under monotone transformations, i.e. if α and β are strictly increasing, then $C_{\alpha(X),\beta(Y)}(u, v) = C_{X,Y}(u, v)$. In this sense, copulas describe the

structure of dependence. Each copula induces a probability distribution with uniform marginals on the unit square. Nelsen [290] discusses further properties of copulas and methods for their construction.

We assume that the joint distribution of $F_X(X)$ and $F_Y(Y)$ is absolutely continuous, and refer to its density, $c(u,v)$, as copula density. On the basis of a sample, $(X_1,Y_1),\ldots,(X_N,Y_N)$, this function can be estimated by the empirical copula density,

$$c_N\left(\frac{i}{N},\frac{j}{N}\right)=\begin{cases}1/N, & \text{if } \left(X_{(i)},Y_{(j)}\right) \text{ is in the sample,}\\ 0, & \text{otherwise.}\end{cases}$$

Note that c_N is a probability mass function assigning the weight $1/N$ to the point $(R(i)/N, S(i)/N)$, where both $R(i)$ and $S(i)$ are the ranks of X_i and Y_i, respectively. The empirical copula is defined as

$$C_N\left(\frac{i}{N},\frac{j}{N}\right)=\sum_{p=1}^{i}\sum_{q=1}^{j}c_N\left(\frac{p}{N},\frac{q}{N}\right).$$

As the exact distribution of the scan statistic (6.1) for general (dependent) scores appears to be intractable, we give the limiting distribution of $T(u)$ when $N\to\infty$,

$$\frac{u}{N}\to\lambda,\quad \frac{M}{N}\to a,\quad \text{with}\quad 0<\lambda<1-a,\quad 0<a<1. \tag{6.2}$$

With $C(u,v)$ denoting the copula for (X,Y),

$$N^{-1}\sum_{r=u}^{u+M-1}P(W_i=r)\to P(\lambda\le F_X(X)\le\lambda+a,\ \lambda\le F_Y(Y)\le\lambda+a)$$

$$=C(\lambda+a,\lambda+a)+C(\lambda,\lambda)-C(\lambda+a,\lambda)-C(\lambda,\lambda+a),$$

which gives the asymptotic behavior of the mean of the scan statistic.

Theorem 6.2. *Under regularity conditions R1-R5 in [342] when $N\to\infty$, the distribution of $\sqrt{N}\left(\frac{T(u)}{N}-\int_\lambda^{\lambda+a}\int_\lambda^{\lambda+a}c(s,t)\,ds\,dt\right)$ converges to the normal distribution with zero mean and with variance*

$$\sigma^2=Var\Big(I(\lambda\le U\le\lambda+a)I(\lambda\le V\le\lambda+a)+I(U\le\lambda)\int_\lambda^{\lambda+a}c(\lambda,v)dv$$

$$-I(U\le\lambda+a)\int_\lambda^{\lambda+a}c(\lambda+a,v)dv+I(V\le\lambda)\int_\lambda^{\lambda+a}c(u,\lambda)du$$

$$-I(V\le\lambda+a)\int_\lambda^{\lambda+a}c(u,\lambda+a)du\Big).$$

Here λ and a are defined in (6.2), and U and V are random variables with the joint distribution function $C(u,v)$ and the joint density $c(u,v)$.

Theorem 6.2 suggests that the observed bathtub behavior of the scan statistics reflects the form of the underlying copula for the scores. The copulas with a bowl-shaped function of u, $\int_{u/N}^{(u+M)/N} \int_{u/N}^{(u+M)/N} c(s,t)\, ds\, dt$, appear in all FERET algorithms pairs. These copulas correspond to mixtures of two unimodal copulas: one with the bulk of the mass concentrated at the origin $(0,0)$ (small ranks), and the second one concentrated around $(1,1)$ (large ranks). In other terms, the density $c(u,v)$ is bimodal: one peak is at $(0,0)$, and another at $(1,1)$. The set $\{(u,v) : c(u,v) \geq c\}$ is a union of two sets: C_0, which is star-shape about $(0,0)$, and C_1, which is star-shape about $(1,1)$.

In particular, the distribution having such a copula satisfies the definition of *left (right) tail monotonicity* of one random variable U in another random variable V [290]. Namely, $P(U \leq u | V \leq v)$ is a nondecreasing function of v for any fixed u. Also $P(U > u | V > v)$ is a nondecreasing function of v for any fixed u. Each of these conditions implies positive quadrant dependence: $P(U \leq u, V \leq v) \geq P(U \leq u)P(V \leq v)$, (i.e. $C(u,v) \geq uv$), and, under these monotonicity conditions, Spearman's rho is larger than Kendall's tau, which must be positive. All these properties have been observed in all FERET datasets.

In practical terms, tail monotonicity properties mean that the strongest correlations between algorithms similarity scores happen for *both large and small rankings*. Thus, in all observed cases the algorithms behave somewhat similarly not only by assigning the closest images in the gallery, but also by deciding which gallery object is most dissimilar to the given image. The explored algorithms pairs behave more or less independently one from another only in the middle range of the rankings. In the FERET experiment only algorithms I and III (both MIT algorithms, MIT, March 96, and MIT, Sept 96) showed fairly high correlation even for the medium ranks. This finding leads to the conclusion that the partial correlation coefficients, which are based only on small ranks, in principle, cannot capture the full dependence between algorithms.

Verification of the suppressed regularity conditions in Theorem 6.2 for specific families of copulas is usually straightforward. For example, for $\alpha > 0, \beta \geq 1$,

$$C(u,v) = C_{\alpha,\beta}(u,v) = \left\{ \left[\left(u^{-\alpha} - 1\right)^{\beta} + \left(v^{-\alpha} - 1\right)^{\beta} \right]^{1/\beta} + 1 \right\}^{-1/\alpha}, \qquad (6.3)$$

satisfies these regularity conditions ensuring the asymptotic normality of the statistic $T(u)$. This family, for an appropriate choice of α and β, fits the observed similarity scores fairly well.

The next result concerns the asymptotic behavior of the partial correlation coefficient.

Theorem 6.3. *The asymptotic distribution of* $\sqrt{N}(\tilde{\rho}_S - \mu_\rho)$ *is normal with zero mean and variance* σ_ρ^2. *Here* $a = \lim_{N\to\infty} K/N$,

$$\mu_\rho = \left(\frac{a^3}{12} - \frac{a^2}{4} + \frac{a}{4} \right)^{-1}$$

$$\times\Big[\int_0^a\int_0^a uv\,c(u,v)\,du\,dv-\frac{1}{2}\int_0^a\int_0^a(u+v)\,c(u,v)\,du\,dv+\frac{a}{2}\int_0^a\int_a^1 u\,c(u,v)\,du\,dv$$

$$+\frac{a}{2}\int_a^1\int_0^a v\,c(u,v)\,dv\,du+\frac{1}{4}C(a,a)(a+1)^2-\frac{a^2}{4}(2a+1)\Big],$$

$$\sigma_\rho^2=\left(\frac{a^3}{12}-\frac{a^2}{4}+\frac{a}{4}\right)^{-2}$$

$$\times Var\Big(\left[\left(U-\frac{1}{2}\right)I(U\le a)+\frac{a}{2}I(U>a)\right]\left[\left(V-\frac{1}{2}\right)I(V\le a)+\frac{a}{2}I(V>a)\right]$$

$$+\int_U^a\int_0^a v\,c(u,v)\,dv\,du+\frac{1}{2}(a+1)(C(U,a)+C(a,V))+\int_V^a\int_0^a u\,c(u,v)\,du\,dv\Big),$$

U and V are random variables with joint distribution function $C(u,v)$, and the joint density $c(u,v)$.

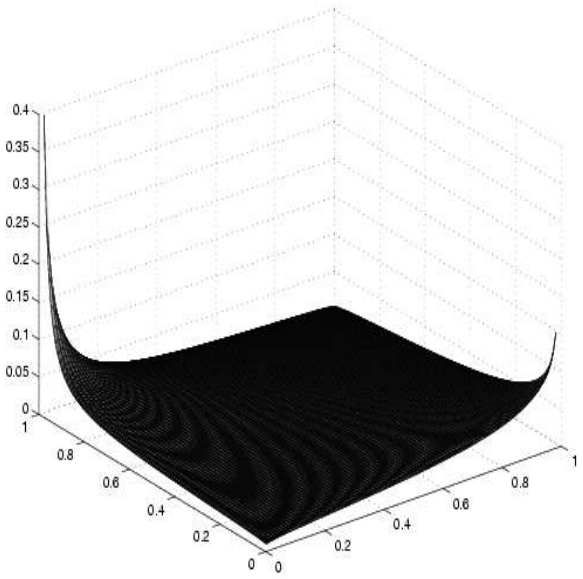

Figure 6.3. *The plot of the estimated theoretical copula for algorithms II and IV with $\alpha = 0.084$, $\beta = 1.227$.*

Genest, Ghoudi, and Rivest [145] discuss pseudo-likelihood estimation of copula parameters. The pseudo-loglikelihood is $l(\alpha,\beta,u,v) = \log c_{\alpha,\beta}(u,v)$. To esti-

mate the parameters α and β, one has to maximize

$$\sum_{i=1}^{N} l\left(\alpha, \beta, \frac{S_i}{N+1}, \frac{R_i}{N+1}\right),$$

which leads to the likelihood-type equations for $\hat{\alpha}$ and $\hat{\beta}$. The complicated form of these equations prevents an explicit form of the estimator. However, the numerical computation is quite feasible. The resulting estimators are asymptotically normal, if in ([95]) $\alpha < \frac{1}{2}$, $\alpha\beta < \frac{1}{2}$, and $\beta < 2$.

Figures 6.3 and 6.4 portray the empirical and theoretical copulas for [95] to pseudo-likelihood estimated α and β.

6.4 Averaging of Ranks via Minimum Distance and Linear Aggregation

A possible model for the combination of, say J, dependent algorithms representable by their random similarity scores $X_1, \ldots, X_J$, involves their joint copula $C_{X_1,\cdots,X_J}$ $(u_1, \cdots, u_J)$, such that

$$C_{X_1,\cdots,X_J}(u_1, \cdots, u_J) = H(F_1^{-1}(u_1), \ldots F_J^{-1}(u_J)),$$

where $F_1, \ldots F_J$ are marginal distribution functions, and H is the joint distribution function of $X_1, \ldots, X_J$.

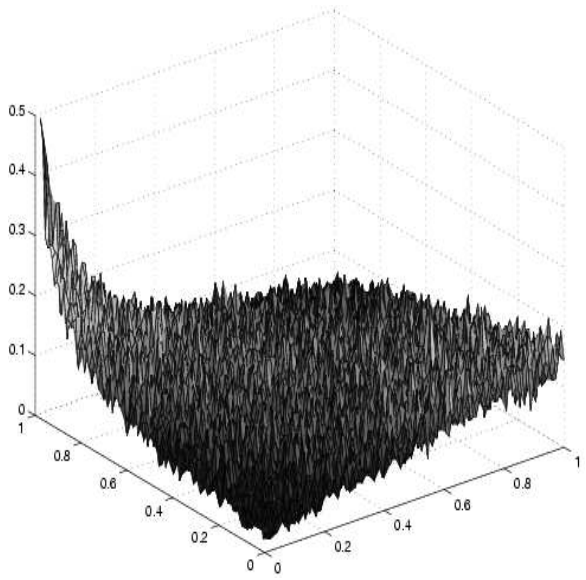

Figure 6.4. *The plot of the empirical copulas for algorithms II and IV.*

If $(X_1^j, \ldots, X_N^j)$ are similarity scores produced by j-th algorithm, the similarity scores of the aggregated algorithm are defined by a convex combination of N-dimensional random vectors $F_j = (F_j^{-1}(X_1^j), \ldots, F_j^{-1}(X_N^j))$, i.e., the score given to the k-th element of the gallery is $\sum_{j=1}^J w_j F_j^{-1}(X_k^j), k = 1, 2, \ldots, N$. To find nonnegative weights (probabilities) $w_1, \ldots, w_J$, such that $w_1 + \cdots + w_J = 1$, we take

$$tr\left(Var(\sum_{j=1}^{J} w_j F_j)\right) = \sum_{j,\ell} w_j w_\ell tr\left(Cov(F_j, F_\ell)\right),$$

as the objective function to be minimized. With the vectors $w = (w_1, \ldots, w_J)^T$, $e = (1, \ldots, 1)^T$, and the matrix S formed by elements $tr\left(Cov(F_j, F_\ell)\right)$, the optimization problem reduces to the minimization of $w^T S w$ under condition $w^T e = 1$ with the solution,

$$w^0 = \frac{S^{-1}e}{e^T S^{-1} e}, \tag{6.4}$$

(assuming that S is nonsingular.)

The matrix S can be estimated from archive data, for example, as the rank correlation matrix based on Spearman rho coefficients in Sec. 6.2. Another possibility is to use the pseudo likelihood estimators of copula parameters (say, α and β in [95]) as discussed in the previous section by plugging them into the formula for $Cov(F_j, F_\ell)$. This typically involves additional numerical integration.

A different (but related) approach is to think of the action of an algorithm (its ranking) as an element of the permutation group $\mathcal{S}_N$. Since the goal is to combine J algorithms whose actions π_j can be considered as permutations of a gallery of size N, the "average permutation," $\hat{\pi}$, of $\pi_1, \ldots, \pi_J$ can be defined by the analogy with classical means. Namely, if $d(\pi, \sigma)$ is a distance between two permutations π and σ, then $\hat{\pi}$ is the minimizer (in π) of $\sum_{j=1}^J d(\pi_j, \pi)$. However, this approach does not take into account different precisions of different algorithms. Indeed, equal weights are implicitly given to all π_i, and the dependence structure of algorithms, which are to be combined, is neglected.

To form a fusion of dependent algorithms, a distance $d((\pi_1, \ldots, \pi_J), (\sigma_1, \ldots, \sigma_J))$, on the direct product $\mathcal{S}_N \bigotimes \cdots \bigotimes \mathcal{S}_N$ of J copies of the permutation group can be used. Then the combined (average) ranking $\hat{\pi}$ of observed rankings $\pi_1, \ldots, \pi_J$ is the minimizer of $d((\pi_1, \ldots, \pi_J), (\pi, \ldots, \pi))$. The simplest metric is the sum $\sum_{j=1}^J d(\pi_j, \pi)$, as above.

To define a more appropriate distance, we associate with a permutation π the $N \times N$ permutation matrix P with elements $p_{i\ell} = 1$, if $\ell = \pi(i)$; $= 0$, otherwise. A distance between two permutations π and σ can be defined as the matrix norm of the difference between the corresponding permutation matrices. For a matrix P, one of the most useful matrix norms is $||P||^2 = tr(PP^T) = \sum_{i,\ell} p_{i\ell}^2$. For two permutation matrices P and S corresponding to permutations π and σ, the resulting distance $d(\pi, \sigma) = ||P - S||$ essentially coincides with Hamming's metric,

$$d_H(\pi, \sigma) = N - \text{card}\,\{i : \pi(i) = \sigma(i)\}.$$

For a positive definite symmetric matrix C, a convenient distance $d((\pi_1,\dots,\pi_J),(\sigma_1,\dots,\sigma_J))$ is defined as

$$d_C((\pi_1,\dots,\pi_J),(\sigma_1,\dots,\sigma_J)) = tr((\Psi-\Sigma)C(\Psi-\Sigma)^T),$$

with $\Psi = P_1 \oplus \cdots \oplus P_J$ denoting the direct sum of permutation matrices corresponding to $\pi_1,\dots,\pi_J$, and Σ having a similar meaning for $\sigma_1,\dots,\sigma_J$.

The optimization problem, which one has to solve for this metric, consists of finding the permutation matrix Π minimizing the trace of the block matrix formed by submatrices $(P_j-\Pi)C_{jk}(P_k-\Pi)^T$, with $C_{jk}, j,k=1,\dots,J$ denoting $N\times N$ submatrices of the partitioned matrix C. In other terms, one has to minimize

$$\sum_{j=1}^{J} tr((P_j-\Pi)C_{jj}(P_j-\Pi)^T)$$

$$= tr\left(\Pi\sum_j C_{jj}\Pi\right)^T - 2tr\left(\Pi\sum_j C_{jj}P_j^T\right) + tr\left(\sum_j P_jC_{jj}P_j^T\right).$$

Matrix differentiation shows that the minimum is attained at the matrix

$$\Pi_0 = \left[\sum_j P_jC_{jj}\right]\left[\sum_j C_{jj}\right]^{-1}.$$

The matrix Π_0^T is stochastic, i.e., $e^T\Pi_0 = e^T$, but typically it is not a permutation matrix, and the problem of finding the closest permutation matrix, determined by a permutation π, remains. In this problem with $\Pi_0 = \{\hat{p}_{i\ell}\}$ we seek the permutation $\hat{\pi}$, which maximizes $\sum_i \hat{p}_{i\pi(i)}$,

$$\hat{\pi} = \arg\max_{\pi} \sum_i \hat{p}_{i\pi(i)}.$$

An efficient solution to this problem can be obtained from the Hungarian method for the assignment problem of linear programming (see [21], Sec. 10.7 for details).

In this setting one has to use an appropriate matrix C, which must estimate on the basis of the training data; C^{-1} is the covariance matrix of all permutations $\pi_1,\dots,\pi_J$ in the training sample.

A simpler aggregated algorithm suggested by Rukhin and Malioutov [341] can be defined by the matrix P, which is a convex combination of the permutation matrices $P_1,\dots,P_J$, $P=\sum_{j=1}^{J} w_jP_j$. Again the problem is that of assigning nonnegative weights $w_1,\dots,w_J$, such that $w_1+\cdots+w_J=1$, to matrices $P_1,\dots,P_J$. The fairness of all (dependent) algorithms can be interpreted as $EP_i=\mu$ with the same "central" matrix μ (in average, for a given probe, all algorithms measure the same quantity), the main difference between them is the accuracy. The optimal weights $w_1^0,\dots,w_J^0$, minimize $E||\sum_j w_j(P_j-\mu)||^2$. Let Σ denote the positive definite matrix formed by the elements $Etr((P_k-\mu)(P_j-\mu)^T)$, $k,j=1,\dots,J$.

The optimization problem still consists in minimization of $w^T\Sigma w$ under condition, $w^T e = 1$. The solution has the form

$$w^0 = \frac{\Sigma^{-1}e}{e^T\Sigma^{-1}e},$$

provided that Σ is nonsingular.

The "covariance matrix" Σ can be estimated by, say, $\hat{\Sigma}$, from the available training data. Note that for all k,

$$Etr(P_k P_k^T) = E\sum_{r,q}\delta_{r\pi(q)} = N,$$

and for $k \neq j$,

$$Etr(P_j P_k^T) = E\text{card}\,\{\ell : \pi_k(\ell) = \pi_j(\ell)\}.$$

Also the training data can be used to estimate μ by the sample mean $\hat{\mu}$ of all matrices in the training set.

Thus, to implement this linear fusion, these estimates are employed to get the estimated optimal weights,

$$\hat{w} = \frac{\hat{\Sigma}^{-1}e}{e^T\hat{\Sigma}^{-1}e}. \tag{6.5}$$

After these weights have been determined from the available data and found to be nonnegative, define a new combined ranking $\hat{\pi}_0$ on the basis of newly observed rankings $\pi_1, \ldots, \pi_J$ as follows. Let the N-dimensional vector $Z = (Z_1, \ldots, Z_N)$ be formed by coordinates $Z_i = \sum_{j=1}^J \hat{w}_j\pi_j(i)$, representing a combined score of element i. Put $\pi_0(i) = \ell$ if and only if Z_i is the ℓ-th smallest of $Z_1, \ldots, Z_N$. In other terms, π_0 is merely the rank corresponding to Z. In particular, according to π_0, the closest image in the gallery is k_0 such that

$$\sum_{j=1}^J \hat{w}_j\pi_j(k_0) = \min_k \sum_{j=1}^J \hat{w}_j\pi_j(k).$$

This ranking π_0 is characterized by the property

$$\sum_{i=1}^N \left(\sum_{j=1}^J \hat{w}_j\pi_j(i) - \pi_0(i)\right)^2 = \min_\pi \sum_{i=1}^N \left(\sum_{j=1}^J \hat{w}_j\pi_j(i) - \pi(i)\right)^2,$$

i.e., π_0 is the permutation that is the closest in the L_2 norm to $\sum_{j=1}^J \hat{w}_j\pi_j$ (see Theorem 2.2, p. 29 in [266]).

If some of the weights $\hat{w}$ are negative, they must be replaced by 0, and the remaining positive weights are to be renormalized by dividing by their sum. This method can be easily extended to the situation discussed in Sec. 6.2 when only partial rankings are available.

A more general approach is to look for matrix-valued weights W_i. These matrices must be nonnegative definite and sum up to identity matrix, $W_1 + \cdots + W_k = I$. The optimization problem remains as above.

Table 6.1. *Size of FERET datasets.*

	D1	D2	D3
Gallery size	1196	552	644
Probe size	234	323	399

The solution has the following, a bit more complicated, form. Let R be the $kN \times kN$ matrix formed by $N \times N$ blocks of the form $E(P_i P_j^T), i, j = 1, \ldots, k$. Partition the inverse matrix $Q = R^{-1}$ in a similar way into submatrices $Q_{ij}, i, j = 1, \ldots, k$. Then the optimal solution is

$$W_i^0 = \sum_j Q_{ij} \left[\sum_{\ell,j} Q_{ij} \right]^{-1}.$$

After the matrix $\hat{P} = \sum_i W_i^0 P_i$ has been found, the combined algorithm ranks the gallery elements as follows:

$$\hat{p}(i) = \arg\max_j p_{ij}.$$

This solution is more computationally intensive as the dimension kN is large, and the matrix R can be ill-conditioned.

6.5 Example: FERET Data

To evaluate the proposed fusion methods, four face-recognition algorithms (I-IV), introduced earlier, were run on three 1996 FERET datasets of facial images, dupI (D1), dupII training (D2), and dupII testing (D3) (see Table 6.1) yielding similarity scores between gallery and probe images. The set D1 was discussed already in Sec. 6.2; the gallery consists of $N = 1196$ images, and 234 probe images were taken between 540 and 1031 days after its gallery match. For the sets D2 and D3 the probe image was taken before 1031 days. The similarity scores were used for training and evaluating the new classifiers; all methods were trained and tested on different datasets.

The primary measures of performance used for evaluation were the recognition rate, or the percent of probe images classified at rank 1 by a method, and the mean rank assigned to the true images. Moreover, the relative recognition abilities were differentiated by the cumulative match characteristic (CMC) curve, which is a plot of the rank against the cumulative match score (the percent of images identified below the rank). Finally, the receiver operating characteristic (ROC) curves were used for measuring the discriminating power of classifiers by plotting the true positive rate against the false positive rate for varying thresholds. The area under ROC curve can be used as another quantitative measure of performance.

Both methods of weighted averaging [100] [101] produced similar weights. For example, the weights obtained from the correlation matrix S based on Spearman

Table 6.2. *Percent of images at rank* 1.

Dataset	Weights	(6.5)	I	II	III	IV
D2	D3	48.6	26.0	59.8	47.1	37.1
D3	D2	67.2	48.4	65.7	72.4	61.4
D1	D3	36.3	17.1	52.1	26.1	20.9

rho coefficients for the training set D1 are $w = (0.22, 0.32, 0.22, 0.24)$, while the weights via [100] are $w = (0.24, 0.27, 0.24, 0.25)$.

These two methods outperformed all but the best of constituent algorithms II. On different pairs of training and testing datasets, the overall recognition rate of these methods fell short of this algorithm by 15% in the worst case and surpassed it by 2% in the best case (Table 6.2). The mean ranks of the two algorithms were generally within 5 ranks of each other.

In terms of CMC curves, the methods of weighted averaging of ranks [100] or [101] improved on all but the best of constituent algorithms, the algorithm II, which was better in the range of ranks from 1 to 30. It looks like this phenomenon is general for linear weighting, namely for small ranks the best algorithm outperforms [100] and [101] for all weights giving this particular algorithm a weight smaller than 1. However, the weighted averaging method [100] was better than all of the four algorithms in the interval of ranks larger than 30 in the D2 dataset (Fig. 6.4). For each of these methods there was about an 85% chance of the true image being ranked 50 or below, which significantly narrowed down the number of possible candidates from more than a 1000 images to only 50.

The experiment showed that the weights derived from training for the different algorithms were all very close, which suggested that equal weights might be given to the different rankings. Although a simple averaging of ranks is a viable alternative to weighted averaging in terms of its computational efficiency, in our examples it was consistently inferior to the methods [100] or [101], and the benefit of training seems apparent.

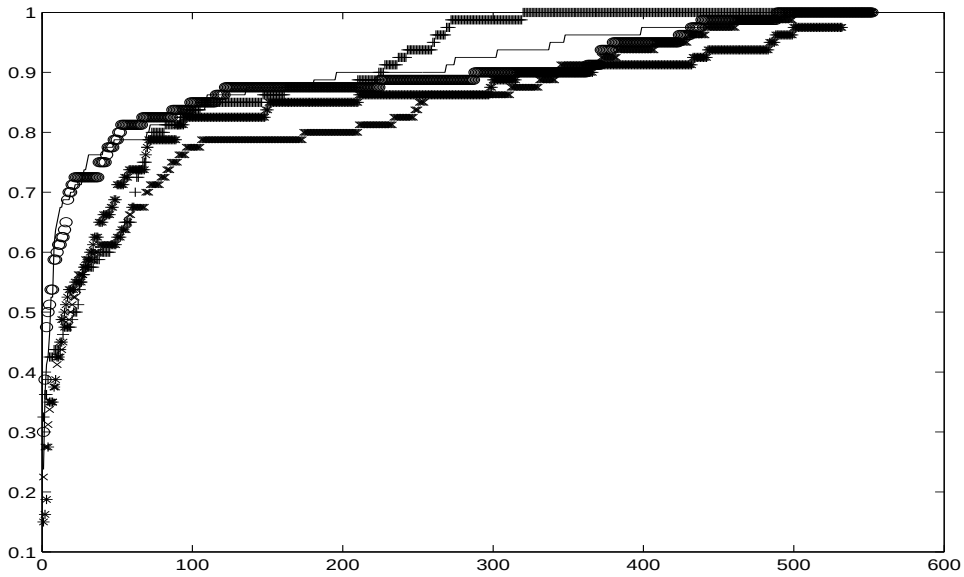

Figure 6.5. *Graphs of the cumulative match curves for algorithms $I - IV$ (marked by *,+, o, x) and the linear aggregation (6.5) (marked by $-$).*