Analyzing time-to-first-spike coding schemes

Lina Bonilla, Jacques Gautrais, Simon Thorpe, Timothée Masquelier

frontiers | Frontiers in Neuroscience

TYPE Original Research
PUBLISHED 26 September 2022
DOI 10.3389/fnins.2022.971937

Check for updates

Analyzing time-to-first-spike coding schemes: A theoretical approach

Lina Bonilla[1]*, Jacques Gautrais[2,3], Simon Thorpe[1] and Timothée Masquelier[1]

[1]CERCO UMR5549, CNRS – Université Toulouse III, Toulouse, France, [2]Centre de Recherches sur la Cognition Animale (CRCA), Centre de Biologie Intégrative (CBI), Université de Toulouse, Toulouse, France, [3]CNRS, UPS, Toulouse, France

OPEN ACCESS

EDITED BY
Arindam Basu,
City University of Hong Kong,
Hong Kong SAR, China

REVIEWED BY
Lei Deng,
Tsinghua University, China
Jyotibdha Acharya,
Institute for Infocomm Research (A*STAR), Singapore

*CORRESPONDENCE
Lina Bonilla
libonill@gmail.com

SPECIALTY SECTION
This article was submitted to Neuromorphic Engineering, a section of the journal Frontiers in Neuroscience

RECEIVED 17 June 2022
ACCEPTED 26 August 2022
PUBLISHED 26 September 2022

CITATION
Bonilla L, Gautrais J, Thorpe S and Masquelier T (2022) Analyzing time-to-first-spike coding schemes: A theoretical approach. *Front. Neurosci.* 16:971937. doi: 10.3389/fnins.2022.971937

Spiking neural networks (SNNs) using time-to-first-spike (TTFS) codes, in which neurons fire at most once, are appealing for rapid and low power processing. In this theoretical paper, we focus on information coding and decoding in those networks, and introduce a new unifying mathematical framework that allows the comparison of various coding schemes. In an early proposal, called rank-order coding (ROC), neurons are maximally activated when inputs arrive in the order of their synaptic weights, thanks to a shunting inhibition mechanism that progressively desensitizes the neurons as spikes arrive. In another proposal, called NoM coding, only the first *N* spikes of *M* input neurons are propagated, and these "first spike patterns" can be readout by downstream neurons with homogeneous weights and no desensitization: as a result, the exact order between the first spikes does not matter. This paper also introduces a third option—"Ranked-NoM" (R-NoM), which combines features from both ROC and NoM coding schemes: only the first *N* input spikes are propagated, but their order is readout by downstream neurons thanks to inhomogeneous weights and linear desensitization. The unifying mathematical framework allows the three codes to be compared in terms of discriminability, which measures to what extent a neuron responds more strongly to its preferred input spike pattern than to random patterns. This discriminability turns out to be much higher for R-NoM than for the other codes, especially in the early phase of the responses. We also argue that R-NoM is much more hardware-friendly than the original ROC proposal, although NoM remains the easiest to implement in hardware because it only requires binary synapses.

KEYWORDS
spiking neural networks, temporal coding, time-to-first-spike coding, rank-order coding, N-of-M coding

1. Introduction

The last decade has seen an explosion in the use of neural networks for demanding AI problems that include computer vision, speech and audio processing, and natural language processing. Indeed, neural networks trained with Deep Learning are now state of the art in many domains. All such systems can be thought of as "neuromorphic" in that they involve large networks of neuron-like elements with connections that resemble the synapses of biological brains. However, there is currently an intense debate about whether future systems will need to include additional neuromorphic features. One key difference between these state-of-the-art AI systems and biology is how information is represented. Artificial systems typically perform calculations using floating-point variables to represent both the neuronal activation levels and the strength of synaptic connections. In contrast, real neurons send information as discrete all or none pulses—spikes. Is this difference important? Spiking Neural Networks (SNNs) are becoming increasingly popular, especially for low-power embedded systems. But many mainstream researchers consider that this difference is essentially irrelevant. Many assume that neurons send information using a firing rate code in which the neuron's activation level is represented by the number of spikes emitted in a given time window. If that was the case, replacing the firing rate with a floating-point number is a perfectly reasonable strategy. However, it has been argued that this sort of firing rate code would be intrinsically very inefficient because you would need a lot of spikes to encode information with any degree of accuracy (Gautrais and Thorpe, 1998). For example, suppose that we wanted to represent the activation level with a precision of 8-bits. To do this using a conventional rate code would mean waiting long enough for the neuron to emit 255 spikes when maximally activated—and this would mean waiting for a second or more to make even the most basic decisions. This very low efficiency has led some researchers to rule out spike-based coding schemes. They point out that it is much simpler, and much more accurate, to represent information as a floating-point number that can be transmitted in a single clock cycle *via* a 32-bit bus.

You could argue that there are alternative ways of implementing a firing rate based code that are much faster. For example, rather than sending an 8-bit activation level using a single neuron that emits between 0 and 255 spikes in a given time window, you could have 255 neurons in parallel, each of which only needs to emit at most one spike in, say, 10 ms. But this sort of population rate coding scheme would also be very inefficient because it would need very large numbers of neurons.

You might also argue that it is possible to estimate the instantaneous firing rate of a neuron by looking at the interval between two spikes. An interspike interval of exactly 4.0 ms would correspond to an instantaneous firing rate of 250 spikes/second. And, in such a case, the accuracy with which the underlying rate can be determined would be limited only by the temporal precision with which the neuron can emit spikes. If the precision was 0.1 ms, you could encode many different activation values in 25 ms. But while possible in principle, such a scheme would require very complex mechanisms to decode as well as being unusable until the neuron has emitted 2 spikes.

It would appear that the fundamental problem here is that researchers have apparently been assuming that spike-based coding has to be some sort of rate coding scheme. But this is certainly not the case. Even the simplest neuronal models have the property that the time taken for a neuron to reach threshold depends on the intensity of the input. And this means that the latency of the first spike in response to a stimulus can be used as a code. Remarkably, variations in spike latency with input intensity were demonstrated in the very first recordings of activity in the optic nerve by Lord Edgar Adrian in Cambridge in the 1920s (Adrian, 1928). But this basic physiological fact was essentially ignored for several decades, before being demonstrated again by neurophysiological studies (Gollisch and Meister, 2008).

Once one accepts the idea that the timing of the first spike provides an alternative way to encode information—a scheme known as time-to-first spike coding (TTFS)—, there are a number of very interesting options that can be considered. In principle, you could use the latency at which a single neuron fires in response to an input to derive information about the intensity of the activation. For example, a neurophysiologist could use an oscilloscope to determine a neuron's latency. But this requires knowing precisely when the stimulus came on. Inside the brain, there is no way to know this. Hence, in this paper we consider an alternative strategy: looking across a population of neurons and determining the order in which they fire. Note that TTFS is not well-suited for dynamic inputs, since coding changes in the input requires additional spikes. We thus focus on static inputs, e.g., flashed images. For simplicity and hardware-friendliness, we also restrict ourselves to non-leaky neurons. A leak is useful to process dynamic inputs because the oldest inputs should be forgotten. Yet it is not required with the static inputs used in this paper.

Historically, TTFS was first proposed to explain the phenomenal speed of processing in the brain for certain tasks, such as object recognition (Thorpe and Imbert, 1989). More recently, TTFS has attracted much attention from the AI community (Mostafa, 2017; Rueckauer and Liu, 2018; Zhou et al., 2019; Kheradpisheh and Masquelier, 2020; Park et al., 2020; Sakemi et al., 2020; Zhang et al., 2020; Comsa et al., 2021; Mirsadeghi et al., 2021), because it can be efficiently implemented on low power event-driven neuromorphic chips (Abderrahmane et al., 2020; Nair et al., 2020; Srivatsa et al., 2020; Göltz et al., 2021; Liang et al., 2021; Oh et al., 2022), leveraging two key features. The first one is sparsity (Frenkel, 2021). Neurons fire at most once, but usually most neurons do not fire at all. Processing thus consumes very few spikes, and thus very little energy, because usually idle neurons do not consume much (Davies et al., 2018). The second one is time. If using event-driven processing, for example, address event representation (AER), time represents itself (Mead, 1990). Thus one can compute with time without ever storing timestamps. For example, a decision can be made based on the first neuron to fire in the readout layer. And this is possible even if the firing time difference is infinitesimally small. Conversely, a readout based on the activation levels requires storing these activation levels with high precision to be able to always distinguish the most active neuron.

It is worth mentioning that neurons are intrinsically sensitive to the timing of their inputs: shifting the input spike times obviously shifts the response time. But here, we consider additional mechanisms that allow neurons to respond selectively to certain input spike time patterns. For example, Rueckauer and Liu (2018), Sakemi et al. (2020), Srivatsa et al. (2020), and Zhang et al. (2020) used linearly increasing excitatory postsynaptic potentials, such that early spikes contribute more. To obtain a similar effect, Park et al. (2020) used a decaying dendritic kernel. Yet in this paper, we focus on spike-based, rather than time-based mechanisms: the input spikes' contribution depends on their arrival ranks rather than on their precise times. The idea is always that the first input spikes contribute more, while later input spikes contribute less, or not at all. This is implemented with a modulation function that decreases with the rank, for example, linearly or geometrically. The net contribution of each input spike to the neuron's potential is then the product of the modulation function with the synaptic weight. The modulation function can also have a cut-off so that the last spikes make no contribution at all.

Our main goal, below, is to lay the foundation of a mathematical framework in order to assess, from a theoretical point of view, the potential of such order-based TTFS coding schemes. As an illustration of this framework, the analysis will be performed upon three instances of such coding schemes: two previous proposals (Rank Order Coding and NoM coding) and a combination of both (Ranked-NoM Coding).

Rank Order Coding (ROC) was an early proposal (Thorpe and Gautrais, 1998). With ROC, all the M afferents of a neuron fire a spike (Figure 1). The modulation is a real number which decreases geometrically with the input spike rank. That means in particular that it is always strictly positive. The synaptic weights are M, $M-1$, ... 1. The final potential is maximal when input spikes arrive in the order of the weights: the first spike should arrive through the synapse with weight M, the second one through the synapse with weight $M-1$, and so on.

N-of-M (NoM) coding is another proposal, in which only the N first spikes among M afferents are propagated (Furber et al., 2004; Thorpe et al., 2019). This first spike pattern can be read out by neurons with binary weights (Figure 2): $W = 4$ ones, and $M - W = 12$ zeros. With random inputs, the final potential has a hypergeometric distribution with N draws from a population of size M containing W successes—or, equivalently, W draws from a population of size M containing N successes (Furber et al., 2004).

For this paper, we have also designed a third type of coding scheme, that we call "Ranked-NoM" (R-NoM) coding, and which incorporates features of both ROC and NoM coding (Figure 2): only the N first spikes among M afferents are propagated, but readout neurons can be selective to a particular order of the N spikes thanks to inhomogeneous weights, and a decreasing modulation function. Later on, we came across an article by Furber et al. (2007) where a similar proposal has been explored in the context of sparse distributed memory (SDM) research. Below, both the weights and the modulation decrease linearly, although other schemes could also be explored using a similar approach (e.g., geometric series as in Furber et al., 2007).

All these codes have been formalized in our unifying mathematical framework that involves:

- A set of weights, which can be homogeneous (as in NoM), or decreasing, either linearly (as in original ROC), or geometrically. This set contains W non-zero weights.
- A modulation function which can be constant (as in NoM), or decreasing, either linearly, or geometrically (as in original ROC). This modulation can also have a cut-off, i.e., becomes zero after the first N spikes.

Our unifying framework allows comparing these codes in terms of discriminative power. We introduce a discriminability measure that quantifies how much more a neuron responds to its preferred pattern than to random inputs. The unifying mathematical framework also allows tuning the parameters of the codes in order to optimize their discriminative power.

We conclude that Ranked-NoM Coding with linearly decreasing modulation and weights offer a particularly interesting compromise between discriminative power and hardware-friendliness.

The paper is organized as follows: the Section 2 briefly introduces the unifying mathematical framework and the discriminability measure. Then, it gives the main analytical formulas for the discriminability of R-NoM, NoM, and ROC, but not their derivations, which can be found in the Supplementary material. Next, we report a numerical study in which we explored the speed-accuracy trade-off for the three different codes. Finally, a brief Discussion summarizes the main results and gives some perspectives.

2. Results

2.1. Mathematical translation of the three coding schemes

The goal is to measure the discriminability power of these codes. We define a measure of selectivity (Equation 2.7) which quantifies how much more the neuron responds to its preferred pattern than to random stimuli.

We first define a random experiment for the spikes generated by M neurons (see Supplementary Section 1.2). For a given stationary stimulus, each of the M input neurones emits one spike. Input patterns will then translate into vectors of size M. We denote Λ the ascending lexically ordered set of the possible permutations over the set $\mathcal{M} = \{0, \ldots, M-1\}$. Cardinality of

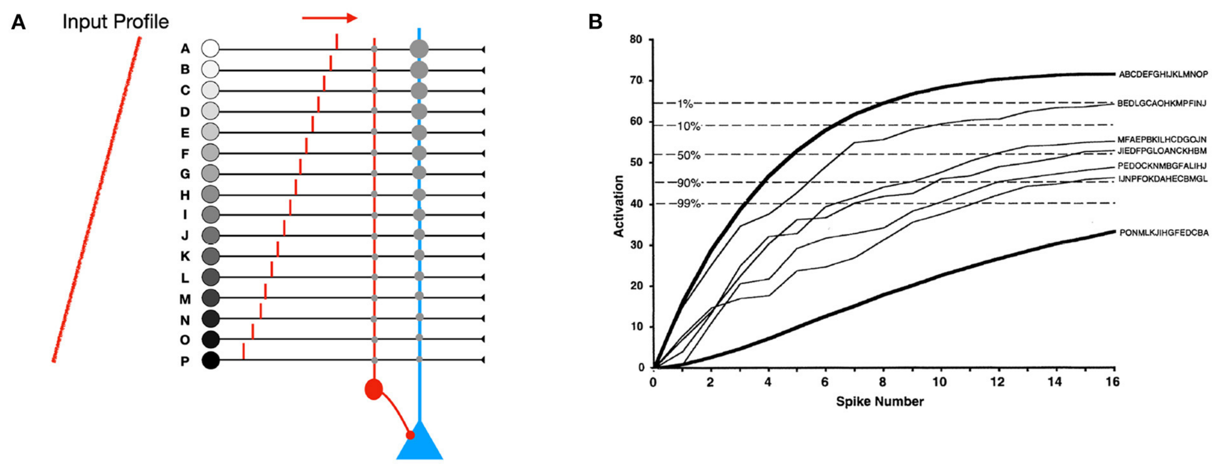


FIGURE 1
(A) Rank order coding (ROC) with $M = 16$ afferents. All the afferents fire exactly one spike. Here we show a neuron selective to the input spike order: A, B, ... , P. Its synaptic weights are linearly decreasing: M for input A, $M - 1$ for input B, and so on, down to 1 for input P. The modulation decreases geometrically with the input spike rank. In practice, this modulation could be implemented with shunting inhibition, as shown with the red inhibitory neuron. **(B)** The increase in activation level depends on the order of firing. Maximal activation occurs when the inputs fire in the order of the weights (A, B, ... , P). Activation is minimal when the order is reversed. Intermediate lines correspond to 5 randomly selected input patterns chosen from the $16! = 20,922,789,888,000$ possible input spike orders. The five dotted lines specify the proportion of such random patterns that will exceed a given final activation level. Modified from Thorpe and Gautrais (1998).

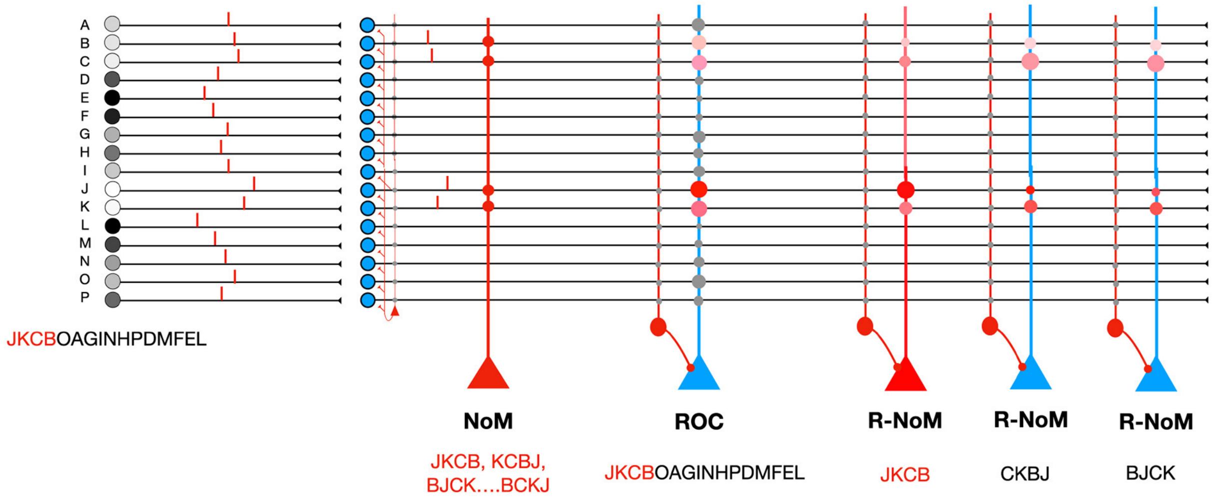


FIGURE 2
Comparison of different codes. On the left, the $M = 16$ afferents fire in the order JKCBOAGINHPDMFEL, but a 4-winner-take-all mechanism only lets the $N = 4$ first spikes through. NoM coding: the readout neuron uses binary weights: $W = 4$ ones, and $M - W = 12$ zeros. The final potential reaches the maximal value of 4 if the N first spikes correspond to the W non-zero weights. The order of these 4 first spikes does not matter. Rank Order coding **ROC.**: the neuron is set up to respond maximally to the order JKCBOAGINHPDMFEL, even though here only the 4 input spikes are propagated. Ranked-NoM coding **R-NoM**:: we show three readout neurons that are selective to three different orders for the 4 first spikes, among the $4! = 24$ possible orders, thanks to graded weights and modulations, both in $\{1, 2, 3, 4\}$.

Λ is then $M!$. We define an application $\mathcal{R}$ that takes values in $\mathcal{D}_K = \{1, 2, .., M!\}$ (ranks of input order in Λ) and returns a vector $\mathbf{r}^k = \mathcal{R}(k)$ in Λ.

To randomly generate sets of input patterns, we define a discrete random variable K over $\mathcal{D}_K$. We can then consider $\mathbf{X} = \mathcal{R}(K)$ as a random vector, and all possible outputs are collected in $\mathcal{D}_\mathbf{X} = \Lambda$. We consider that all input orders have the same probability to occur.

By construction, each component X_i is a discrete random variable taking values from the set $\mathcal{D}_{X_i} = \{0, 1, \ldots, M -$

1} with marginal probability distribution $P_{X_i}(r) = \frac{1}{M}$, and multivariate joint probability distribution $P_{X_1X_2...X_M} = \frac{1}{M!}$. X_i are identically distributed and they are not independent since realizations of $\mathbf{X}$ are permutations from a unique set of values, the one prescribed by the coding scheme, which implies correlation, so that: $\mathrm{Cov}(W_i, W_j) \neq 0$.

This input order is transformed into a vector of weights. For this, we transform the random variable $\mathbf{X}$ in a deterministic way by defining the affine transformation $\mathbf{W} = \Phi(\mathbf{X})$:

$$\Phi(\mathbf{X}) = M - \mathbf{X} = \mathbf{W} \tag{2.1}$$

The marginal and joint probability distributions of the new random variable W_i are determined from the probability distributions of X_i by the change of variables theorem in multivariate calculus. At this stage, the random experiment is fully defined by the random variable $\mathbf{X}$, taking values in Λ, and the bijective function Φ.

We denote Ω the set of the weights vectors. Ω is the base to establish the support of each coding scheme. For this, we define, for each scheme C, a vector-value function Φ_C from Ω to Ω_C and we use the term *score vector* to denote elements in Ω_C.

For ROC (denoted by R), the function Φ_R is the identity function and so its cardinality is $M!$.

For Ranked-NoM Coding (denoted by H), we build the scores-support Ω_H using a function which depends on the parameter $\mathcal{W}$:

$$\Phi_H(\mathbf{w}) = \max(0, \mathbf{w} - M + \mathcal{W}) \tag{2.2}$$

Note that Φ_H maps different permutations onto the same vector permutation. Hence, a subset of vectors that are pure internal permutations among negative or null values will map to the same element of Ω_H. Since the cardinality of these subsets is the number of permutations of the $M - \mathcal{W}$ null elements, the cardinality of Ω_H is:

$$|\Omega_H| = \frac{M!}{(M - \mathcal{W})!} \tag{2.3}$$

For NoM coding (denoted F), we define the scores-support Ω_F from the scores-support Ω_H by the compositions of the indicator function $\mathbf{1}_A$ with Φ_H. Thus we have

$$\Phi_F(\mathbf{w}) = \mathbf{1}_A(\Phi_H(\mathbf{w})) = \mathbf{1}_A(\max(0, \mathbf{w} - M + \mathcal{W})) \tag{2.4}$$

By the indicator function, the vectors in Ω_H get converted into vectors of ones and zeros. As a consequence, the support Ω_F of NoM is reduced because the order is no longer important. Then, we divide by the number of ways you can arrange $\mathcal{W}$ numbers, which is $\mathcal{W}!$. Thus, the cardinality of Ω_F is:

$$|\Omega_F| = \frac{|\Omega_H|}{\mathcal{W}!} = \frac{M!}{\mathcal{W}!(M - \mathcal{W})!} = \binom{M}{\mathcal{W}} \tag{2.5}$$

Having defined the scores vectors for each coding scheme by their scores-support; Ω_H, Ω_F and Ω_R, we can establish the probability and statistics to get the first two moments of the weights for each coding scheme (see Supplementary material).

Next, we define, for each scheme, a *modulations vector* $\mathbf{v}_C^1 = \Psi_C[\Phi(\mathcal{R}(1))]$, considering that, for the neuron under consideration, the preferred pattern corresponds to the first input pattern in Λ. For ROC, it depends on a modulation parameter $m \in \{1/n : n \in \mathbb{Z}, n \neq 1\}$, with $\mathbf{v}_R^1 = (m^0, m^1, m^2, ..., m^M)$. For Ranked-NoM, $\Psi_H \equiv \Phi_H$ (2.2), and for the NoM scheme $\Psi_F \equiv \Phi_F$ (2.4).

Finally, we define an integration function — effectively equivalent to the membrane potential — which indicates how well the random scores vector matches the fixed modulations vector.

To formally translate intermediate states (i.e., before the propagation is over), we first define the gate function $G_I : \Xi_C \to \mathbb{R}^M$ which nullifies all components of the modulation vector for ranks beyond I. Then, over the first I inputs, the integration function $S_C(\mathbf{w}, I)$ reads:

$$S_C(\mathbf{w}, I) = \left\langle G_I\left(\mathbf{v}_C^1\right), \Phi_C(\mathbf{w})\right\rangle \tag{2.6}$$

Given that Ranked-NoM and NoM are defined for values $\mathcal{N} < M$, the final potential is obtained when $I = \mathcal{N}$ and we would have intermediate states only for values $I < \mathcal{N}$. For ROC, the final potential is obtained when $I = M$ and we would have intermediate states for all values $I < M$.

2.2. Coding schemes comparison

2.2.1. Comparing discriminability

Since $\mathbf{w}$ is a random vector, then $S_C(\mathbf{w}, I)$ is a random function. Let $S_{C,I}$ denote the corresponding output random variable. Its distribution depends on the coding scheme. We compare the three coding schemes in terms of discriminative power, characterizing its distribution by the difference between its best possible value and its expected values, scaled by its variance.

Definition 2.1. We define discriminability $D_C(I)$ as:

$$D_C(I) = \frac{\max(S_{C,I}) - \mathrm{E}[S_{C,I}]}{\sqrt{\mathrm{Var}[S_{C,I}]}} \tag{2.7}$$

where $I \in \mathbb{Z}$ and takes values for ROC in the interval [1, $\mathcal{M}$] and for Ranked-NoM and NoM coding in the interval [1, $\mathcal{N}$]. This discriminability is also known as the signal-to-noise ratio in other papers (Masquelier, 2018; Masquelier and Kheradpisheh, 2018; Jordan et al., 2021). Given that for values $\mathcal{N} < I < M$, Ranked-NoM and NoM are not defined, we set those values to the final integration corresponding to each scheme.

TABLE 1 Formulas for the maximum value of integration $S_{C,I}$ for each scheme.

	$\max(S_{C,I})$
Ranked-NoM(H)	$\mathcal{W}\mathcal{N}\left(\frac{\mathcal{N}+1}{2}\right)+\frac{\mathcal{N}(1-\mathcal{N}^2)}{6}$
NofM(F)	$\mathcal{N}$
ROC(R)	$\frac{(1-m)(1+M)-(1-m^{M+1})}{(1-m)^2}$

The $\max(S_{C,I})$ (see Supplementary Sections 2.6.1, 3.6.1, and 4.6.1), for $\mathcal{W} > \mathcal{N}$, are given in Table 1.

The expectation $\mathrm{E}[S_{C,I}]$ and variance $\mathrm{Var}[S_{C,I}]$ of integration at intermediate states of each scheme C depend on the mean μ_{W_C}, variance $\mathrm{Var}W_C$ and covariance $\mathrm{Cov}_C(W_i, W_j)$ of the scores for the corresponding coding scheme C (see Supplementary Sections for Ranked-NoM 2.7.1, 2.7.7, for NoM 3.7.1, 3.7.2, and for ROC 4.7.1, 4.7.2). Their full expressions are given in Table 2.

As a general pattern, we have the following non-linear functions,

$$\mathrm{E}[S_{C,I}] = \lambda_C\, \mu_{W_C} \tag{2.8}$$

$$\mathrm{Var}[S_{C,I}] = \alpha_C\, \mathrm{Var}W_C + \beta_C\, \mathrm{Cov}_C(W_i, W_j) \tag{2.9}$$

where the constants λ_C, α_C and β_C for each scheme are provided in Table 3.

2.2.2. Behavior of discriminability for final potential

Having established the complete expression of discriminability for the three schemes, we can now compare how they perform.

We first illustrate how the total number of available inputs (M) affects discriminability (Figure 3).

Setting $\mathcal{N} = \mathcal{W} = M/2$ for Ranked-NoM and NoM codes, we get the same function for both schemes (see Supplementary Sections 2.8 and 3.8):

$$D_H(M) = \sqrt{M-1} \tag{2.10}$$

For ROC, we found (see Supplementary Section 4.8):

$$\lim_{M\to\infty} D_F(M) = \frac{\sqrt{3}}{1-m}\sqrt{1-m^2} \tag{2.11}$$

For $m = 0.8$, the function $Y = D_F(M)$ has a horizontal asymptote in $Y \simeq 5.2$:

$$\lim_{M\to\infty} D_F(M) = \frac{\sqrt{3}}{1-0.8}\sqrt{1-0.8^2} \simeq 5.2 \tag{2.12}$$

In light of these behaviors, we propose that Ranked-NoM and NoM are to be preferred over ROC.

2.2.3. Behavior of discriminability during propagation

We now contrast, for a given $M = 31$, how discriminability increases as more and more inputs become available (namely, potential integration, Figure 4).

As shown above, discriminability saturates to the same value for Ranked-NoM and NoM (here, $\mathcal{N} = \mathcal{W}$), while, for ROC, it saturates at a lower value, which depends on the ROC-parameter m (here $m = 0.8$).

We also observe that NoM performs poorly early on since discriminability increases nearly linearly, while both ROC and Ranked-NoM increase more like an exponential relaxation to the final value.

In contrast to NoM, Ranked-NoM Coding then displays a much faster increase in discriminability in the early phase of input integration and reaches a higher value than ROC.

In this regard, Ranked-NoM displays the best performance, with a high discriminability for the very early inputs.

2.2.4. Exploring the speed-accuracy trade-off through simulations

Importantly, our discriminability measure (Equation 2.7) is based on the unconstrained membrane potential, i.e., ignoring the threshold. But of course, in a real scenario, a threshold is needed, especially for neurons in the hidden layers (otherwise, they will not fire!). When choosing a threshold, a high value:

- Ensures that the probability of reaching it with random input (which may be seen as a false alarm, FA) is low.
- Causes a longer latency even when the preferred pattern is given as input.

Conversely, a low threshold does the opposite (shorter latency but higher FA rate). This can be seen as a speed-accuracy trade-off.

We explored this trade-off through numerical simulations. We fixed $M = 20$ and estimated the false alarm probabilities for ROC ($m = 0.8$), R-NoM ($\mathcal{W} = \mathcal{N} = 10$), and NoM (also $\mathcal{W} = \mathcal{N} = 10$), as a function of the threshold, using 2.10^5 random input spike patterns. In Figure 5, we plotted those probabilities as a function of the latency (expressed in input spike number, not in seconds) for the firing response to the preferred pattern (latency which in turn depends on the threshold). This plot confirms the supremacy of R-NoM, especially in the early stage of the response, in agreement with Figure 4. For example, here the preferred pattern has $\mathcal{N} = 10$ spikes. Let's say we want the receiver neuron to fire as soon as the fifth input spike is received. For R-NoM, this means the threshold should be in the $[294, 330]$ range. Choosing 330 will minimize the FA rate, which will be around 3.10^{-4}. For ROC, the corresponding threshold would be 28.36,

TABLE 2 Formulas for the expectation, variance and covariance of the scores random variable W for each scheme.

C	μ_{W_C}	$\mathrm{Var}W_C$	$\mathrm{Cov}_C(W_i, W_j)$
Ranked-NoM(H)	$\frac{\mathcal{W}(\mathcal{W}+1)}{2M}$	$\mu_{W_H}\left(\frac{2\mathcal{W}+1}{3}-\mu_{W_H}\right)$	$\frac{\mu_{W_H}}{M-1}\left(\mu_{W_H}-\frac{2\mathcal{W}+1}{3}\right)$
NofM(F)	$\frac{\mathcal{W}}{M}$	$\mu_{W_F}(1-\mu_{W_F})$	$\mu_{W_F}\left(\frac{\mathcal{W}-1}{M-1}-\mu_{W_F}\right)$
ROC(R)	$\frac{M+1}{2}$	$\mu_{W_R}\left(\frac{M-1}{6}\right)$	$\frac{\mu_{W_R}}{M-1}\left(\mu_{W_R}-\frac{2M+1}{3}\right)$

TABLE 3 Formulas for the expectation and variance coefficients of the different Integration schemes.

C	λ_C	α_C	β_C
Ranked-NoM(H)	$\frac{I(2\mathcal{N}-I+1)}{2}$	$\mathcal{N}\,I(\mathcal{N}-I+1)+\frac{I(I-1)(2I-1)}{6}$	$\mathcal{N}\,I(I-1)(\mathcal{N}-I+1)+\frac{I^2(I-1)^2}{4}-\frac{I(I-1)(2I-1)}{6}$
NofM(F)	I	I	$I\,(I-1)$
ROC(R)	$\frac{1-m^I}{1-m}$	$\frac{1-m^{2I}}{1-m^2}$	$\left(\frac{1-m^I}{1-m}\right)^2-\frac{1-m^{2I}}{1-m^2}$

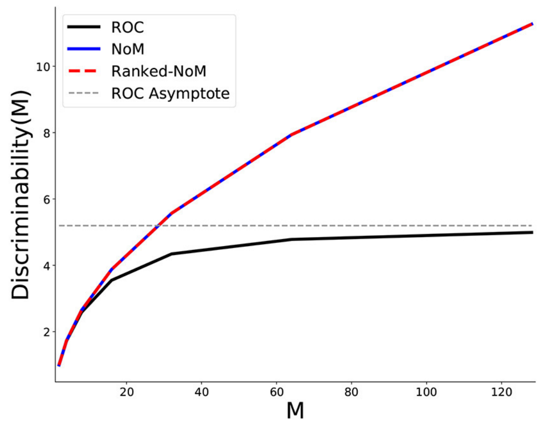


FIGURE 3
Behavior of the maximal discriminability as a function of the total number of inputs M. NoM and Ranked-NoM (set to $\mathcal{W}=\mathcal{N}=M/2$) converge to the same maximal values and so the two curves overlap. For these two schemes, the maximal discriminability scales as $D_H(M)=D_F(M)=\sqrt{M-1}$ (proof to be found in Supplementary material). Hence, discriminability is not limited, and adding inputs will always improve it. By contrast, maximal discriminability for ROC saturates at an asymptotic value ($Y\simeq 5.2$ for $m=0.8$ here).

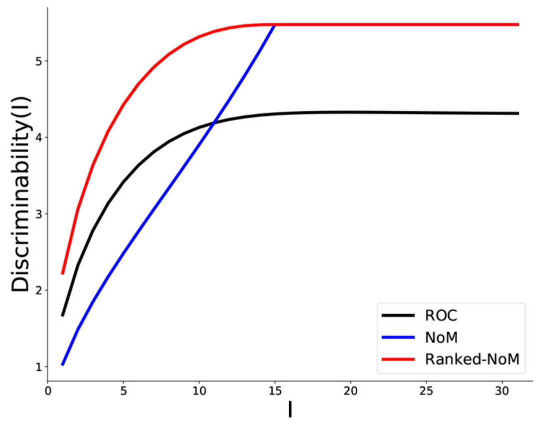


FIGURE 4
Comparison of discriminability for the three coding schemes during propagation. The Discriminability, $D_C(I)$ (Equation 2.7) is reported as the number of inputs I builds up, for each coding scheme: ROC (black), NoM (blue), and Ranked-NoM (red). For ROC coding, the inputs I accumulate up to the maximal number (here, $M=31$) while, in the two others, propagation stops beyond $\mathcal{N}=15$ (in this case, we retain the value $D_C(\mathcal{N})$ for later values).

leading to a much higher FA rate of 0.1. Finally, for NoM, the threshold would be 5, and the FA rate 0.7, which would be totally unacceptable!

Here again, our attempt to speculate upon how to combine computation-power of float-based TTFSs schemes and power-saving integer-based TTFS schemes offers a promising avenue: FA rate could be cut by a factor of three orders of magnitude compared with the former, and four orders compared with the latter.

3. Discussion

In this paper, we presented a new mathematical framework which allows unifying various TTFS codes. This framework introduces the concept of modulation: a decreasing function such that the earliest input spikes matter more. This broad definition of modulation encompasses previous proposals (ROC, NoM) as well as new ones. The activation is maximal when the spikes arrive in the order of the weights: the first spike should arrive through the strongest weight, and so on. This

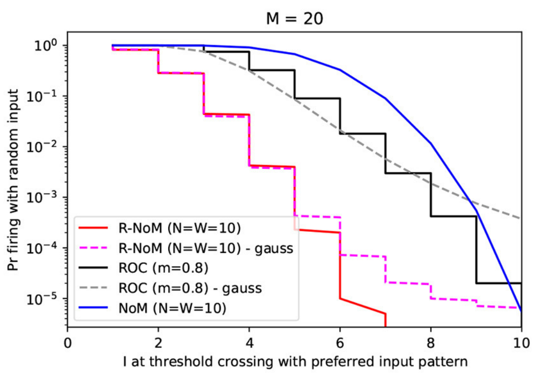


FIGURE 5
Speed-accuracy trade-off. Here, we represented, for the different codes, the probability of firing to random inputs as a function of *I* at threshold crossing for the preferred pattern (both variables depend on the threshold which is varied, but not represented on this figure). Solid lines indicate the probabilities estimated by Monte Carlo simulations (n=2.10^5 runs for each value). Dashed lines indicate the probability assuming a Gaussian distribution for the potential, with the mean and variance computed from Table 3 formulas. These Gauss-based values match well the simulations for low latencies. For higher latencies, they tend to overestimate the FA rate. This suggests that the potential is roughly normally distributed everywhere except in the right tail of the distribution.

defines the preferred input spike pattern of a neuron. Then, we defined discriminability, which measures how much more the neuron responds to its preferred pattern than to random inputs. Our framework allows us to compute this discriminability analytically. Thus various TTFS codes can easily be compared in terms of discriminability. The framework also allows the design of new codes that maximize this discriminability. In particular, we propose a new code that we dubbed "Ranked-NoM" (R-NoM), which makes use of integer modulation and weights that both decrease linearly. We demonstrated that R-NoM has much more discriminative power than ROC and NoM, especially in the early phase of the response, which is already very selective. Thus it allows detectors that are both accurate and reactive. In addition, the fact that R-NoM uses only integers makes it much more hardware-friendly than ROC, and the geometric modulation suggested in Furber et al. (2007).

There are however situations where NoM coding can be particularly interesting for hardware implementations. The advantages of R-NoM coding described here apply in situations where incoming spikes are processed one by one. However, in some designs, it is possible to process spikes as a packet. For example, you could define an input array with M bits that are initially all set to zero. As spikes come in, the corresponding input lines can be flipped on until a fixed number of bits (N) are set to one. At this point, it is easy to determine the level of activation of a target neuron by performing a logical AND operation between the array of input spikes and a second array of bits corresponding to the connected weights. Counting the number of "hits" and comparing the result to the neuron's threshold can be done in a single clock cycle with specialized FPGA or ASIC hardware. Similar results can be obtained using memristor-based cross-bar arrays.

That said, the current analysis provides a strong argument for using implementations that process incoming spikes in order since it is the only way to take advantage of the remarkable early discriminative power of R-NoM coding. Such an approach goes a long way toward ensuring that computations can be done with the minimum number of spiking events.

One important issue that we did not address in this paper is learning. We plan to address it in future work. Only then we will be able to confront the different coding schemes with real-world data (e.g., CIFAR, ImageNet, Google Speech Commands) and compare their performance, possibly using the methodology of Guo et al. (2021). For unsupervised learning, we think that the STDP-like learning rule that we proposed in Thorpe et al. (2019) could be adapted for the integer, non-binary, weights that are required for R-NoM. In short, part of the weights from unused synapses could be moved to used but not saturated synapses. For supervised learning, backpropagation has already been adapted to TTFS codes (Mostafa, 2017; Zhou et al., 2019; Kheradpisheh and Masquelier, 2020; Park et al., 2020; Sakemi et al., 2020; Zhang et al., 2020; Comsa et al., 2021; Mirsadeghi et al., 2021). Yet none of these approaches included the concept of a spike-based decreasing modulation. We will explore that possibility in future work.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary material, further inquiries can be directed to the corresponding author/s.

Author contributions

ST and TM designed the project. LB and JG performed the mathematical derivations. JG and TM did the numerical simulations. All authors wrote the paper. All authors contributed to the article and approved the submitted version.

Funding

The authors gratefully acknowledge financial support from the Colombian non-profit foundation COLFUTURO and the partial funding of Le Centre de Recherche Cerveau et Cognition (CerCo).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnins.2022.971937/full#supplementary-material

References

Abderrahmane, N., Lemaire, E., and Miramond, B. (2020). Design space exploration of hardware spiking neurons for embedded artificial intelligence. *Neural Netw*. 121, 366–386. doi: 10.1016/j.neunet.2019.09.024

Adrian, E. D. (1928). *The Basis of Sensation*. New York, NY: Christophers.

Comsa, I.-M., Potempa, K., Versari, L., Fischbacher, T., Gesmundo, A., and Alakuijala, J. (2021). Temporal coding in spiking neural networks with alpha synaptic function: learning with backpropagation. *IEEE Trans. Neural Netw. Learn. Syst*. 1–14. doi: 10.1109/TNNLS.2021.3071976

Davies, M., Srinivasa, N., Lin, T. H., Chinya, G., Joshi, P., Lines, A., et al. (2018). Loihi: a neuromorphic manycore processor with on-chip learning. *IEEE Micro* 38, 82–99. doi: 10.1109/MM.2018.112130359

Frenkel, C. (2021). Sparsity provides a competitive advantage. *Nat. Mach. Intell*. 3, 742–743. doi: 10.1038/s42256-021-00387-y

Furber, S. B., Brown, G., Bose, J., Cumpstey, J. M., Marshall, P., and Shapiro, J. L. (2007). Sparse distributed memory using rank-order neural codes. *IEEE Trans. Neural Netw*. 18, 648–659. doi: 10.1109/TNN.2006.890804

Furber, S. B., John Bainbridge, W., Mike Cumpstey, J., and Temple, S. (2004). Sparse distributed memory using N-of-M codes. *Neural Netw*. 17, 1437–1451. doi: 10.1016/j.neunet.2004.07.003

Gautrais, J., and Thorpe, S. (1998). Rate coding versus temporal order coding: a theoretical approach. *Biosystems* 48, 57–65. doi: 10.1016/S0303-2647(98)00050-1

Gollisch, T., and Meister, M. (2008). Rapid neural coding in the retina with relative spike latencies. *Science* 319, 1108–1111. doi: 10.1126/science.1149639

Göltz, J., Kriener, L., Baumbach, A., Billaudelle, S., Breitwieser, O., Cramer, B., et al. (2021). Fast and energy-efficient neuromorphic deep learning with first-spike times. *Nat. Mach. Intell*. 3, 823–835. doi: 10.1038/s42256-021-00388-x

Guo, W., Fouda, M. E., Eltawil, A. M., and Salama, K. N. (2021). Neural coding in spiking neural networks: a comparative study for robust neuromorphic systems. *Front. Neurosci*. 15, 638474. doi: 10.3389/fnins.2021.638474

Jordan, J., Schmidt, M., Senn, W., and Petrovici, M. A. (2021). Evolving interpretable plasticity for spiking networks. *eLife* 10, 1–33. doi: 10.7554/eLife.66273

Kheradpisheh, S. R., and Masquelier, T. (2020). Temporal backpropagation for spiking neural networks with one spike per neuron. *Int. J. Neural Syst*. 30, 2050027. doi: 10.1142/S0129065720500276

Liang, M., Zhang, J., and Chen, H. (2021). "A 1.13μJ/classification spiking neural network accelerator with a single-spike neuron model and sparse weights," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)* (Daegu: IEEE), 1–5.

Masquelier, T. (2018). STDP allows close-to-optimal spatiotemporal spike pattern detection by single coincidence detector neurons. *Neuroscience* 389, 133–140. doi: 10.1016/j.neuroscience.2017.06.032

Masquelier, T., and Kheradpisheh, S. R. (2018). Optimal localist and distributed coding of spatiotemporal spike patterns through STDP and coincidence detection. *Front. Comput. Neurosci*. 12, 74. doi: 10.3389/fncom.2018.00074

Mead, C. (1990). Neuromorphic electronic systems. *Proc. IEEE* 78, 1629–1636. doi: 10.1109/5.58356

Mirsadeghi, M., Shalchian, M., Kheradpisheh, S. R., and Masquelier, T. (2021). STiDi-BP: spike time displacement based error backpropagation in multilayer spiking neural networks. *Neurocomputing* 427, 131–140. doi: 10.1016/j.neucom.2020.11.052

Mostafa, H. (2017). Supervised learning based on temporal coding in spiking neural networks. *IEEE Trans. Neural Netw. Learn. Syst*. 29, 1–9. doi: 10.1109/TNNLS.2017.2726060

Nair, H., Shen, J. P., and Smith, J. E. (2020). Direct CMOS implementation of neuromorphic temporal neural networks for sensory processing. *arXiv preprint arXiv:2009.00457*.

Oh, S., Kwon, D., Yeom, G., Kang, W.-M., Lee, S., Woo, S. Y., et al. (2022). Hardware implementation of spiking neural networks using time-to-first-spike encoding. *IEEE Access*. 10, 24444–24455. doi: 10.1109/access.2022.3149577

Park, S., Kim, S., Na, B., and Yoon, S. (2020). T2FSNN: deep spiking neural networks with time-to-first-spike coding. *arXiv preprint arXiv:2003.11741*. doi: 10.1109/DAC18072.2020.9218689

Rueckauer, B., and Liu, S.-C. (2018). "Conversion of analog to spiking neural networks using sparse temporal coding," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)* (Florence: IEEE), 1–5. doi: 10.1109/ISCAS.2018.8351295

Sakemi, Y., Morino, K., Morie, T., and Aihara, K. (2020). A supervised learning algorithm for multilayer spiking neural networks based on temporal coding toward energy-efficient VLSI processor design. *arXiv preprint arXiv:2001.05348*. doi: 10.1109/TNNLS.2021.3095068

Srivatsa, P., Timothy, K., Chu, N., Tavva, Y., and Wu, J. (2020). You only spike once: improving energy-efficient neuromorphic inference to ANN-level accuracy. *arXiv*. doi: 10.48550/ARXIV.2006.09982

Thorpe, S. J., and Gautrais, J. (1998). "Rank order coding," in *Computational Neuroscience: Trends in Research*, ed J. M. Bower (New York, NY: Plenum Press), 113–118. doi: 10.1007/978-1-4615-4831-7_19

Thorpe, S. J., and Imbert, M. (1989). "Biological constraints on connectionist modeling," in *Connectionism in Perspective*, eds R. Pfeifer, Z. Schreter, F. Fogelman-Soulie, and L. Steels (Amsterdam: Elsevier), 63–92.

Thorpe, S. J., Masquelier, T., Martin, J., Yousefzadeh, A., and Linares-Barranco, B. (2019). Method, digital electronic circuit and system for unsupervised detection of repeating patterns in a series of events. US20190286944A1.

Zhang, M., Wang, J., Amornpaisannon, B., Zhang, Z., Miriyala, V., Belatreche, A., et al. (2020). Rectified linear postsynaptic potential function for backpropagation in deep spiking neural networks. *arXiv preprint arXiv:2003.11837v2*.

Zhou, S., LI, X., Chen, Y., Chandrasekaran, S. T., and Sanyal, A. (2019). Temporal-coded deep spiking neural network with easy training and robust performance. *arXiv preprint arXiv:1909.10837*. doi: 10.1109/TNNLS.2021.3110991

Supplementary Material

Here, we provide the full details of the theoretical mathematical framework of the temporal coding schemes. This Theoretical development is based upon multivariate calculus and probability theory. In the first section (section 1), we build the general theory by defining a random experiment in which we transform the possible input orders into a probabilized weights support. This set is then a base to build a scores support for each scheme. In this section, we define how the scores support translate into integration through modulation. As our goal is to compare the codes' discriminability power, we then establish the theory of the latter, in order to find general formulas for the maximal value of integration, its mean and variance.

Then, we apply this general theory to Ranked-NoM Coding (section 2), NoM Coding (section 3) and ROC Coding (section 4).

We provide, as another SM, a Jupyter-Notebook that numerically validates all results.

1 GENERAL FRAMEWORK

1.1 General principles

Let us consider an input layer made of M spiking neurones, denoted $I_0, .., I_{(M-1)}$, which send spikes to an efferent neurone.

For a given stationary stimulus, each of the M input neurones emits one spike, and the order in which spikes are emitted depends on the stimulus. We encode the order of the input spikes by the vector corresponding to the order of neurones' indices. For instance, the order $(I_0, .., I_{(M-1)})$ will translate into $(0, 1, \dots, M-1)$, and the inverse order will translate into $(M-1, M-2, \dots, 1, 0)$.

This input order is transformed into a vector of weights.

Depending on the scheme, the vector of weights is used to build a vector of scores.

The response of an efferent neurone is then built as a sum of the scores multiplied by a modulation which is specific for each input neurone.

1.2 Random Experiment

We consider the order generated by the M input neurones as a random sequence, where all input orders have the same probability.

1.2.1 Input orders

Let Λ denote the ascending lexically ordered set of the possible permutations over the set $\mathcal{M} = \{0, \dots, M-1\}$: $\Lambda = \{\mathbf{r}^1, ..., \mathbf{r}^{M!}\}$ where $\mathbf{r} \in \mathbb{R}^M$ and $r_i^j \in \mathcal{M}$. To map a rank into an input order in Λ, we define the application $\mathcal{R} : \{1, 2, .., M!\} \rightarrow \Lambda$ such that $\mathcal{R}(k) = \mathbf{r}^k$.

We consider the discrete random variable K, defined as:

$$K = \begin{cases} \mathcal{D}_K = \{1, 2, .., M!\} \\ \mathrm{P}_K(k) = \frac{1}{M!} \end{cases}$$

Building upon K, we can then consider $\mathbf{X} = \mathcal{R}(K) = (X_1, X_2, \dots, X_M)$ [1] as a random vector or multivariate random variable with support the ordered set Λ:

$$\Lambda_{X_1X_2\dots X_M} = \left\{ \begin{bmatrix} 0 \\ 1 \\ \vdots \\ M-1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ M-1 \end{bmatrix} \dots \begin{bmatrix} M-1 \\ \vdots \\ 1 \\ 0 \end{bmatrix} \right\} \tag{1.1}$$

By construction, it is a vector of size M, and each component X_i is a random variable defined on the same sample space obeying:

$$X_i = \begin{cases} \mathcal{D}_{X_i} = \{0, \dots, M-1\} \\ \\ \mathrm{P}_{X_i}(r_i) = \frac{1}{M} \end{cases}$$

Let its multivariate joint probability mass function be

$$P_{X_1X_2\dots X_M} = \begin{cases} \dfrac{1}{M!} & \mathbf{r}^k \in \Lambda_{X_1X_2\dots X_M} \\ 0 & \text{Otherwise} \end{cases} \tag{1.2}$$

$X_1, X_2, \dots, X_M$ are identically distributed, hence they have the same expectation and variance; they are not independent given that:

$$P_{X_1}(r)P_{X_2}(r)\dots P_{X_M}(r) = \left(\frac{1}{M}\right)^M \neq \frac{1}{M!} = P_{X_1X_2\dots X_M} \tag{1.3}$$

thus $\mathrm{Cov}(X_i, X_j) \neq 0$.

1.2.2 From input order to weights

We define the affine transformation Φ mapping vectors from Λ (input orders) to the set Ω (weights):

$$\begin{aligned} \Phi : \quad \Lambda \subset \mathbb{R}^M &\longrightarrow \quad \Omega \subset \mathbb{R}^M \\ (r_1^k, \dots, r_M^k) &\longrightarrow (\hat{w}_1^k, \dots, \hat{w}_M^k) \\ \mathbf{r}^k &\longrightarrow \quad \hat{\mathbf{w}}^k \end{aligned} \tag{1.4}$$

Each vectorial component is given by

$$\phi_i(\mathbf{r}^k) = \phi_i(r_1^k, \dots, r_M^k) = M - r_i^k = \hat{w}_i^k \tag{1.5}$$

so we have:

$$\Phi(\mathbf{r}^k) = \Phi(r_1^k, \dots, r_M^k) \tag{1.6}$$
$$= (\phi_1(r_1^k, \dots, r_M^k), \phi_2(r_1^k, \dots, r_M^k), \dots, \phi_M(r_1^k, \dots, r_M^k)) \tag{1.7}$$
$$= (M - r_1^k, M - r_2^k, \dots, M - r_M^k) \tag{1.8}$$
$$= (\hat{w}_1^k, \dots, \hat{w}_M^k) = \hat{\mathbf{w}}^k \tag{1.9}$$

Consider $X_1, \dots, X_M$, a set of random variables with joint probability distribution function $P_{X_1\dots X_M}(x_1, \dots, x_M)$ and support Λ. Let us denote $\hat{W}_i^k = \phi_i(X_1, \dots, X_M)$ for $i = 1, \dots, M$. We

[1] Throughout the article, we will use rows or columns interchangeably to denote random variables and define vector-value functions.

transform the random variables X_i (input order) into the new random variables $\hat{W}_i$ (weights) by the affine transformation Φ.

Since the multivariate transformation Φ is one to one, the transformation is invertible and can be solved for the equations $r_i^k = \phi_i^{-1}(\hat{w}_1^k, \ldots, \hat{w}_M^k)$ for $i = 1, \ldots, M$.

The Jacobian of this multivariate transformation is

$$J\Phi = \begin{vmatrix} \nabla\phi_1 \\ \nabla\phi_2 \\ \vdots \\ \nabla\phi_M \end{vmatrix} = \begin{vmatrix} \dfrac{\partial\phi_1}{\partial r_1^k} & \dfrac{\partial\phi_1}{\partial r_2^k} & \cdots & \dfrac{\partial\phi_1}{\partial r_M^k} \\ \dfrac{\partial\phi_2}{\partial r_1^k} & \dfrac{\partial\phi_2}{\partial r_2^k} & \cdots & \dfrac{\partial\phi_2}{\partial r_M^k} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial\phi_M}{\partial r_1^k} & \dfrac{\partial\phi_M}{\partial r_2^k} & \cdots & \dfrac{\partial\phi_M}{\partial r_M^k} \end{vmatrix} = \begin{vmatrix} -1 & 0 & \ldots & 0 \\ 0 & -1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & -1 \end{vmatrix} = -1 \tag{1.10}$$

Let $|J\Phi|$ denote the absolute value of the determinant J . Then the joint pdf of $\hat{W}_1^k, \ldots, \hat{W}_M^k$ is

$$P_{\hat{W}_1 \ldots \hat{W}_M}(\hat{w}_1^k, \ldots, \hat{w}_M^k) = P_{X_1 \ldots X_M}(\phi_1^{-1}(\hat{w}_1^k), \ldots, \phi_M^{-1}(\hat{w}_1^k))|J\Phi| \tag{1.11}$$

The inverse for $i = 1, \ldots, M$ is given by

$$\phi_i^{-1} = M - w_i^k = r_i^k \tag{1.12}$$

Therefore, $\hat{\mathbf{W}} = (\hat{W}_1, \hat{W}_2 \ldots, \hat{W}_M)$ is a discrete random vector with weights-support the ordered set $\Omega = \{\hat{\mathbf{w}}_1^k, \ldots, \hat{\mathbf{w}}_{M!}^k\}$

$$\Omega_{\hat{W}_1 \hat{W}_2 \ldots \hat{W}_M} = \left\{ \begin{bmatrix} M \\ M-1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} M-1 \\ M \\ \vdots \\ 1 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ \vdots \\ M-1 \\ M \end{bmatrix} \right\} \tag{1.13}$$

and its joint pdf is given by

$$\begin{aligned} P_{\hat{W}_1 \ldots \hat{W}_M}(\hat{w}_1^k, \ldots, \hat{w}_M^k) &= P_{X_1 \ldots X_M}(\phi_1^{-1}(\hat{w}_1^k), \ldots, \phi_M^{-1}(\hat{w}_1^k))|J\Phi| \\ &= P_{X_1 \ldots X_M}(\phi_1^{-1}(\hat{w}_1^k), \ldots, \phi_M^{-1}(\hat{w}_1^k)) \\ &= P_{X_1 \ldots X_M}(M - w_1^k, \ldots, M - w_M^k) \\ &= \frac{1}{M!} \end{aligned} \tag{1.14}$$

In order to find the *marginal probability mass function* of $\hat{W}_1, \hat{W}_2, \ldots, \hat{W}_M$, we start from the joint probability distribution of $\hat{W}_i$, and proceed by summation (see Fig. S1).

Thus, the *marginal probability mass function* of $\hat{W}_1, \hat{W}_2, \ldots, \hat{W}_M$ defined on the same sample space $\mathcal{D}_{X_i} = \{M, M-1, ..., 1\}$ for $i = 1, ..., M$, with joint probability distribution defined in the equation (1.14), and for equation (1.17) is given by,

$$P_{\hat{W}_i}(w) = \begin{cases} \dfrac{(M-1)!}{M!} = \dfrac{1}{M} & w = 1, 2, \ldots M \\ 0 & \text{Otherwise} \end{cases} \tag{1.18}$$

Let $Y_1, Y_2, \ldots, Y_n$ be discrete random variables defined on the same sample space $\mathcal{D}_{Y_j} = \{y_1, y_2, \ldots, y_m\}$ and $P_{Y_1 Y_2 \ldots Y_n}$ be the multivariate probability distribution of the random variables Y_j, then we have:

$$P_{Y_1}(y_1) = \sum_{y_{i_n} \in \mathcal{D}_{Y_n}} \cdots \sum_{y_{i_2} \in \mathcal{D}_{Y_2}} P_{Y_1 Y_2 \ldots Y_n}(y_1, y_{i_2}, \ldots, y_{i_n}) \tag{1.15}$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$P_{Y_1}(y_m) = \sum_{y_{i_n} \in \mathcal{D}_{Y_n}} \cdots \sum_{y_{i_2} \in \mathcal{D}_{Y_2}} P_{Y_1 Y_2 \ldots Y_n}(y_m, y_{i_2}, \ldots, y_{i_n}) \tag{1.16}$$

and the same way, we can compute the probability of each element of the sample space of $Y_2, \ldots, Y_n$. The marginal probability function of βth random variable Y_β of γth output y_γ, $P_{Y_\beta}(y_\gamma)$, is given by:

$$P_{Y_\beta}(y_\gamma) = \sum_{y_{i_n} \in \mathcal{D}_{Y_n}} \cdots \sum_{\substack{y_{i_k} \in \mathcal{D}_{Y_k} \\ k \neq \beta}} \cdots \sum_{y_{i_1} \in \mathcal{D}_{Y_1}} P_{Y_1 \ldots Y_\beta \ldots Y_n}(y_{i_1}, \ldots, y_\gamma, \ldots, y_{i_n}) \tag{1.17}$$

Figure S1. Marginal probability mass function from joint probability distribution

1.3 Coding theory-definitions

A coding scheme is defined by how weights are transformed to scores, and how these scores are integrated with a modulation function.

1.3.1 From weights to scores

To attribute scores to the weights for a given coding scheme, we define the *score vector*, $\mathbf{w}_C^k = (w_1^k, \ldots, w_M^k) \in \mathbb{R}^M$, by the following function Φ_C:

Definition 1.1. We define the vector-valued function,

$$\begin{array}{cccc} \Phi_C : & \Omega \subset \mathbb{R}^M & \longrightarrow & \Omega_C \subset \mathbb{R}^M \\ & (\hat{w}_1^k, \ldots, \hat{w}_M^k) & \longrightarrow & (w_1^l, \ldots, w_M^l) \\ & \hat{\mathbf{w}}^k & \longrightarrow & \mathbf{w}_C^l \end{array} \tag{1.19}$$

such that:

$$\begin{aligned} \Phi_C(\hat{\mathbf{w}}^k) &= \Phi_C(\hat{w}_1^k, \ldots, \hat{w}_M^k) && (1.20)\\ &= (\phi_{C,1}(\hat{w}_1^k, \ldots, \hat{w}_M^k), \ldots, \phi_{C,M}(\hat{w}_1^k, \ldots, \hat{w}_M^k)) && (1.21)\\ &= (w_1^l, \ldots, w_M^l) = \mathbf{w}_C^l && (1.22) \end{aligned}$$

where $l \in \{1, \ldots, |\Omega_C|\}$ and $\phi_{C,i}$ are real functions of several variables, $\phi_{C,i} : \Omega \to \mathbb{R}$ that are defined depending on the coding scheme.

Therefore, let $\mathbf{W} = (W_1, W_2 \ldots, W_M)$ be a discrete random vector with support the ordered set $\Omega_C = \{\mathbf{w}_1^l, \ldots, \mathbf{w}_{|\Omega_C|}^l\}$ where $\mathbf{w}_C^l$ are the scores of scheme C.

Being built upon $\mathbf{W}$, the components W_i of $\mathbf{W}$ are also identically distributed hence they have the same expectation and variance. We will show below they are not independent, and will establish their covariance by *tree method*, one for each scheme (2.32)(3.30)(4.26).

We denote Ω_C the set of possible values for $\mathbf{w}_C^k$, that is, the scores-support and its cardinality $|\Omega_C|$ depends on the coding scheme.

1.3.2 Modulations vector function

In order to sort out among stimuli by output neurones, vectors of modulation values must be defined with the same size as $\mathbf{w}_C^k$. In the same way as we defined $\mathbf{w}^k$, modulation vectors $\mathbf{v}^l$ can be defined by a modulation function following:

Definition 1.2. We define the application,

$$\begin{array}{cccc} \Psi_C : & \Omega \subset \mathbb{R}^M & \longrightarrow & \Xi_C \subset \mathbb{R}^M \\ & (\hat{w}_1^k, \ldots, \hat{w}_M^k) & \longrightarrow & (v_1^l, \ldots, v_M^l) \\ & \hat{\mathbf{w}}^k & \longrightarrow & \mathbf{v}_C^l \end{array} \tag{1.23}$$

such that:

$$\begin{aligned} \Psi_C(\hat{\mathbf{w}}^k) &= \Psi_C(\hat{w}_1^k, \ldots, \hat{w}_M^k) && (1.24)\\ &= (\psi_{C,1}(\hat{w}_1^k, \ldots, \hat{w}_M^k), \ldots, \psi_{C,M}(\hat{w}_1^k, \ldots, \hat{w}_M^k)) && (1.25)\\ &= (v_1^l, \ldots, v_M^l) = \mathbf{v}_C^l && (1.26) \end{aligned}$$

where $l \in \{1, \ldots, |\Xi|\}$ and $\psi_{C,i}$ are real functions of several variables, $\psi_{C,i} : \Omega \to \mathbb{R}$ that depend upon the coding scheme.

We denote Ξ_C the set of possible values for $\mathbf{v}_C^k$ and its cardinality depends on the coding scheme through Ψ_C.

The modulation vector determines which input order is preferred by the efferent neurone under consideration. For the sake of clarity, we will consider from now on only the output neurone for which the preferred stimulus (input order) is the one corresponding to $\mathbf{r}^1$. Hence, we will use

$$\mathbf{v}_C^1 = \Psi_C(\hat{\mathbf{w}}^1) = \Psi_C(\Phi(\mathbf{r}^1)) \tag{1.27}$$

as the modulation vector.

1.3.3 Integration function $S_C(\mathbf{w}_C^k, I)$

To compare how *scores vector* matches well with *modulations vector*, we define an *integration function* following:

Definition 1.3. We define the application $S_C(\mathbf{w}_C^k, I) : \Omega_C \times \{0, \ldots, M\} \to \mathbb{R}$ such as:

$S_C(\mathbf{w}_C^k, I)$ is the inner product of *modulations vector* $\mathbf{v}_C^1$ (set for best matching $\hat{\mathbf{w}}^1$ for the coding scheme C) by *score vector* $\mathbf{w}_C^k$, over the I first components of vectors.

In order to formally translate intermediate states, we first define the gate function $G_I : \Xi_C \to \mathbb{R}^M$ which nullifies all components of the modulation vector for ranks beyond I.

Then, the integration function reads:

$$S_C(\mathbf{w}_C^k, I) = \left\langle G_I\left(\Psi_C(\hat{\mathbf{w}}^1)\right), \mathbf{w}_C^k \right\rangle \tag{1.28}$$

where $G_I\left(\Psi_C(\hat{\mathbf{w}}^1)\right) = G_I\left(\mathbf{v}_C^1\right) = \mathbf{v}_{C,I}^1$. We used bracket notation for inner product.

Building upon the random variable $\mathbf{W}$, we finally define the random variable $S_{C,I} = S_C(\mathbf{W}, I)$:

$$S_{C,I} = \begin{cases} \mathcal{D}_{S_{C,I}} = \{\textit{depends upon } C\} \\ \\ \mathrm{P}_{S_{C,I}} = \{\textit{depends upon } C\} \end{cases}$$

We can then define the best order $\max(S_{C,I})$, expectation $\mathrm{E}[S_{C,I}]$ and variance $\mathrm{Var}[S_{C,I}]$ for each coding scheme.

1.4 Discriminability power measure

The goal is to compare the coding schemes by their power to discriminate among stimuli.

1.4.1 Discriminability

Considering integration $S_{C,I}$ random variable, we define *discriminability* $D_C(I)$ as the difference between its best possible value and its expectation, scaled by its variance :

Definition 1.4.

$$D_C(I) = \frac{\max(S_{C,I}) - \mathrm{E}[S_{C,I}]}{\sqrt{\mathrm{Var}[S_{C,I}]}}, I \in \{1, \ldots, M\} \tag{1.29}$$

where $I \in \{1, \ldots, M\}$ for ROC and $I \in \{1, \ldots, \mathcal{N}\}$ for Ranked-NoM and NoM coding. Given that the Ranked-NoM and NoM schemes are not defined for $\mathcal{N} < I < M$, we use the final integration value.

To compute discriminability for the different coding schemes, we then now turn to the expressions of $\max(S_{C,I})$, $\mathrm{E}[S_{C,I}]$ and $\mathrm{Var}[S_{C,I}]$.

1.4.2 max($S_{C,I}$)

The rearrangement inequality states that

$$\begin{pmatrix} x_n \\ \vdots \\ x_1 \end{pmatrix}^{\mathsf{T}} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \leq \begin{pmatrix} x_{\sigma(1)} \\ \vdots \\ x_{\sigma(n)} \end{pmatrix}^{\mathsf{T}} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \leq \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}^{\mathsf{T}} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \tag{1.30}$$

for every choice of real numbers $x_1 \leq \cdots \leq x_n$ and $y_1 \leq \cdots \leq y_n$ and every permutation $x_{\sigma(1)}, \ldots, x_{\sigma(n)}$ of $x_1, \ldots, x_n$.

Then the lower bound is attained only for the permutation which reverses the order, that is, $\sigma(i) = n - i + 1$ for all $i = 1, \ldots, n$, and the upper bound is attained only for the identity, that is, $\sigma(i) = i$.

From equation (1.28), the rearrangement inequality yields that $\max(S_{C,I})$ is given by:

$$\max(S_{C,I}) =< \mathbf{v}^1_{C,I}, \mathbf{w}^1_C > \tag{1.31}$$

1.4.3 Expectation of $S_{C,I}$

For the sake of clarity, let $\mathbf{v}_{C,I} = \mathbf{v}^1_{C,I} = G_I\left(\Psi_C(\mathbf{r}^1)\right)$ denote the modulation vector, gated up to the first I components.

We have:

$$\begin{aligned} \mathrm{E}[S_{C,I}] &= \mathrm{E}[S_C(\mathbf{W}, I)] \\ &= \mathrm{E}\left[\langle \mathbf{v}_{C,I}, \mathbf{W} \rangle\right] \\ &= \mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathrm{E}\left[\mathbf{W}\right] \end{aligned} \tag{1.32}$$

where the expected value of a random vector is the vector whose elements are the expected values of the respective random variables.

1.4.4 Variance of $S_{C,I}$

Using the same notation $\mathbf{v}_{C,I}$, the variance of $S_C(K, I)$ reads:

$$\begin{aligned} \mathrm{Var}\left[S_{C,I}\right] &= \mathrm{Var}\left[S_C(\mathbf{W}, I)\right] && (1.33) \\ &= \mathrm{Var}\left[\mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathbf{W}\right] && (1.34) \\ &= \mathrm{E}[\mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathbf{W} \cdot \mathbf{W}^{\mathsf{T}} \cdot \mathbf{v}_{C,I}] - \mathrm{E}[\mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathbf{W}] \cdot \mathrm{E}[\mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathbf{W}]^{\mathsf{T}} && (1.35) \\ &= \mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathrm{E}[\mathbf{W} \cdot \mathbf{X}^{\mathsf{T}}] \cdot \mathbf{v}_{C,I} - \mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathrm{E}[\mathbf{W}] \cdot \mathrm{E}[\mathbf{W}]^{\mathsf{T}} \cdot \mathbf{v}_{C,I} && (1.36) \\ &= \mathbf{v}^{\mathsf{T}}_{C,I} \cdot (\mathrm{E}[\mathbf{W} \cdot \mathbf{W}^{\mathsf{T}}] - \mathrm{E}[\mathbf{W}] \cdot \mathrm{E}[\mathbf{W}]^{\mathsf{T}}) \cdot \mathbf{v}_{C,I} && (1.37) \\ &= \mathbf{v}^{\mathsf{T}}_{C,I} \cdot \mathbf{K}_{WW} \cdot \mathbf{v}_{C,I} && (1.38) \end{aligned}$$

where $\mathbf{K}_{WW}$ is the $M \times M$ Variance-Covariance Matrix of $\mathbf{W}$:

$$\mathbf{K}_{WW} = \mathrm{E}[(\mathbf{W} - \mathrm{E}[\mathbf{W}])(\mathbf{W} - \mathrm{E}[\mathbf{W}])^{\mathsf{T}}] \tag{1.39}$$

$$= \mathrm{E}[\mathbf{W}\ \mathbf{W}^{\mathsf{T}}] - \mathrm{E}[\mathbf{W}]\ \mathrm{E}[\mathbf{W}]^{\mathsf{T}}$$

$$= \begin{bmatrix} \mathrm{Cov}(W_1, W_1) & \mathrm{Cov}(W_1, W_2) & \dots & \mathrm{Cov}(W_1, W_M) \\ \mathrm{Cov}(W_2, W_1) & \mathrm{Cov}(W_2, W_2) & \dots & \mathrm{Cov}(W_2, W_M) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}(W_M, W_1) & \mathrm{Cov}(W_M, W_2) & \dots & \mathrm{Cov}(W_M, W_M) \end{bmatrix} \tag{1.40}$$

where covariance $\mathrm{Cov}(W_i, W_j)$ is defined as:

$$\mathrm{Cov}(W_i, W_j) = \mathrm{E}[(W_i - \mathrm{E}[W_i])(W_j - \mathrm{E}[W_j])] \tag{1.41}$$

$$= \sum_{i,j} (w_i - \mu_W)(w_j - \mu_W) f(W_i = w_i, W_j = w_j) \tag{1.42}$$

where $\mu_W = \mathrm{E}[W_i] = \mathrm{E}[W_j]$ and $f(W_i, W_j)$ is the bivariate joint probability distribution.

The diagonal elements are the variance of W_i with $\mathrm{Cov}(W_i, W_i) = \mathrm{E}[(W_i - \mathrm{E}[W_i])^2] = \mathrm{Var}(W_i)$.

Since $\mathbf{W}$ elements are identically distributed, they all have the same variance and the same co-variance, and $\mathrm{Cov}(W_i, W_j) = \mathrm{Cov}(W_j, W_i)$.

$\mathbf{K}_{WW}$ is then symmetric and has only two values, an on-diagonal value $\mathrm{Var}(W_i) \equiv \mathrm{Var}W$, and an off-diagonal value $\mathrm{Cov}(W_i, W_j) \equiv \vartheta$, reading:

$$\mathbf{K}_{WW} = \begin{bmatrix} \mathrm{Var}W & \vartheta & \dots & \vartheta \\ \vartheta & \mathrm{Var}W & \dots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \dots & \mathrm{Var}W \end{bmatrix} \tag{1.43}$$

We denote the variance-covariance matrix for each C scheme as $\mathbf{K}^C_{WW}$.

2 APPLICATION TO RANKED-NOM CODING

2.1 Score vector function

Ranked-NoM Coding is parameterized by the number $\mathcal{W} \in \{1, \dots, M\}$. Under this coding, we want the best weight vector $\mathbf{w}^1$ made up of an arithmetic sequence, over the first $\mathcal{W}$ elements, starting from $\mathcal{W}$, with rate -1, down to zero, and the other elements set to zero.

The scores are obtained by the vector-value function $\Phi_H(\hat{\mathbf{w}}^k) = \mathbf{w}^l_H$ defined in (1.1). Then the vectorial components are given by,

$$\phi_{H,i}(\hat{\mathbf{w}}^k) = \phi_{H,i}(\hat{w}^k_1, \dots, \hat{w}^k_M) = \max(0, \hat{w}^k_i - M + \mathcal{W}) \tag{2.1}$$

such that:

$$\Phi_H(\hat{\mathbf{w}}^k) = \Phi_H(\hat{w}^k_1, \dots, \hat{w}^k_M) \tag{2.2}$$

$$= \Phi_H(M - r^k_1, M - r^k_2, \dots, M - r^k_M) \tag{2.3}$$

$$= (\phi_{H,1}(\hat{w}^k_1, \dots, \hat{w}^k_M), \dots, \phi_{H,M}(\hat{w}^k_1, \dots, \hat{w}^k_M)) \tag{2.4}$$

$$= (\max(0, \mathcal{W} - r^k_1), \dots, \max(0, \mathcal{W} - r^k_M))$$

$$= (w^l_1, \dots, w^l_M) = \mathbf{w}^l_H \tag{2.5}$$

Note that for $k = 1$, we have $\hat{\mathbf{w}}^1 = (M, M-1\ldots,1)$ or equivalently $\mathbf{r}_i^1 = (0,\ldots,M-1)$

$$\Phi_H(\hat{\mathbf{w}}^1) = (\max(0,\mathcal{W}),\ldots,\max(0,\mathcal{W}-M+1)) \tag{2.6}$$
$$= (\mathcal{W},\mathcal{W}-1,\ldots,0) = \mathbf{w}_H^1 \tag{2.7}$$

which contains $M-\mathcal{W}$ zeros.

Example 2.1. As an illustration, let $M = 4$ and $\mathcal{W} = 2$ ($|\Omega| = 24$ permutations). We would have, for the best order:

$$\Phi_H(\hat{\mathbf{w}}^1) = (\max(0,2),\max(0,1),\max(0,0),\max(0,-1)) \tag{2.8}$$
$$= (2,1,0,0) = \mathbf{w}_H^1 \tag{2.9}$$

and for the worst order:

$$\Phi_H(\hat{\mathbf{w}}^{24}) = (\max(0,-1),\max(0,0),\max(0,1),\max(0,2)) \tag{2.10}$$
$$= (0,0,1,2) \tag{2.11}$$

Note that for $\hat{\mathbf{w}}^{18} = (2,1,3,4)$, we would also obtain:

$$\Phi_H(\hat{\mathbf{w}}^{18}) = (\max(0,0),\max(0,-1),\max(0,1),\max(0,2)) \tag{2.12}$$
$$= (0,0,1,2) \tag{2.13}$$

therefore, $\Phi_H(\hat{\mathbf{w}}^{18}) = \Phi_H(\hat{\mathbf{w}}^{24})$.

Indeed, Φ_H maps different weights permutations onto the same score permutation. Hence, a subset of vectors that are pure internal permutations of negative or null values will map to the same element of Ω_H. Since the cardinality of these subsets is the number of permutations of the $M-\mathcal{W}$ null elements, the cardinality of Ω_H is:

$$|\Omega_H| = \frac{M!}{(M-\mathcal{W})!} \tag{2.14}$$

In the illustrative example, $|\Omega_H| = \dfrac{4!}{(4-2)!} = 12$ permutations, and we would have $\Phi_H(\hat{\mathbf{w}}^{18}) = \Phi_H(\hat{\mathbf{w}}^{24}) = \mathbf{w}^{12}$.

2.2 Probability distribution functions for scores

Let $\mathbf{W} = (W_1, W_2, \ldots, W_M)$ be the discrete random vector with support the ordered set $\Omega_H = \{\mathbf{w}^1,\ldots,\mathbf{w}^{\frac{M!}{(M-\mathcal{W})!}}\}$ which is generated by function $\hat{\Phi}_H$ defined in 2.1, and :

$$\Omega_H = \Omega_{W_1W_2\ldots W_M} = \left\{ \begin{bmatrix} \mathcal{W} \\ \mathcal{W}-1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{bmatrix} \mathcal{W}-1 \\ \mathcal{W} \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \ldots \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ \mathcal{W}-1 \\ \mathcal{W} \end{bmatrix} \right\} \tag{2.15}$$

Due to (2.14) let its *joint probability mass function* be

$$P_{W_1 W_2 \dots W_M} = \begin{cases} \dfrac{1}{|\Omega_H|} = \dfrac{(M-\mathcal{W})!}{M!} & \mathbf{w}^k \in \Omega_H \\ 0 & \text{Otherwise} \end{cases} \tag{2.16}$$

and from (1.17), the *marginal probability mass function* of $W_1, W_2, \dots, W_M$ with sample space $\mathcal{D}_{W_i} = \{\mathcal{W}, \mathcal{W}-1, ..., 1, 0\}$ for $i = 1, ..., M$, is given by:

$$P_{W_i}(w=0) = \frac{M-\mathcal{W}}{M} \tag{2.17}$$

$$P_{W_i}(w=k) = \frac{1}{M} \tag{2.18}$$

where $k = 1, ..., \mathcal{W}-1, \mathcal{W}$.

Therefore,

$$W_i = \begin{cases} \mathcal{D}_{W_i} = \{0, 1, ..., \mathcal{W}\} \\ \mathrm{P}_{W_i}(w) = \begin{cases} \dfrac{M-\mathcal{W}}{M} & w = 0 \\ \dfrac{1}{M} & w = 1, 2, \dots \mathcal{W} \end{cases} \end{cases} \tag{2.19}$$

2.3 Expectation vector of scores

2.3.1 $\boldsymbol{\mu_W}$, expected value of $\boldsymbol{W_i}$

$$\mu_W \equiv \mathrm{E}[W_i] = \sum_{k=0}^{M} k \; P_{W_i}(k) = 0\frac{M-\mathcal{W}}{M} + \sum_{k=1}^{\mathcal{W}} k \; \frac{1}{M} \tag{2.20}$$

$$= \frac{1}{M}\sum_{k=1}^{\mathcal{W}} k$$

$$= \frac{1}{M}\frac{\mathcal{W}(\mathcal{W}+1)}{2} \tag{2.21}$$

2.3.2 $\mathrm{E}[\mathbf{W}]$, expectation of $\mathbf{W}$

Therefore,

$$\mathrm{E}[\mathbf{W}] = \begin{bmatrix} E[W_1] \\ E[W_2] \\ \vdots \\ E[W_M] \end{bmatrix} = \begin{bmatrix} \dfrac{\mathcal{W}(\mathcal{W}+1)}{2M} \\ \vdots \\ \dfrac{\mathcal{W}(\mathcal{W}+1)}{2M} \end{bmatrix} \tag{2.22}$$

2.4 Variance-Covariance matrix of scores

The matrix $\mathbf{K}_{WW}^{H}$ is defined in (1.43) then we will find the variance $\mathrm{Var}W$ and the covariance $\mathrm{Cov}(W_i, W_j) = \vartheta$.

2.4.1 Var$\boldsymbol{W}$, variance of $\boldsymbol{W_i}$

Following 2.19, we have:

$$\mathrm{E}[W_i^2] = \sum_{k=0}^{\mathcal{W}} k^2 \; P_{W_i}(k) = 0^2 \frac{M-\mathcal{W}}{M} + \sum_{k=1}^{\mathcal{W}} k^2 \frac{1}{M} \tag{2.23}$$

$$= \frac{1}{M} \sum_{i=1}^{\mathcal{W}} k^2$$

$$= \frac{1}{M} \frac{\mathcal{W}(\mathcal{W}+1)(2\mathcal{W}+1)}{6} \tag{2.24}$$

Therefore,

$$\mathrm{Var}[W_i] = \mathrm{E}[W_i^2] - \mathrm{E}[W_i]^2 \tag{2.25}$$

$$= \frac{1}{M} \frac{\mathcal{W}(\mathcal{W}+1)(2\mathcal{W}+1)}{6} - \left(\frac{1}{M} \frac{\mathcal{W}(\mathcal{W}+1)}{2} \right)^2 \tag{2.26}$$

$$= \mu_W \; \frac{(2\mathcal{W}+1)}{3} - \mu_W^2 \tag{2.27}$$

2.4.2 $\boldsymbol{\vartheta}$, covariance of $\boldsymbol{W_i, W_j}$

From equations(2.16) and (2.19), the score random variables W_i of Ranked-NoM coding are not independent because:

$$\prod_{i=1}^{M} P_{W_i}(w) = \left(\frac{M-N}{M} \right)^{M-N} \left(\frac{1}{M} \right)^{N} \neq \frac{(M-N)!}{M!} = P_{W_1 W_2 \ldots W_M} \tag{2.28}$$

To find the covariance (1.42), we must find the bivariate joint probability distribution $f(W_i = w_i, W_j = w_j)$, for $i, j = 1, 2, \ldots, M$.

We will use the following facts:

1. Each random variable W_i can take on $\mathcal{W}$ same values: $0, 1, 2, \ldots, \mathcal{W}$.
2. The bivariate joint probability distribution $f(W_i, W_j)$ is the same for any $i, j = 1, \ldots, M$.
3. $f(W_i = k, W_j = k) = 0$ for $k \in \{1, 2, \ldots \mathcal{W}\}$
4. The support Ω_H, which is the sample space, consists of $|\Omega_H| = \dfrac{M!}{(M-\mathcal{W})!}$ possible pair outcomes that are equally likely, that is, the probability for each random outcome is $\dfrac{1}{|\Omega_H|}$.

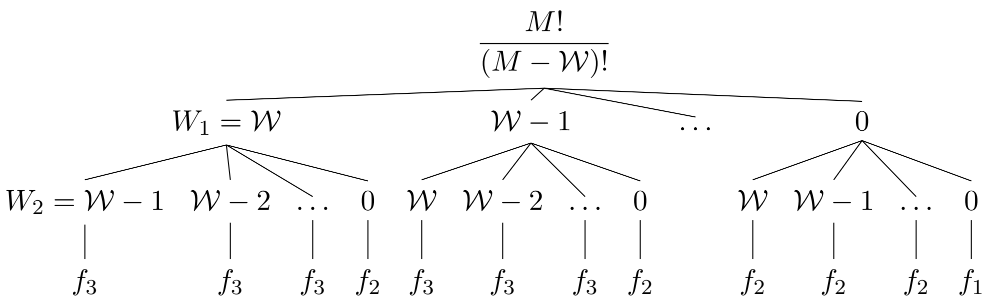
From fact (2), we only need to establish $f(W_1, W_2)$. From fact (3), we only need to establish the outcome frequency for each pair in:

$$(\mathcal{W}, \mathcal{W}-1), (\mathcal{W}, \mathcal{W}-2), \ldots, (\mathcal{W}-1, \mathcal{W}), (\mathcal{W}-1, \mathcal{W}-2), \ldots, (0, 0).$$

We first note that there are only three possibilities for the outcome frequencies, that is, for the pairs: $(0,0), (0,y), (x,y)$ where $x, y \in \{\mathcal{W}, \mathcal{W}-1, \mathcal{W}-2, \ldots, 1\}$ can take any non-zero integer value. We denote :

$$(0,0)_f = f_1 \tag{2.29}$$
$$(0,y)_f = (y,0)_f = f_2 \tag{2.30}$$
$$(x,y)_f = (y,x)_f = f_3 \tag{2.31}$$

We can then enumerate the $|\Omega_H|$ possible outcomes of $\mathbf{W}$ within a tree, with the first level corresponding to values taken by W_1, and the second level corresponding to values taken by W_2 once W_1 is fixed, with leaves indicating the number of times each pair repeats:



Tree for Ranked-NoM code (2.32)

For each branch, once values are given to W_1 and W_2, there are $(M-2)!$ permutations to consider, among which permutations corresponding to switching two null values should be considered the same. For a given branch, this number of permutations switching null values depends on the number of null already attributed to W_1 or W_2, hence:

$$f_1 = \frac{(M-2)!}{(M-\mathcal{W}-2)!} \tag{2.33}$$
$$f_2 = \frac{(M-2)!}{(M-\mathcal{W}-1)!} \tag{2.34}$$
$$f_3 = \frac{(M-2)!}{(M-\mathcal{W}-0)!} \tag{2.35}$$

Considering (from fact (4)) that the total number of possible pairs outcomes is $|\Omega_H| = \dfrac{M!}{(M-\mathcal{W})!}$, the corresponding probabilities are given by:

$$P_1 = \frac{f_1}{\frac{M!}{(M-\mathcal{W})!}} = \frac{\frac{(M-2)!}{(M-\mathcal{W}-2)!}}{\frac{M!}{(M-\mathcal{W})!}} = \frac{(M-\mathcal{W}-1)(M-\mathcal{W})}{(M-1)M} \tag{2.36}$$
$$P_2 = \frac{f_2}{\frac{M!}{(M-\mathcal{W})!}} = \frac{\frac{(M-2)!}{(M-\mathcal{W}-1)!}}{\frac{M!}{(M-\mathcal{W})!}} = \frac{M-\mathcal{W}}{(M-1)M} \tag{2.37}$$
$$P_3 = \frac{f_3}{\frac{M!}{(M-\mathcal{W})!}} = \frac{\frac{(M-2)!}{(M-W)!}}{\frac{M!}{(M-\mathcal{W})!}} = \frac{1}{(M-1)M} \tag{2.38}$$

We summarize the distribution f in the following table:

$$
\begin{array}{c|c|c|c|c|c|c}
f(W_i, W_j) & 0 & 1 & 2 & \dots & \mathcal{W} & f_{W_j}(w_j) \\
\hline
0 & P_1 & P_2 & P_2 & \dots & P_2 & P_1 + \mathcal{W}P_2 \\
\hline
1 & P_2 & 0 & P_3 & \dots & P_3 & P_2 + (\mathcal{W}-1)P_3 \\
\hline
2 & P_2 & P_3 & 0 & \dots & P_3 & P_2 + (\mathcal{W}-1)P_3 \\
\hline
\vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\
\hline
\mathcal{W} & P_2 & P_3 & P_3 & \dots & 0 & P_2 + (\mathcal{W}-1)P_3 \\
\hline
f_{W_i}(w_i) & P_1 + \mathcal{W}P_2 & & & \dots & P_2 + (\mathcal{W}-1)P_3 & 1
\end{array}
\tag{2.39}
$$

Calculating the covariance defined in the equation (1.42) by rows c_i, from the bivariate joint probability table, we have that $\mathrm{Cov}(W_i, W_j) = c_0 + c_1 + \cdots + c_{\mathcal{W}}$ where

$$
\begin{aligned}
c_0 &= P_1(0-\mu_W)(0-\mu_W) + P_2(0-\mu_W)(1-\mu_W) + \cdots + P_2(0-\mu_W)(\mathcal{W}-\mu_W) \\
c_1 &= P_2(1-\mu_W)(0-\mu_W) + P_3(1-\mu_W)(2-\mu_W) + \cdots + P_3(1-\mu_W)(\mathcal{W}-\mu_W) \\
c_{\mathcal{W}} &= P_2(\mathcal{W}-\mu_W)(0-\mu_W) + P_3(\mathcal{W}-\mu_W)(1-\mu_W) + \cdots + P_3(\mathcal{W}-\mu_W)(\mathcal{W}-1-\mu_W)
\end{aligned}
$$

Resolving the sum and simplifying, we determine that the covariance of the scores random variable for Ranked-NoM coding is given by:

$$
\mathrm{Cov}(W_i, W_j) = \frac{\mu_W}{M-1}\left(\mu_W - \frac{2\mathcal{W}+1}{3}\right) \tag{2.40}
$$

2.5 Modulation vector function

For (1.2), we denote $\Psi_H : \Omega \to \Xi_H$, the function for generating modulations vector for Ranked-NoM coding. The function we chose in the present article is parameterized by the number $\mathcal{N} \in \{1, \dots, M\}$ and defined as:

$$
\Psi_H(\hat{\mathbf{w}}^k; \mathcal{N}) \equiv \Phi_H(\hat{\mathbf{w}}^k; \mathcal{N}) \tag{2.41}
$$

Below, we consider $\mathbf{v}_H^1 = \Psi_H(\hat{\mathbf{w}}^1; \mathcal{N}) = (\mathcal{N}\ (\mathcal{N}-1) \dots 1\ 0 \dots 0)^\mathsf{T}$.

For Ranked-NoM coding, the modulation vector gated up to the first $I < \mathcal{N}$ components is then defined by

$$
\mathbf{v}_{H,I}^1 = (\mathcal{N}\ (\mathcal{N}-1) \dots (\mathcal{N}-(I-1))\ 0 \dots 0)^\mathsf{T} \tag{2.42}
$$

Then, for $I = \mathcal{N}$, that is, for the final potential, the modulation vector is given by

$$
\mathbf{v}_{H,\mathcal{N}}^1 = (\mathcal{N}\ (\mathcal{N}-1)\ \dots\ 1\ 0\ \dots\ 0)^\mathsf{T} \tag{2.43}
$$

For $I > \mathcal{N}$, we also take $\mathbf{v}_{H,\mathcal{N}}^1$.

2.6 $S_{H,\mathcal{N}}$. Integration-Final Potential $I = N$

2.6.1 max($S_{H,\mathcal{N}}$)

The maximum value of integration $S_{H,\mathcal{N}}$ is defined in equation (1.31). Thus for Ranked-NoM coding we have that $\mathbf{w}^1 = (\mathcal{W}, \mathcal{W}-1, \dots, 1, 0, \dots, 0)$ and modulation vector is given in the equation (2.43), therefore $\max(S_{H,\mathcal{N}})$ is,

$$
\begin{aligned}
&=< \mathbf{w}^1, \mathbf{v}^1_{H,\mathcal{N}} > \\
&= \mathcal{W}\mathcal{N} + (\mathcal{W}-1)(\mathcal{N}-1) + \ldots + (\mathcal{W}-(\mathcal{N}-1))(\mathcal{N}-(\mathcal{N}-1)) \\
&= \mathcal{N}^2\mathcal{W} - (\mathcal{W}+\mathcal{N})\left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - \mathcal{N}\right) + \left(\frac{\mathcal{N}(\mathcal{N}+1)(2\mathcal{N}+1)}{6} - \mathcal{N}^2\right)
\end{aligned} \tag{2.44}
$$

simplifying, the expression of the max value of integration is given by,

$$
\max(S_{H,\mathcal{N}}) = \mathcal{W}\mathcal{N}\left(\frac{\mathcal{N}+1}{2}\right) + \frac{\mathcal{N}(1-\mathcal{N}^2)}{6} \tag{2.45}
$$

2.6.2 $\mathbf{E}[\boldsymbol{S_{H,\mathcal{N}}}]$, expectation of final potential

$$
\mathrm{E}[S_{H,\mathcal{N}}] = \mathrm{E}[\mathbf{v}^{\mathsf{T}}_{H,\mathcal{N}}\,\mathbf{W}] = \mathbf{v}^{\mathsf{T}}_{H,\mathcal{N}}\,\mathrm{E}[\mathbf{W}] \tag{2.46}
$$

$$
= \mathbf{v}^{\mathsf{T}}_{H,\mathcal{N}} \cdot \frac{1}{M}\left(\frac{\mathcal{W}(\mathcal{W}+1)}{2}, ..., \frac{\mathcal{W}(\mathcal{W}+1)}{2}\right)^{\mathsf{T}} \tag{2.47}
$$

$$
= (\mathcal{N}\ (\mathcal{N}-1)\ \ldots\ 1\ 0\ \ldots\ 0) \cdot \frac{1}{M}\left(\frac{\mathcal{W}(\mathcal{W}+1)}{2}, ..., \frac{\mathcal{W}(\mathcal{W}+1)}{2}\right)^{\mathsf{T}} \tag{2.48}
$$

$$
\begin{aligned}
&= \frac{1}{M}\left(\frac{\mathcal{N}\mathcal{W}(\mathcal{W}+1)}{2} + \frac{\mathcal{W}(\mathcal{N}-1)(\mathcal{W}+1)}{2} + ... + \frac{\mathcal{W}(\mathcal{W}+1)}{2}\right) \\
&= \frac{1}{M}\frac{\mathcal{W}(\mathcal{W}+1)}{2}(\mathcal{N} + (\mathcal{N}-1) + ... + 1)
\end{aligned}
$$

$$
= \frac{1}{M}\frac{\mathcal{W}(\mathcal{W}+1)}{2}\frac{\mathcal{N}(\mathcal{N}+1)}{2} \tag{2.49}
$$

$$
= \frac{1}{4M}\,\mathcal{W}(\mathcal{W}+1)\,\mathcal{N}(\mathcal{N}+1) \tag{2.50}
$$

Then the expectation of Integration $\mathrm{E}[S_{H,\mathcal{N}}]$ is given by

$$
\mathrm{E}[S_{H,\mathcal{N}}] = \frac{1}{4M}\,\mathcal{W}(\mathcal{W}+1)\,\mathcal{N}(\mathcal{N}+1) \tag{2.51}
$$

2.6.3 $\mathbf{Var}[\boldsymbol{S_{H,\mathcal{N}}}]$, variance of final potential

From (1.38), the variance of integration is given by $\mathrm{Var}[S_{H,\mathcal{N}}] = \mathbf{v}^{\mathsf{T}}_{H,\mathcal{N}} \cdot \mathbf{K}^{H}_{WW} \cdot \mathbf{v}_{H,\mathcal{N}}$.

Using (1.43) for the expression of Variance-covariance matrix $\mathbf{K}^H_{WW}$ and the modulation vector $\mathbf{v}^1_{H,\mathcal{N}}$ for final potential given by (2.43), we have :

$$\mathrm{Var}[S_{H,\mathcal{N}}] = \mathbf{v}^{\mathsf{T}}_{H,\mathcal{N}} \cdot \mathbf{K}^H_{WW} \cdot \mathbf{v}_{H,\mathcal{N}} \tag{2.52}$$

$$= (\mathcal{N}\ (\mathcal{N}-1)\ \ldots\ 1\ 0\ \ldots\ 0) \begin{bmatrix} \mathrm{Var}W & \vartheta & \ldots & \vartheta \\ \vartheta & \mathrm{Var}W & \ldots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \ldots & \mathrm{Var}W \end{bmatrix} \begin{bmatrix} \mathcal{N} \\ \mathcal{N}-1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{2.53}$$

$$= \mathcal{N}^2\mathrm{Var}W + \mathcal{N}\ \vartheta \left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - \mathcal{N}\right) + (\mathcal{N}-1)^2\mathrm{Var}W + \tag{2.54}$$

$$(\mathcal{N}-1)\ \vartheta \left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - (\mathcal{N}-1)\right) + \cdots + \vartheta \left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - 1)\right) + \mathrm{Var}W$$

$$= \mathrm{Var}W\ (\mathcal{N}^2 + (\mathcal{N}-1)^2 + \cdots + 1) + \vartheta\ \left[\mathcal{N}\left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - N\right)\right] + \tag{2.55}$$

$$\vartheta\ \left[(\mathcal{N}-1)\left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - (\mathcal{N}-1)\right) + \cdots + \left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - 1\right)\right]$$

$$= \mathrm{Var}W\ \left(\frac{\mathcal{N}(\mathcal{N}+1)(2\mathcal{N}+1)}{6}\right) + \vartheta \left[\sum_{i=1}^{\mathcal{N}} i\left(\frac{\mathcal{N}(\mathcal{N}+1)}{2} - i\right)\right] \tag{2.56}$$

$$= \mathrm{Var}W\ \frac{\mathcal{N}(\mathcal{N}+1)}{2}\frac{2\mathcal{N}+1}{3} + \vartheta\ \left[\left(\frac{\mathcal{N}(\mathcal{N}+1)}{2}\right)^2 - \frac{\mathcal{N}(\mathcal{N}+1)}{2}\frac{2\mathcal{N}+1}{3}\right] \tag{2.57}$$

Denoting $p = \dfrac{\mathcal{N}(\mathcal{N}+1)}{2}$ and $q = \dfrac{2\mathcal{N}+1}{3}$, it reads:

$$\mathrm{Var}[S_{H,\mathcal{N}}] = (p\ q)\mathrm{Var}W\ + (p^2 - p\ q)\ \mathrm{Cov}(W_i, W_j) \tag{2.58}$$

where $\mathrm{Var}W$ is given by (2.27) and $\mathrm{Cov}(W_i, W_j)$ by (2.40).

In case $\mathcal{N} = \mathcal{W}$, we have:

$$\mathrm{Var}[S_{H,\mathcal{N}}] = \frac{M^2\ \mu_W^2}{M-1}\,(q - \mu_W)^2 \tag{2.59}$$

where μ_W is given by (2.21).

2.7 $S_{H,I}$. Integration-intermediate states, $I < \mathcal{N}$

The modulation vector gated up to the first I components is defined by $\mathbf{v}^1_{H,I} = (\mathcal{N}\ (\mathcal{N}-1)\ldots(\mathcal{N} - (I-1))\ 0\ldots0)^{\mathsf{T}}$ in equation (2.42).

2.7.1 $\mathbf{E}[S_{H,I}]$, expectation at intermediate states

$$
\begin{aligned}
\mathrm{E}[S_{H,I}] =& \mathrm{E}[\mathbf{v}_{H,I}^{\mathsf{T}}\,\mathbf{W}] = \mathbf{v}_{H,I}^{\mathsf{T}}\,\mathrm{E}[\mathbf{W}] && (2.60)\\
=&\mathbf{v}_{H,I}^{\mathsf{T}} \cdot \frac{1}{M}\left(\frac{\mathcal{W}(\mathcal{W}+1)}{2}\dots\frac{\mathcal{W}(\mathcal{W}+1)}{2}\right)^{\mathsf{T}} && (2.61)\\
=&(\mathcal{N}\ (\mathcal{N}-1)\dots(N-(I-1))\ 0\dots0)\cdot && (2.62)\\
&\frac{1}{M}\left(\frac{\mathcal{W}(\mathcal{W}+1)}{2}\dots\frac{\mathcal{W}(\mathcal{W}+1)}{2}\right)^{\mathsf{T}}\\
=&\frac{\mathcal{W}(\mathcal{W}+1)}{2M}(\mathcal{N}+\mathcal{N}-1+\dots+\mathcal{N}-(I-1)) && (2.63)\\
=&\frac{\mathcal{W}(\mathcal{W}+1)}{2M}\left(I\,\mathcal{N}-\frac{I(I-1)}{2}\right) && (2.64)
\end{aligned}
$$

The expectation of Integration at intermediate states $S_{H,I}$ is then given by:

$$
\mathrm{E}[S_{H,I}] = \frac{I\mathcal{W}(\mathcal{W}+1)}{4M}(2\,\mathcal{N}-I+1) \tag{2.65}
$$

2.7.2 $\mathbf{Var}[S_{H,I}]$, variance at intermediate states

From (1.38), we have $\mathrm{Var}[S_{H,I}] = \mathbf{v}_{H,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{H} \cdot \mathbf{v}_{H,I}$. The Variance-covariance matrix $\mathbf{K}_{WW}^{H}$ is given by (1.43) and modulation vector $\mathbf{v}_{H,I}^{1}$ by (2.42). Therefore we have:

$$
\begin{aligned}
\mathrm{Var}[S_{H,I}] &= \mathbf{v}_{H,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{H} \cdot \mathbf{v}_{H,I} \\
&= (\mathcal{N}\ (\mathcal{N}-1)\ldots(\mathcal{N}-(I-1))\ 0\ldots 0) \cdot \mathbf{K}_{WW}^{H} \cdot \begin{bmatrix} \mathcal{N} \\ (\mathcal{N}-1) \\ \vdots \\ (\mathcal{N}-(I-1)) \\ 0 \\ \vdots \\ 0 \end{bmatrix} && (2.66) \\
&= \mathrm{Var}W[\mathcal{N}^2 + (\mathcal{N}-1)^2 + \ldots + ((\mathcal{N}-(I-1))^2)] + \vartheta\,\mathcal{N} && (2.67) \\
&[\mathcal{N} + \mathcal{N} - 1 + \ldots + (\mathcal{N}-(I-1)] + \vartheta\,(\mathcal{N}-1) \\
&[\mathcal{N} + \mathcal{N} - 2 + \ldots + (\mathcal{N}-(I-1)] + \ldots + \vartheta\,(\mathcal{N}-(I-1)) \\
&[\mathcal{N} + \mathcal{N} - 1 + \ldots + (\mathcal{N}-(I-2))] \\
&= \mathrm{Var}W\left[\sum_{i=0}^{I-1}(\mathcal{N}-i)^2\right] + \vartheta\left[\sum_{i=0}^{I-1}(\mathcal{N}-i)(\mathcal{N}(I-1))\right] - && (2.68) \\
&\vartheta\left[\sum_{i=0}^{I-1}(\mathcal{N}-i)\left(\frac{I(I-1)}{2} - i\right)\right] \\
&= \mathrm{Var}W\left[\mathcal{N}^2 I - 2\,\mathcal{N}\frac{I(I-1)}{2} + \frac{I(I-1)}{2}\frac{(2\,I-1)}{3}\right] + \vartheta && (2.69) \\
&\left[\mathcal{N}\frac{I(I-1)}{2}(2\,\mathcal{N} - I + 1)\right] - \vartheta \\
&\left[\frac{I(I-1)}{2}\left(\mathcal{N}(I-1) - \frac{I(I-1)}{2} + \frac{(2\,I-1)}{3}\right)\right]
\end{aligned}
$$

Defining $\hat{p} = \dfrac{I(I-1)}{2}$ and $\hat{q} = \dfrac{2\,I-1}{3}$ and simplifying, we obtain the formula for the variance at intermediate states:

$$
\begin{aligned}
\mathrm{Var}[S_{H,I}] = [\mathcal{N}(\mathcal{N}I - 2\hat{p}) + \hat{p}\,\hat{q}]\ \mathrm{VarW} + \\
\hat{p}\,[2\mathcal{N}(\mathcal{N} - I + 1) + \hat{p} - \hat{q}]\ \mathrm{Cov}(W_i, W_j)
\end{aligned} \qquad (2.70)
$$

2.8 Behavior of discriminability for final potential

We set $\mathcal{N} = \mathcal{W} = M/2$

Let us define $p = \dfrac{\mathcal{N}(\mathcal{N}+1)}{2}$ and $q = \dfrac{2\mathcal{N}+1}{3}$, and substituting $\mathcal{N} = \mathcal{W}$, in each expression we get,

$$\max(S_{H,\mathcal{N}}) = \mathcal{W}\mathcal{N}\left(\frac{\mathcal{N}+1}{2}\right) + \frac{\mathcal{N}(1-\mathcal{N}^2)}{6} \tag{2.71}$$
$$= \mathcal{N}\mathcal{N}\left(\frac{\mathcal{N}+1}{2}\right) + \frac{\mathcal{N}(1+\mathcal{N})}{2}\frac{(1-\mathcal{N})}{3} \tag{2.72}$$
$$= \mathcal{N}\,p + p\,\frac{1-\mathcal{N}}{3} \tag{2.73}$$
$$= p\,(\mathcal{N} + \frac{1-\mathcal{N}}{3}) \tag{2.74}$$
$$= p\left(\frac{2\mathcal{N}+1}{3}\right) \tag{2.75}$$
$$= p\,q \tag{2.76}$$

$$\mathrm{E}[S_{H,\mathcal{N}}] = \lambda_C\,\mu_{W_C} \tag{2.77}$$
$$= \frac{\mathcal{N}(\mathcal{N}+1)}{2}\cdot\frac{\mathcal{W}(\mathcal{W}+1)}{2M} \tag{2.78}$$
$$= \frac{\mathcal{N}(\mathcal{N}+1)}{2}\cdot\frac{\mathcal{N}(\mathcal{N}+1)}{2M} \tag{2.79}$$
$$= \frac{p^2}{M} \tag{2.80}$$

For the variance $\mathrm{Var}[S_{H,\mathcal{N}}]$ we have that

$$\mathrm{Var}W_H = \mu_{W_H}\left(\frac{2\mathcal{W}+1}{3} - \mu_{W_H}\right) = \mu(q-\mu)$$

$$\mathrm{Cov}_H(W_i,W_j) = \frac{\mu_{W_H}}{M-1}\left(\mu_{W_H} - \frac{2\mathcal{W}+1}{3}\right) = \frac{\mu}{M-1}(\mu-q)$$

therefore,

$$\mathrm{Var}[S_{H,\mathcal{N}}] = (p\,q)\mathrm{Var}W\ + (p^2 - p\,q)\,\mathrm{Cov}(W_i,W_j) \tag{2.81}$$
$$= p\,q\,\mu(q-\mu) + (p^2 - p\,q)\,\frac{\mu}{M-1}(\mu-q) \tag{2.82}$$
$$= \frac{p\,\mu(q-\mu)(M\,q-p)}{M-1} \tag{2.83}$$

then substituting in the formula of discriminability, max (2.76), expectation(2.80) and variance (2.83) of integration for Ranked-NoM code, we get,

$$D_H(M) = \frac{\max(S_{H,\mathcal{N}}) - \mathrm{E}[S_{C,\mathcal{N}}]}{\sqrt{\mathrm{Var}[S_{C,\mathcal{N}}]}} \tag{2.84}$$

$$= \frac{p\,q - \dfrac{p^2}{M}}{\sqrt{\dfrac{p\,\mu(q-\mu)(M\,q-p)}{M-1}}} \tag{2.85}$$

$$= \sqrt{\frac{(p\,q - p^2/M)^2}{\dfrac{p\,\mu(q-\mu)(M\,q-p)}{M-1}}} \tag{2.86}$$

$$= \sqrt{\frac{p^2/M^2(qM-p)^2}{\dfrac{p\,\mu(q-\mu)(M\,q-p)}{M-1}}} \tag{2.87}$$

$$= \sqrt{\frac{(qM-p)(M-1)\,p}{M^2\mu(q-\mu)}} \tag{2.88}$$

we can see that we finally get:

$$D_H(M) = \sqrt{M-1} \tag{2.89}$$

3 APPLICATION TO N-OF-M (NOM) CODING

Since this scheme is already known as N-of-M coding, we keep the name. In our formalism, N will however become $\mathcal{W}$.

3.1 Scores vector function

The support Ω_F is generated by using the function of Ranked-NoM coding Φ_H. For NoM, Φ_F is a composition of the indicator function $\mathbf{1}_A$ with the Ranked-NoM function Φ_H where $A = \{w_i^k \in \Omega_H : w_i^k = \max(0, \hat{w}_i^k - M + \mathcal{W}) \neq 0\}$.

The scores are obtained by the vector-value function $\Phi_F(\hat{\mathbf{w}}^k) = \mathbf{w}_F^l$ defined in (1.1). Then the vectorial components are given by:

$$\phi_{F,i}(\hat{\mathbf{w}}^k) = \phi_{F,i}(\hat{w}_1^k, \dots, \hat{w}_M^k) = \mathbf{1}_A \circ \phi_{H,i}(\hat{w}_1^k, \dots, \hat{w}_M^k), \tag{3.1}$$

such as

$$\begin{aligned}
\Phi_F(\hat{\mathbf{w}}^k) &= \Phi_F(\hat{w}_1^k, \dots, \hat{w}_M^k) && (3.2)\\
&= (\phi_{F,1}(\hat{w}_1^k, \dots, \hat{w}_M^k), \dots, \phi_{F,M}(\hat{w}_1^k, \dots, \hat{w}_M^k)) && (3.3)\\
&= (\mathbf{1}_A(\phi_{H,1}(\hat{w}_1^k, \dots, \hat{w}_M^k)), \dots, \mathbf{1}_A(\phi_{H,M}(\hat{w}_1^k, \dots, \hat{w}_M^k))) && (3.4)\\
&= (\mathbf{1}_A(\max(0, \hat{w}_i^k - M + \mathcal{W}), \dots, (\mathbf{1}_A(\max(0, \hat{w}_i^k - M + \mathcal{W})))) && (3.5)\\
&= (w_1^l, \dots, w_M^l) = \mathbf{w}_F^l && (3.6)
\end{aligned}$$

Note that for $k = 1$, we have that $\hat{\mathbf{w}}^1 = (M, M-1 \dots, 1)$, thus

$$\begin{aligned}
\Phi_F(\hat{\mathbf{w}}^1) &= (\mathbf{1}_A(\phi_{H,1}(\hat{w}_1^1, \dots, \hat{w}_M^1)), \dots, \mathbf{1}_A(\phi_{H,M}(\hat{w}_1^1, \dots, \hat{w}_M^1))) && (3.7)\\
&= (\mathbf{1}_A(\max(0, \mathcal{W})), \dots, \mathbf{1}_A(\max(0, \mathcal{W} - M + 1))) && (3.8)\\
&= (\mathbf{1}_A(\mathcal{W}), \mathbf{1}_A(\mathcal{W} - 1), \dots, \mathbf{1}_A(0)) && (3.9)\\
&= (1, 1, \dots, 0) = \mathbf{w}_F^1 && (3.10)
\end{aligned}$$

The vectors in the support of Ranked-NoM coding by the function Φ_F get the vectors in Ω_H converted into vectors of ones and zeros. Therefore, we have that Ω_F gets reduced vectors each time we generated vectors from Ω_H, because we are no longer interested in their order. Then we divide by the number of ways that you can arrange $\mathcal{W}$ numbers, which is $\mathcal{W}!$. Thus the cardinality of support of NoM, Ω_F is:

$$\frac{|\Omega_H|}{\mathcal{W}!} = \frac{M!}{\mathcal{W}!(M - \mathcal{W})!} = \binom{M}{\mathcal{W}} = |\Omega_F| \tag{3.11}$$

and thus $\Omega_F = \{\mathbf{w}^1, ..., \mathbf{w}^{|\Omega_F|}\}$.

Example 3.1. As an illustration, let $M = 4$ and $\mathcal{W} = 2$ ($|\Omega| = 24$, $|\Omega_H| = 12$ permutations and $|\Omega_F| = \binom{4}{2} = 6$ combinations.). We would have, for the best order:

$$\begin{aligned}
\Phi_F(\hat{\mathbf{w}}^1) &= (\mathbf{1}_A(\max(0, 2)), \mathbf{1}_A(\max(0, 1)), \mathbf{1}_A(\max(0, 0)), \mathbf{1}_A(\max(0, -1))) && (3.12)\\
&= (\mathbf{1}_A(2), \mathbf{1}_A(1), \mathbf{1}_A(0), \mathbf{1}_A(0))\\
&= (1, 1, 0, 0) = \mathbf{w}_H^1 && (3.13)
\end{aligned}$$

and for the worst order:

$$\begin{aligned}
\Phi_F(\hat{\mathbf{w}}^{24}) &= (\mathbf{1}_A(\max(0, -1)), \mathbf{1}_A(\max(0, 0)), \mathbf{1}_A(\max(0, 1)), \mathbf{1}_A(\max(0, 2))) && (3.14)\\
&= (\mathbf{1}_A(0), \mathbf{1}_A(0), \mathbf{1}_A(1), \mathbf{1}_A(2))\\
&= (0, 0, 1, 1) = \mathbf{w}_H^6 && (3.15)
\end{aligned}$$

Note that for $\hat{\mathbf{w}}^{18} = (2, 1, 3, 4)$ and $\hat{\mathbf{w}}^{23} = (1, 2, 4, 3)$ we would also obtain:

$$\Phi_F(\hat{\mathbf{w}}^{18}) = (\mathbf{1}_A(\max(0,0)), \mathbf{1}_A(\max(0,-1)), \mathbf{1}_A(\max(0,1)), \mathbf{1}_A(\max(0,2))) \tag{3.16}$$
$$= (\mathbf{1}_A(0), \mathbf{1}_A(0), \mathbf{1}_A(1), \mathbf{1}_A(2))$$
$$= (0, 0, 1, 1) = \mathbf{w}_H^6 \tag{3.17}$$
$$\Phi_F(\hat{\mathbf{w}}^{23}) = (\mathbf{1}_A(\max(0,-1)), \mathbf{1}_A(\max(0,0)), \mathbf{1}_A(\max(0,2)), \mathbf{1}_A(\max(0,1))) \tag{3.18}$$
$$= (\mathbf{1}_A(0), \mathbf{1}_A(0), \mathbf{1}_A(2), \mathbf{1}_A(1))$$
$$= (0, 0, 1, 1) = \mathbf{w}_H^6 \tag{3.19}$$

therefore, $\Phi_F(\hat{\mathbf{w}}^{24}) = \Phi_F(\hat{\mathbf{w}}^{18}) = \Phi_F(\hat{\mathbf{w}}^{23}) = \mathbf{w}_H^6$.

3.2 Probability distribution of Scores

Let $\mathbf{W} = (W_1 \; W_2 \; \ldots \; W_M)^\mathsf{T}$ be a discrete random vector with support the ordered set $\Omega_F = \{\mathbf{w}^1, \ldots, \mathbf{w}^{\binom{M}{\mathcal{W}}}\}$, which is generated by the vector-function Φ_F defined in (3.1), thus the support is given by:

$$\Omega_F = \Omega_{W_1 W_2 \ldots W_M} = \left\{ \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \ldots \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 1 \end{bmatrix} \right\} \tag{3.20}$$

Due to (3.11), let its *joint probability mass function* be

$$P_{W_1 W_2 \ldots W_M} = \begin{cases} \dfrac{1}{\binom{M}{\mathcal{W}}} = \dfrac{(M-\mathcal{W})!W!}{\mathcal{W}!} & \mathbf{w}^k \in \Omega_{W_1 W_2 \ldots W_M} \\ 0 & \text{Otherwise} \end{cases} \tag{3.21}$$

From (1.17), the *marginal probability mass function* of $W_1, W_2, \ldots, W_M$ with sample space $\mathcal{D}_{W_i} = \{1, 0\}$ for $i = 1, ..., M$ is given by:

$$P_{W_i}(w) = \begin{cases} \dfrac{\binom{M-1}{\mathcal{W}}}{\binom{M}{\mathcal{W}}} = \dfrac{M-\mathcal{W}}{M} & w = 0 \\ \dfrac{\binom{M-1}{\mathcal{W}-1}}{\binom{M}{\mathcal{W}}} = \dfrac{\binom{M-1}{\mathcal{W}-1}}{\frac{M}{\mathcal{W}}\binom{M-1}{\mathcal{W}-1}} = \dfrac{\mathcal{W}}{M} & w = 1 \\ 0 & \text{Otherwise} \end{cases} \tag{3.22}$$

3.3 Expectation vector of scores

3.3.1 $\boldsymbol{\mu_W}$, expected value of $\boldsymbol{W_i}$

$$\mathrm{E}[W] = \sum_{i=1}^{2} w_i \, P(w_i) = 1 \, \frac{\mathcal{W}}{M} = \frac{\mathcal{W}}{M} = \mu_W \tag{3.23}$$

3.3.2 $\mathrm{E}[\mathbf{W}]$, expectation of $\mathbf{W}$,

Therefore, the expected value of random vector $\mathbf{W}$ is:

$$\mathrm{E}[\mathbf{W}] = \begin{bmatrix} E[W_1] \\ E[W_2] \\ \vdots \\ E[W_M] \end{bmatrix} = \begin{bmatrix} \frac{\mathcal{W}}{M} \\ \vdots \\ \frac{\mathcal{W}}{M} \end{bmatrix} \tag{3.24}$$

3.4 Variance-Covariance matrix of scores

The matrix $\mathbf{K}^F_{WW}$ is defined in (1.43) then we will find the variance Var$W$ and the covariance $\mathrm{Cov}(W_i, W_j) = \vartheta$.

3.4.1 Var$\boldsymbol{W}$, variance of $\boldsymbol{W_i}$

$$\mathrm{Var}(W) = \mathrm{E}[W^2] - \mathrm{E}[W]^2 \tag{3.25}$$

$$= \sum w_i^2 P(w_i) - \left(\frac{\mathcal{W}}{M}\right)^2 \tag{3.26}$$

$$= \frac{\mathcal{W}}{M} - \left(\frac{\mathcal{W}}{M}\right)^2 = \mu_W(1 - \mu_W) \tag{3.27}$$

Thus the variance of the random variable W is given by,

$$\mathrm{Var}(W) = \mu_W(1 - \mu_W) \tag{3.28}$$

3.4.2 $\boldsymbol{\vartheta}$, covariance of $\boldsymbol{W_i, W_j}$

From equations(3.21) and (3.22), the score random variable W_i of NoM coding are not independent because:

$$\prod_{i=1}^{M} P_{W_i}(w) = \left(\frac{M - \mathcal{W}}{M}\right)^{M-\mathcal{W}} \left(\frac{\mathcal{W}}{M}\right)^{\mathcal{W}} \neq \frac{(M - \mathcal{W})!\mathcal{W}!}{M!} = P_{W_1 W_2 \dots W_M} \tag{3.29}$$

To find the covariance (1.42), we must find the bivariate joint probability distribution $f(W_i = w_i, W_j = w_j)$, for $i, j = 1, 2, \dots, M$.

We have the following facts:

- Each random variable can take on 2 same binary values: $0, 1$.
- For any combination of w_i, w_j with $i, j = 1, \dots M$ the joint probability distribution f are the same.
- The support Ω_F which is the sample space consists of $|\Omega_F| = \binom{M}{\mathcal{W}}$ random possible pair outcomes that are equally likely, that is, the probability for each random outcome is $\frac{1}{|\Omega_F|}$.
- $f(W_i, W_j) = f(W_j, W_i)$

To establish the joint probability distribution f, we need a general method to find the frequency of the various pair outcomes. As we have the fact that the random pair outcomes are equally likely, we only need to count the number of times each pair repeats.

Let us introduce a notation to define the support Ω_F (equation (3.20)) as a matrix and from here we get the frequency hence the joint probability distribution.

Definition 3.1. Let $\widehat{\binom{M}{\mathcal{W}}}$ being a $M \times \binom{M}{\mathcal{W}}$ matrix. $\binom{M}{k}_1$ denote the number of 1's and $\binom{M}{k}_0$ the number of 0's at the matrix $\widehat{\binom{M}{\mathcal{W}}}$. $[\binom{\mathcal{M}}{k}]$ denote a $\mathcal{M} \times \binom{\mathcal{M}}{k}$ matrix that represents all the combinations $\mathcal{M}$ choose k. For $k = 0$, $[\binom{\mathcal{M}}{0}]$ is a $\mathcal{M} \times 1$ column of zeros, and $k = \mathcal{M}$, $[\binom{\mathcal{M}}{\mathcal{M}}]$ is a $\mathcal{M} \times 1$ column of ones.

To build the support Ω_F as a matrix, we first set the number one in the first row of the matrix and then set the zero. We know that the sum of ones and zeros is $|\Omega_F| = \binom{M}{\mathcal{W}}$. To know how many numbers one $(1's)$ we have, we use the combination formula for one less "1" and one less "element" of the vector. Using the notation above we will have $\binom{M-1}{\mathcal{W}-1}_1$ ones $(1's)$.

The number of zeros $(0's)$ is $\binom{M-1}{\mathcal{W}}_0$ since the size of the vector decreases but the ones $(1's)$ that are W, remain fixed. So we have $\binom{M-1}{\mathcal{W}-1}_1 + \binom{M-1}{\mathcal{W}}_0 = \binom{M}{\mathcal{W}}$. In the second row we set the ones and zeros again in the same way but for each row of ones and zeros, and so on.

In general, the matrix $\widehat{\binom{M}{\mathcal{W}}}$ can be represented as a tree, where each line j represents the jth row of the matrix

$$
\begin{array}{cccccccc}
 & & & \widehat{\binom{M}{\mathcal{W}}} & & & & \\
 & \binom{M-1}{\mathcal{W}-1}_1 & & & & \binom{M-1}{\mathcal{W}}_0 & & \\
\binom{M-2}{\mathcal{W}-2}_1 & & \binom{M-2}{\mathcal{W}-1}_0 & & \binom{M-2}{\mathcal{W}-1}_1 & & \binom{M-2}{\mathcal{W}}_0 & \\
\vdots \quad \vdots & & \binom{M-3}{\mathcal{W}-2}_1 \quad \vdots & & \vdots \quad \binom{M-3}{\mathcal{W}-1}_0 & & \binom{M-3}{\mathcal{W}-1}_1 \quad \vdots & \\
\binom{M-j}{\mathcal{W}-j}_1 \quad \binom{M-j}{\mathcal{W}-(j-1)}_0 & & \vdots & & \vdots & \vdots & \binom{M-j}{\mathcal{W}-1}_1 \quad \binom{M-j}{\mathcal{W}}_0 &
\end{array}
$$

$$\textit{Tree for NoM coding} \tag{3.30}$$

Note that the last row using the tree is when $\mathcal{W} = j$ but it is not the last row of the matrix. We define the matrices $[\binom{\mathcal{M}}{k}]$, to denote the entire matrix in the definition (3.1).

For instance, with $M = 4, \mathcal{W} = 2$ we have

$$\Omega_{W_1W_2W_3W_4} = \widehat{\begin{pmatrix}4\\2\end{pmatrix}} = \begin{bmatrix} & \binom{3}{1}_1 & & & \binom{3}{2}_0 & \\ \binom{2}{0}_1 & & \binom{2}{1}_0 & \binom{2}{1}_1 & & \binom{2}{2}_0 \\ [\binom{2}{0}] & & [\binom{2}{1}] & [\binom{2}{1}] & & [\binom{2}{2}] \end{bmatrix} \tag{3.31}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}_{4\times\binom{4}{2}} \tag{3.32}$$

the first row corresponds to the first random variable W_1, the second row to the W_2 random variable and so on.

To find the frequency we can consider two first random variables W_1, W_2 in Ω_F,

$$\Omega_{W_1 \dots W_M} = \widehat{\binom{M}{\mathcal{W}}} = \left[\begin{array}{cccccc} & \binom{M-1}{\mathcal{W}-1}_1 & & & \binom{M-1}{\mathcal{W}}_0 & \\ \binom{M-2}{\mathcal{W}-2}_1 & & \binom{M-2}{\mathcal{W}-1}_0 & \binom{M-2}{\mathcal{W}-1}_1 & & \binom{M-2}{\mathcal{W}}_0 \\ \vdots & & \vdots & \vdots & & \vdots \end{array}\right] \tag{3.33}$$

$$= \left[\begin{array}{cccccc} & \binom{M-1}{\mathcal{W}-1}_1 & & & \binom{M-1}{\mathcal{W}}_0 & \\ f_3 & & f_2 & f_2 & & f_1 \\ \vdots & & \vdots & \vdots & & \vdots \end{array}\right] \tag{3.34}$$

the frequency of the pair outcome $(1, 1)$ is $\binom{M-2}{\mathcal{W}-2}_1 = f_3$, of the pair $(1, 0) = (0, 1)$ is $\binom{M-2}{\mathcal{W}-1}_0 = \binom{M-2}{\mathcal{W}-1}_1 = f_2$ and $(0, 0)$ is $\binom{M-2}{\mathcal{W}}_0 = f_1$ therefore we have that,

$$f(W_1, W_1) = P_1 = \frac{f_1}{\binom{M}{\mathcal{W}}} = \frac{\binom{M-2}{\mathcal{W}}}{\binom{M}{\mathcal{W}}} \tag{3.35}$$

$$f(W_1, W_2) = P_2 = \frac{f_2}{\binom{M}{\mathcal{W}}} = \frac{\binom{M-2}{\mathcal{W}-1}}{\binom{M}{\mathcal{W}}} \tag{3.36}$$

$$f(W_2, W_2) = P_3 = \frac{f_3}{\binom{M}{\mathcal{W}}} = \frac{\binom{M-2}{\mathcal{W}-2}}{\binom{M}{\mathcal{W}}} \tag{3.37}$$

We summarize the bivariate joint probability distribution f in the following table,

$f(W_i, W_j)$	0	1	$f_{W_j}(W_j)$
0	P_1	P_2	$P_1 + P_2$
1	P_2	P_3	$P_2 + P_3$
$f_{W_i}(W_i)$	$P_1 + P_2$	$P_2 + P_3$	1

(3.38)

Now we can calculate the covariance(1.42). Doing the operations by rows, we have that $\text{Cov}(W_i, W_j) = c_0 + c_1$ where

$$c_0 = P_1\,(0 - \mu_W)(0 - \mu_W) + P_2\,(0 - \mu_W)(1 - \mu_W) \tag{3.39}$$

$$c_1 = P_2\,(1 - \mu_W)(0 - \mu_W) + P_3\,(1 - \mu_W)(1 - \mu_W) \tag{3.40}$$

Using the combination formula,

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}$$

the following properties of the combination

$$\binom{n}{1} = n, \quad \binom{n}{0} = \binom{n}{n} = 1, \quad \binom{n}{n-r} = \binom{n}{r}, \quad \binom{n}{r} + \binom{n}{r-1} = \binom{n+1}{r} \tag{3.41}$$

and simplifying, we get,

$$\mathrm{Cov}(W_i, W_j) = c_0 + c_1 \tag{3.42}$$
$$= P_1\, \mu_W^2 + P_2\, \mu_W(1-\mu_W) - P_2\, \mu_W(1-\mu_W) + P_3\,(1-\mu_W)^2 \tag{3.43}$$
$$= \mu_W^2 - 2\mu_W\, P_3 \frac{\mathcal{W}-1}{M-1} + P_3 \tag{3.44}$$
$$= \mu_W \frac{\mathcal{W}-1}{M-1} - \mu_W^2 \tag{3.45}$$

Thus the covariance $\mathrm{Cov}(W_i, W_j)$ of scores of NoM is given by:

$$\mathrm{Cov}(W_i, W_j) = \mu_W \frac{\mathcal{W}-1}{M-1} - \mu_W^2 \tag{3.46}$$

where μ_W is defined in the equation (3.23).

3.5 Modulation vector function

For (1.2), we denote $\Psi_F : \Omega \to \Xi_F$, the function for generating modulations vector for NoM coding. The function we chose in the present article is parameterized by the number $\mathcal{N} \in \{1, \ldots, M\}$ and defined as:

$$\Psi_F(\hat{\mathbf{w}}^k; \mathcal{N}) \equiv \Phi_F(\hat{\mathbf{w}}^k; \mathcal{N}) \tag{3.47}$$

Below, we consider $\mathbf{v}_F^1 = \Psi_F(\hat{\mathbf{w}}^1; \mathcal{N}) = (1, \ldots, 1_{\mathcal{N}}, 0, \ldots 0)^\mathsf{T}$, which contains $\mathcal{N}$ ones.

For NoM, the modulation vector gated up to the first I components is then defined by

$$\mathbf{v}_{F,I}^1 = (1, \ldots, 1, 0, \ldots 0)^\mathsf{T} \tag{3.48}$$

which contains $I < \mathcal{N}$ ones.

3.6 $S_{F,\mathcal{N}}$, integration-Final Potential $I = N$

3.6.1 max($S_{F,\mathcal{N}}$)

The maximum value of integration $S_{F,\mathcal{N}}$ is defined in equation (1.31). Thus, for NoM coding, we have that $\mathbf{w}^1 = (1, \ldots, 1_{\mathcal{W}}, 0, \ldots, 0)$ and modulation vector is given in the equation (3.48), therefore,

$$\max(S_{F,\mathcal{N}}) = 1 + 1 \ldots + 1 = \mathcal{N} \tag{3.49}$$

considering $\mathcal{W} > \mathcal{N}$. If $\mathcal{W} < \mathcal{N}$, $\max(S_{F,\mathcal{N}}) = \mathcal{W}$.

3.6.2 $\mathrm{E}[S_{F,\mathcal{N}}]$, expectation of final potential

$$\mathrm{E}[S_{F,\mathcal{N}}] = \mathrm{E}[\mathbf{v}_{F,I}^\mathsf{T}\, \mathbf{W}] = \mathbf{v}_{F,I}^\mathsf{T}\, \mathrm{E}[\mathbf{W}] \tag{3.50}$$
$$= \mathbf{v}_{F,I}^\mathsf{T} \cdot \left(\frac{\mathcal{W}}{M}, \ldots, \frac{\mathcal{W}}{M}\right)^\mathsf{T} \tag{3.51}$$
$$= (1\;1\;\ldots\;0) \cdot \left(\frac{\mathcal{W}}{M}, \ldots, \frac{\mathcal{W}}{M}\right)^\mathsf{T}$$
$$= \left(\frac{\mathcal{W}}{M} + \ldots + \frac{\mathcal{W}}{M}\right) = \frac{\mathcal{N}\,\mathcal{W}}{M} \tag{3.52}$$

Then the expectation of Integration $S_{F,\mathcal{N}}$ is given by

$$\mathrm{E}[S_{F,\mathcal{N}}] = \frac{\mathcal{N}\,\mathcal{W}}{M} \tag{3.53}$$

3.6.3 $\mathbf{Var}[S_{F,\mathcal{N}}]$, variance of final potential

From (1.38), we have $\mathrm{Var}[S_{F,\mathcal{N}}] = \mathbf{v}_{F,\mathcal{N}}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{F} \cdot \mathbf{v}_{F,\mathcal{N}}$. The Variance-covariance matrix, $\mathbf{K}_{WW}^{F}$, is defined in equation (1.43), and the modulation vector $\mathbf{v}_{F,\mathcal{N}}^{1}$ of final potential for NoM coding is given by (3.48). Therefore,

$$\mathrm{Var}[S_{F,\mathcal{N}}] = \mathbf{v}_{F,\mathcal{N}}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{F} \cdot \mathbf{v}_{F,\mathcal{N}} \tag{3.54}$$

$$= (1\ 1 \ldots 1\ 0\ \ldots 0) \begin{bmatrix} \mathrm{Var}W & \vartheta & \ldots & \vartheta \\ \vartheta & \mathrm{Var}W & \ldots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \ldots & \mathrm{Var}W \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.55}$$

$$= \mathrm{Var}W + (\mathcal{N}-1)\,\vartheta + \ldots + \mathrm{Var}W + (\mathcal{N}-1)\,\vartheta \tag{3.56}$$

$$= \mathcal{N}\ \mathrm{Var}W + \mathcal{N}\ (\mathcal{N}-1)\vartheta \tag{3.57}$$

Then the variance of integration is given by:

$$\mathrm{Var}[S_{F,\mathcal{N}}] = \mathcal{N}\ \mathrm{Var}W + \mathcal{N}\ (\mathcal{N}-1)\ \mathrm{Cov}(W_i, W_j) \tag{3.58}$$

where $\mathrm{Var}W$ is given by (3.28) and $\mathrm{Cov}(W_i, W_j)$ by (3.46).

We check that $\mathrm{Var}[S_{F,\mathcal{N}}]$ equation (3.58) is equivalent to the variance considering that NoM model follows the hypergeometric distribution:

$$\mathrm{Var}[S_{F,\mathcal{N}}] = \mathcal{N}\ \mathrm{Var}(W) + \mathcal{N}\ (\mathcal{N}-1)\ \mathrm{Cov}(W_i, W_j) \tag{3.59}$$

$$= \mathcal{N}(\mu_W(1-\mu_W)) + \mathcal{N}\ (\mathcal{N}-1)\left(\mu_W \frac{\mathcal{W}-1}{M-1} - \mu_W^2\right) \tag{3.60}$$

$$= N\ \mu_W[1-\mu_W + (\mathcal{N}-1)\frac{\mathcal{W}-1}{M-1} - (\mathcal{N}-1)\mu_W] \tag{3.61}$$

$$= \mathcal{N}\ \mu_W \left[1 - \frac{\mathcal{W}\,\mathcal{N}}{M} + \frac{(\mathcal{N}-1)(\mathcal{W}-1)}{M-1}\right] \tag{3.62}$$

$$= \frac{\mathcal{W}\,\mathcal{N}}{M}\left[\frac{(M-\mathcal{N})(M-\mathcal{W})}{M(M-1)}\right] \tag{3.63}$$

3.7 $S_{F,I}$. Integration-intermediate states, $I < \mathcal{N}$

3.7.1 $\mathbf{E}[S_{F,I}]$, expectation at intermediate states

$$\mathrm{E}[S_{F,I}] = \mathrm{E}[\mathbf{v}_{F,I}^{\mathsf{T}}\, \mathbf{W}] = \mathbf{v}_{F,I}^{\mathsf{T}}\, \mathrm{E}[\mathbf{W}] \tag{3.64}$$

$$= \mathbf{v}_{F,I}^{\mathsf{T}} \cdot \left(\frac{\mathcal{W}}{M}, ..., \frac{\mathcal{W}}{M}\right)^{\mathsf{T}}$$

$$= (1\ 1\ \ldots\ 0) \cdot \left(\frac{\mathcal{W}}{M}, ..., \frac{\mathcal{W}}{M}\right)^{\mathsf{T}}$$

$$= \left(\frac{\mathcal{W}}{M} + ... + \frac{\mathcal{W}}{M}\right) = \frac{I\,\mathcal{W}}{M} \tag{3.65}$$

Then the expectation of Integration $S_{F,I}$ is given by

$$\mathrm{E}[S_{F,I}] = \frac{I\,\mathcal{W}}{M} \tag{3.66}$$

3.7.2 $\mathbf{Var}[S_{F,I}]$, variance at intermediate states

From (1.38), the variance at intermediate states is given by $\mathrm{Var}[S_{F,I}] = \mathbf{v}_{F,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{F} \cdot \mathbf{v}_{F,I}$. From (1.43), we have the Variance-covariance matrix $\mathbf{K}_{WW}^{F}$ and the modulation vector $\mathbf{v}_{F,I}^{1}$ for NoM code is given by (3.48). Therefore,

$$\mathrm{Var}[S_{F,I}] = \mathbf{v}_{F,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{F} \cdot \mathbf{v}_{F,I}$$

$$= (1\ldots1\ 0\ \ldots0) \begin{bmatrix} \mathrm{Var}W & \vartheta & \ldots & \vartheta \\ \vartheta & \mathrm{Var}W & \ldots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \ldots & \mathrm{Var}W \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.67}$$

$$= \mathrm{Var}W + (I-1)\,\vartheta + \ldots + \mathrm{Var}W + (I-1)\,\vartheta \tag{3.68}$$

$$= I\ \mathrm{Var}W + I\ (I-1)\vartheta \tag{3.69}$$

Then the variance of integration is given by

$$\mathrm{Var}[S_{F,I}] = I\ \mathrm{Var}W + I\ (I-1)\ \mathrm{Cov}(W_i, W_j) \tag{3.70}$$

where $\mathrm{Var}W$ is given by (3.28) and $\mathrm{Cov}(W_i, W_j)$ by (3.46).

3.8 Behavior of discriminability for final potential

Substituting $\mathcal{N} = \mathcal{W} = M/2$, we have,

$$\max(S_{F,\mathcal{N}}) = \mathcal{N} = \frac{M}{2} \tag{3.71}$$

$$\mathrm{E}[S_{F,\mathcal{N}}] = \lambda_F\ \mu_{W_F} \tag{3.72}$$

$$= \mathcal{N}\frac{\mathcal{W}}{M} = \frac{M}{4} \tag{3.73}$$

thus for $\mathcal{N} = M/2$ and simplifying we get, For the variance $\mathrm{Var}[S_{F,\mathcal{N}}]$, we have that

$$\text{Var}W_F = \mu_{W_F}\left(1 - \mu_{W_F}\right) = \frac{\mathcal{W}}{M}\left(1 - \frac{\mathcal{W}}{M}\right) = 1/4$$

$$\text{Cov}_H(W_i, W_j) = \frac{\mu_{W_H}}{M-1}\left(\mu_{W_H} - \frac{2\mathcal{W}+1}{3}\right) = \frac{1}{4(1-M)}$$

therefore,

$$\mathrm{Var}[S_{F,\mathcal{N}}] = \mathcal{N}\text{Var}W \ + \mathcal{N}(\mathcal{N}-1)\,\text{Cov}(W_i, W_j) \tag{3.74}$$

$$= \frac{M}{2}\frac{1}{4} + \frac{M}{4}(M-2)\,\frac{1}{4(1-M)} \tag{3.75}$$

$$= \frac{M^2}{16(M-1)} \tag{3.76}$$

then substituting in the formula of discriminability, max (3.71), expectation(3.73) and variance (3.76) of integration for NoM code, we get a function depending on M,

$$D_F(M) = \frac{\max(S_{F,\mathcal{N}}) - \mathrm{E}[S_{F,\mathcal{N}}]}{\sqrt{\mathrm{Var}[S_{F,\mathcal{N}}]}} \tag{3.77}$$

$$= \frac{\frac{M}{2} - \frac{M}{4}}{\sqrt{\frac{M^2}{16(M-1)}}} \tag{3.78}$$

$$= \sqrt{M-1} \tag{3.79}$$

therefore,

$$D_F(M) = \sqrt{M-1} \tag{3.80}$$

4 APPLICATION TO RANK-ORDER CODING

4.1 Scores vector function

Rank-Order Coding weights have no other parameter than M. Under this coding, we want the best weights vector $\mathbf{w^1} = \hat{\mathbf{w}}^\mathbf{1} = \Phi(\mathcal{R}(1))$ made up of an arithmetic sequence from M down to zero, with rate -1. The weights are obtained by the vector-value function $\Phi_R(\hat{\mathbf{w}}^k) = \mathbf{w}_R^k$ define in (1.1). We then define $\phi_{R,i}$ as the identity function,

$$\phi_{R,i}(\hat{\mathbf{w}}^k) = \hat{\mathbf{w}}^k = \mathbf{w}_R^k \tag{4.1}$$

such that:

$$\begin{aligned}\Phi_R(\hat{\mathbf{w}}^k) &= \Phi_R(M - r_1^k, M - r_2^k, \ldots, M - r_M^k) && (4.2)\\ &= (M - r_1^k, M - r_2^k, \ldots, M - r_M^k) && (4.3)\\ &= (w_1^k, \ldots, w_M^k) = \mathbf{w}_R^k && (4.4)\end{aligned}$$

Note that, for $k = 1$, we have that $\hat{\mathbf{w}}^1 = (M, M-1 \ldots, 1)$, given that $\mathbf{r}^1 = (r_1^1, \ldots, r_M^1) = (0, 1, \ldots, M-1)$. Thus

$$\begin{aligned}\Phi_R(\hat{\mathbf{w}}^1) &= \Phi_R(M, M-1, \ldots, 1) && (4.5)\\ &= (M, M-1, \ldots, 1) = (w_1^1, \ldots, w_M^1) = \mathbf{w}_R^1 && (4.6)\end{aligned}$$

For $k = M!$, given that $\mathbf{r}^{M!} = (r_1^{M!}, r_2^{M!}, \ldots, r_M^{M!}) = ((M-1), (M-2), \ldots, 0)$ we have that $\hat{\mathbf{w}}^{M!} = (1, 2, \ldots, M)$, so we get:

$$\begin{aligned}\Phi_R(\hat{\mathbf{w}}^{M!}) &= \Phi_R(1, 2, \ldots, M) && (4.7)\\ &= (1, 2, \ldots, M) = (w_1^{M!}, \ldots, w_M^{M!}) = \mathbf{w}_R^{M!} && (4.8)\end{aligned}$$

Φ_R being a bijection, we have:

$$|\Omega| = |\Omega_R| = M! \tag{4.9}$$

Example 4.1. As an illustration, let $M = 4$ ($|\Omega_C| = 24$ permutations). We would have, for the best order:

$$\Phi_R(\hat{\mathbf{w}}^1) = \Phi_R(4, 3, 2, 1) = (4, 3, 2, 1) = \mathbf{w}_R^1 \tag{4.10}$$

and for the worst order

$$\Phi_R(\hat{\mathbf{w}}^{24}) = \Phi_R(1, 2, 3, 4) = (1, 2, 3, 4) = \mathbf{w}_R^{24} \tag{4.11}$$

4.2 Probability distribution function of scores

Let $\mathbf{W} = (W_1\ W_2\ \ldots\ W_M)^\mathsf{T}$ be the discrete random vector with support the ordered set $\Omega_R = \{\mathbf{w}^1, \ldots, \mathbf{w}^{M!}\}$, which is generated by the vector-value function $\hat{\Phi}_R$ defined in (4.1), therefore,

$$\Omega_R = \Omega_{W_1 W_2 \ldots W_M} = \left\{ \begin{bmatrix} M \\ M-1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} M-1 \\ M \\ \vdots \\ 1 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ \vdots \\ M-1 \\ M \end{bmatrix} \right\} \tag{4.12}$$

Due to (4.9), its *joint probability mass function* is

$$P_{W_1 W_2 \ldots W_M} = \begin{cases} \dfrac{1}{M!} & \mathbf{w}^k \in \Omega_{W_1 W_2 \ldots W_M} \\ 0 & \text{Otherwise} \end{cases} \tag{4.13}$$

and from (1.17), the *marginal probability mass function* of $W_1, W_2, \ldots, W_M$ with sample space $\mathcal{D}_{W_i} = \{M, M-1, ..., 1\}$ for $i = 1, ..., M$, is given by:

$$P_{W_i}(w) = \begin{cases} \dfrac{(M-1)!}{M!} = \dfrac{1}{M} & w = 1, \ldots, M \\ 0 & \text{Otherwise} \end{cases} \tag{4.14}$$

4.3 Expectation vector of scores

4.3.1 $\boldsymbol{\mu_W}$, expected value of $\boldsymbol{W_i}$

$$\mathrm{E}[W] = \sum_{i=1}^{M} w_i \, P(w_i) = \sum_{i=1}^{M} w_i \, \frac{1}{M} \tag{4.15}$$

$$= \frac{1}{M} \sum_{i=1}^{M} w_i = \frac{1}{M} \frac{M(M+1)}{2} = \frac{(M+1)}{2} \tag{4.16}$$

4.3.2 $\mathbf{E}[\mathbf{W}]$, expectation of $\mathbf{W}$

Therefore, the expected value of a random vector $\mathbf{W}$ is:

$$\mathrm{E}[\mathbf{W}] = \begin{bmatrix} E[W_1] \\ E[W_2] \\ \vdots \\ E[W_M] \end{bmatrix} = \begin{bmatrix} \dfrac{M+1}{2} \\ \vdots \\ \dfrac{M+1}{2} \end{bmatrix} \tag{4.17}$$

4.4 Variance-Covariance matrix of scores

The matrix $\mathbf{K}^R_{WW}$ is defined in (1.43) then we will find the variance Var$W$ and the covariance Cov$(W_i, W_j) = \vartheta$.

4.4.1 Var$\boldsymbol{W}$, variance of $\boldsymbol{W_i}$

$$\mathrm{Var}(W) = \mathrm{E}[W^2] - \mathrm{E}[W]^2 \tag{4.18}$$

$$= \sum w_i^2 P(w_i) - \frac{(M+1)^2}{4} \tag{4.19}$$

$$= \sum w_i^2 \frac{1}{M} - \frac{(M+1)^2}{4}$$

$$= \frac{1}{M} \left(\frac{M(M+1)(2M+1)}{6} \right) - \frac{(M+1)^2}{4} \tag{4.20}$$

$$= \frac{M^2 - 1}{12} \tag{4.21}$$

thus the variance of random variable W is given by,

$$\mathrm{Var}W = \frac{M^2 - 1}{12} = \frac{\mu_W\,(M-1)}{6} \tag{4.22}$$

4.4.2 $\boldsymbol{\vartheta}$, covariance of $\boldsymbol{W_i, W_j}$

From (4.13) and (4.14), the score random variable W_i of ROC coding are not independent because:

$$\prod_{i=1}^{M} P_{W_i}(w) = \left(\frac{1}{M}\right)^M \neq \frac{1}{M!} = P_{W_1 W_2 \ldots W_M} \tag{4.23}$$

hence $\mathrm{Cov}(W_i, W_j) \neq 0$.

From (1.42),we must find the bivariate joint probability distribution $f(W_i = w_i, W_j = w_j), i, j = 1, 2, \ldots, M$ to find the covariance. We will use the following facts:

- Each random variable can take on M same values: $1, 2, \ldots, M$
- For any combination of w_i, w_j with $i, j = 1, \ldots M$ the joint probability distribution f are the same.
- The support Ω_R which is the sample space consists of $|\Omega_R| = M!$ possible pair outcomes that are equally likely, that is, the probability for each outcome is $\dfrac{1}{|\Omega_R|}$.
- $f(W_i, W_j) = f(W_j, W_i)$
- $f(W_i = w_i, W_j = w_j) = 0$, for $i = j$

To obtain the support Ω_R (4.12) as a matrix where the rows would represent the random variables $W_1, W_2, \ldots, W_M$, we consider all permutations of the set $(M, M-1, \ldots, 1)$, and arrange them as columns in decreasing lexical order.

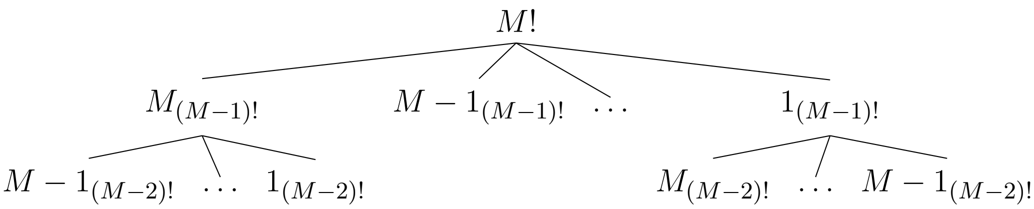
For example, for $M = 3$ the support represented as a matrix is given by:

$$\Omega_1 = \begin{bmatrix} 3 & 3 & 2 & 2 & 1 & 1 \\ 2 & 1 & 3 & 1 & 3 & 2 \\ 1 & 2 & 1 & 3 & 2 & 3 \end{bmatrix}_{3\times 3!} \tag{4.24}$$

and for $M = 4$ we have that,

$$\Omega_2 = \begin{bmatrix} 4 & 4 & 4 & 4 & 4 & 4 & \ldots \\ 3 & 3 & 2 & 2 & 1 & 1 & \ldots \\ 2 & 1 & 3 & 1 & 3 & 2 & \ldots \\ 1 & 2 & 1 & 3 & 2 & 3 & \ldots \end{bmatrix}_{4\times 4!} \tag{4.25}$$

Let us consider the two first variables W_1, W_2. If we fix the value for W_1, we are left with the permutations of the $M-1$ remaining values. Hence, each value appears $(M-1)!$ times. If we now fix also the value for W_2, we are left with the permutations of the $M-2$ remaining values. Hence, we have :



Tree for ROC coding (4.26)

Therefore, the frequency is the same for each possible pair outcome, and $f = (M-2)!$. Then the probability P is given by:

$$P = \frac{(M-2)!}{M!} = \frac{(M-2)!}{(M-2)!\,(M-1)\,M} = \frac{1}{(M-1)\,M} \tag{4.27}$$

We summarize the distribution f in the following table,

$f(W_i, W_j)$	1	2	...	M	$f_{W_j}(W_j)$
1	0	P	...	P	$(M-1)P$
2	P	0	...	P	$(M-1)P$
$\vdots$	$\vdots$	$\vdots$	...	$\vdots$	$\vdots$
M	P	P	...	0	$(M-1)P$
$f_{W_i}(w_i)$	$(M-1)P$	$(M-1)P$	...		1

(4.28)

To calculate the covariance defined in the equation (1.42), we do the operations by rows $c_1, \ldots, c_M$ from the bivariate joint distribution, and obtain:

$$\begin{aligned}
c_1 &= (1-\mu_W)(2-\mu_W)\,P + (1-\mu_W)(3-\mu_W)\,P + \ldots + (1-\mu_W)(M-\mu_W)\,P \\
c_2 &= (2-\mu_W)(1-\mu_W)\,P + (2-\mu_W)(3-\mu_W)\,P + \ldots + (2-\mu_W)(M-\mu_W)\,P \\
&\ldots \\
c_M &= (M-\mu_W)(1-\mu_W)\,P + \ldots + (M-\mu_W)((M-1)-\mu_W)\,P
\end{aligned}$$

Therefore $\text{Cov}(W_i, W_j) = c_1 + c_2 + \ldots + c_M$,

$$\begin{aligned}
\text{Cov}(W_i, W_j) =& (1-\mu_W)\,P[(2-\mu_W) + \ldots + (M-\mu_W)] + \\
& (2-\mu_W)\,P[(1-\mu_W) + (3-\mu_W) \ldots + (M-\mu_W) + \\
& (M-\mu_W)\,P[(1-\mu_W) + (2-\mu_W) \ldots + ((M-1)-\mu_W)] \qquad (4.29)\\
=& (1-\mu_W)\,P\left[\frac{M(M+1)}{2} - 1 - (M-1)\mu_W\right] + \\
& (2-\mu_W)\,P\left[\frac{M(M+1)}{2} - 2 - (M-1)\mu_W\right] + \\
& (M-\mu_W)\,P\left[\frac{M(M+1)}{2} - M - (M-1)\mu_W\right] \qquad (4.30)\\
=& \sum_{i=1}^{M} (i-\mu_W)\,P\,[M\,\mu_W - i - (M-1)\mu_W] \qquad (4.31)\\
=& \sum_{i=1}^{M} (i-\mu_W)\,P\,(\mu_W - i) = -P\,\sum_{i=1}^{M} (i-\mu_W)^2 \qquad (4.32)
\end{aligned}$$

Using the summation formulas and simplifying, we finally get that the covariance of the scores for ROC coding is given by:

$$\text{Cov}(W_i, W_j) = \frac{\mu_W}{M-1}\left(\mu_W - \frac{2M+1}{3}\right) \tag{4.33}$$

4.5 Modulations vector function

Under the geometrical-ROC scheme that we consider in this paper, we want the modulations vector for $\mathbf{r}^1$ to be composed as a decreasing geometric sequence, starting from 1, and with rate m.

For (1.2), we denote $\Psi_R : \Omega \to \Xi_R$, the function for generating modulations vector for ROC.

It is parametized by the numbers $M \in \{1, \ldots, M\}$ and $m \in [0, 1]$.

We then define $\psi_{R,i}$ as:

$$\psi_{R,i}(\hat{\mathbf{w}}^k; m) = \psi_{R,i}((\hat{w}_1^k, \ldots, \hat{w}_M^k); m) = m^{M-\hat{w}_i^k} \tag{4.34}$$

such that

$$\begin{aligned}\Psi_R(\hat{\mathbf{w}}^k) &= \Psi_R(\hat{w}_1^k, \ldots, \hat{w}_M^k) &(4.35)\\ &= (\psi_{R,1}(\hat{w}_1^k, \ldots, \hat{w}_M^k), \ldots, \psi_{R,M}(\hat{w}_1^k, \ldots, \hat{w}_M^k)) &(4.36)\\ &= (m^{M-\hat{w}_1^k}, \ldots, m^{M-\hat{w}_M^k}) = \mathbf{v}_R^l &(4.37)\end{aligned}$$

with $l \in \{1, \ldots |\Omega_R|\}$.

Ψ_R is a bijective function so $|\Omega_R| = |\Lambda| = M!$.

For $k = 1$, we have that $\mathbf{r}^1 = (r_1^1 \; r_2^1 \; \ldots \; r_M^1)^T = (1 \; 2 \; \ldots \; M)^T$ then

$$\begin{aligned}\Psi_R(\hat{\mathbf{w}}^1) &= (\psi_{R,1}(\hat{w}_1^1, \ldots, \hat{w}_M^1), \ldots, \psi_{R,M}(\hat{w}_1^1, \ldots, \hat{w}_M^1)) &(4.38)\\ &= (m^{M-\hat{w}_1^1}, \ldots, m^{M-\hat{w}_M^1}) &(4.39)\\ &= (m^{M-M}, \ldots, m^{M-1}) &(4.40)\\ &= (m^0, m^1 \ldots, m^{M-1}) = \mathbf{v}_R^1 &(4.41)\end{aligned}$$

Below, we consider the modulation vector being : $\mathbf{v}_{R,M}^1 = \Psi_R(\hat{\mathbf{w}}^1; m)$.

The modulation vector gated up to the first $I < M$ components is then defined by

$$\Psi_R(\hat{\mathbf{w}}^1) = (m^0, m^1 \ldots, m^{I-1}, 0, \ldots, 0) \tag{4.42}$$

4.6 $S_{R,M}$. Integration-Final Potential $I = M$

4.6.1 max($S_{R,M}$)

The maximum value of integration $S_{R,M}$ is defined in equation (1.31). Thus, for ROC coding, we have that $\mathbf{w}^1 = (M, M-1, \ldots, 1)$ and modulation vector is given by (4.41). Thus $\max(S_{R,M})$ is

$$\begin{aligned}&=< \mathbf{w}^1, \mathbf{v}_{R,M}^1 >\\ &= (M-0)m^0 + (M-1)m^1 + (M-2)m^2 + \ldots + (M-(M-1))m^{M-1}\\ &= M(m^0 + m^1 + \ldots + m^{M-1}) - (m^1 + 2m^2 + \ldots + (M-1)m^{M-1}) &(4.43)\\ &= M\sum_{i=0}^{M-1} m^i - \sum_{i=1}^{M-1} i\, m^i &(4.44)\\ &= M\left(\frac{1-m^M}{1-m}\right) - \frac{1-m^{M+1}-(Mm^M+1)(1-m)}{(1-m)^2} &(4.45)\end{aligned}$$

Simplifying, the max of integration for ROC is given by:

$$\max(S_{R,M}) = \frac{(1-m)(1+M)-(1-m^{M+1})}{(1-m)^2} \tag{4.46}$$

4.6.2 $\mathbf{E}[S_{R,M}]$, expectation of final potential

$$\begin{aligned}
\mathrm{E}[S_{R,M}] &= \mathrm{E}[\mathbf{v}_{R,M}^{\mathsf{T}}\ \mathbf{W}] = \mathbf{v}_{R,M}^{\mathsf{T}}\ \mathrm{E}[\mathbf{W}] && (4.47)\\
&= \mathbf{v}_{R,M}^{\mathsf{T}} \cdot \left(\frac{M+1}{2}, \ldots, \frac{M+1}{2}\right)^{\mathsf{T}} && (4.48)\\
&= (m^0\ m^1\ \ldots\ m^{M-1}) \cdot \left(\frac{M+1}{2}, \ldots, \frac{M+1}{2}\right)^{\mathsf{T}} && (4.49)\\
&= \left(\frac{m^0(M+1)}{2} + \frac{m^1(M+1)}{2} + \ldots + \frac{m^{M-1}(M+1)}{2}\right) && \\
&= \frac{(M+1)}{2}(m^0 + m^1 + \ldots + m^{M-1}) && (4.50)\\
&= \frac{(M+1)}{2}\frac{1-m^M}{1-m} && (4.51)
\end{aligned}$$

Then the expectation of Integration $S_{R,M}$ is given by

$$\mathrm{E}[S_{R,M}] = \frac{(M+1)}{2}\frac{1-m^M}{1-m} \tag{4.52}$$

4.6.3 $\mathbf{Var}[S_{R,M}]$, variance of final potential

From (1.38), we have $\mathrm{Var}[S_{R,M}] = \mathbf{v}_{R,M}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{R} \cdot \mathbf{v}_{R,M}$. The Variance-covariance matrix, $\mathbf{K}_{WW}^{R}$ is given by (1.43), and the modulation vector $\mathbf{v}_{R,M}^{1}$ of final potential for geometric ROC coding is given by (4.41). Therefore:

$$
\begin{aligned}
&\mathrm{Var}[S_{R,M}] \\
&= \mathbf{v}_{R,M}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{R} \cdot \mathbf{v}_{R,M} \\
&= (m^0\ m^1\ \ldots\ m^{M-1}) \begin{bmatrix} \mathrm{Var}W & \vartheta & \ldots & \vartheta \\ \vartheta & \mathrm{Var}W & \ldots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \ldots & \mathrm{Var}W \end{bmatrix} \begin{bmatrix} m^0 \\ m^1 \\ \vdots \\ m^{M-1} \end{bmatrix} && (4.53) \\
&= m^0(m^0\mathrm{Var}W + m^1\vartheta + \ldots + m^M\vartheta) + m^1(m^0\vartheta + m^1\mathrm{Var}W + \ldots + m^M\vartheta) + && (4.54) \\
&\ldots + m^{M-1}(m^0\vartheta + m^1\vartheta + \ldots + m^{M-1}\mathrm{Var}W) \\
&= \mathrm{Var}W \sum_{i=0}^{M-1} m^{2i} + && (4.55) \\
&\vartheta \left[m^0 \left(\sum_{i=0}^{M-1} m^i - m^0 \right) + m^1 \left(\sum_{i=0}^{M-1} m^i - m^1 \right) + m^{M-1} \left(\sum_{i=0}^{M-1} m^i - m^{M-1} \right) \right] \\
&= \left(\frac{1-m^{2M}}{1-m^2} \right) \mathrm{Var}W + \left[\sum_{i=0}^{M-1} m^i \left(\frac{1-m^M}{1-m} - m^i \right) \right] \vartheta && (4.56)
\end{aligned}
$$

Using the geometric summation formulas, we get:

$$
\mathrm{Var}[S_{R,M}] = \left(\frac{1-m^{2M}}{1-m^2} \right) \mathrm{Var}W + \left[\left(\frac{1-m^M}{1-m} \right)^2 - \frac{1-m^{2M}}{1-m^2} - \right] \mathrm{Cov}(W_i, W_j) \tag{4.57}
$$

where $\mathrm{Var}W$ is given by (4.22) and $\mathrm{Cov}(W_i, W_j)$ by (4.33).

4.7 $S_{R,I}$. Integration-intermediate states, $I < M$

4.7.1 $\mathrm{E}[S_{R,I}]$, expectation at intermediate states

$$
\begin{aligned}
\mathrm{E}[S_{R,I}] &= \mathrm{E}[\mathbf{v}_{R,I}^{\mathsf{T}}\ \mathbf{W}] = \mathbf{v}_{R,I}^{\mathsf{T}}\ \mathrm{E}[\mathbf{W}] && (4.58) \\
&= \mathbf{v}_{R,I}^{\mathsf{T}} \cdot \left(\frac{(M+1)}{2}, \ldots, \frac{(M+1)}{2} \right)^{\mathsf{T}} \\
&= (m^0\ m^1\ \ldots\ m^{I-1}\ 0 \ldots 0) \cdot \left(\frac{(M+1)}{2}, \ldots, \frac{(M+1)}{2} \right)^{\mathsf{T}} \\
&= \left(\frac{m^0(M+1)}{2} + \frac{m^1(M+1)}{2} + \ldots + \frac{m^{I-1}(M+1)}{2} \right) \\
&= \frac{(M+1)}{2}(m^0 + m^1 + \ldots + m^{I-1}) \\
&= \frac{(M+1)}{2} \frac{1-m^I}{1-m} && (4.59)
\end{aligned}
$$

Then the expectation of Integration $S_{R,I}$ is given by

$$
\mathrm{E}[S_{R,I}] = \frac{(M+1)}{2} \frac{1-m^I}{1-m} \tag{4.60}
$$

4.7.2 $\mathrm{Var}[S_{R,I}]$, variance at intermediate states

From (1.38), the variance at intermediate states is given by $\mathrm{Var}[S_{R,I}] = \mathbf{v}_{R,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{R} \cdot \mathbf{v}_{R,I}$. The Variance-covariance matrix $\mathbf{K}_{WW}^{R}$ is given by (1.43) and modulation vector $\mathbf{v}_{R,I}^{1}$ by (4.41). Therefore we have:

$$
\begin{aligned}
&\mathrm{Var}[S_{R,I}] \\
&= \mathbf{v}_{R,I}^{\mathsf{T}} \cdot \mathbf{K}_{WW}^{R} \cdot \mathbf{v}_{R,I} && (4.61)\\
&= (m^0\ m^1\ \ldots\ m^{I-1}\ 0 \ldots 0) \begin{bmatrix} \mathrm{Var}W & \vartheta & \ldots & \vartheta \\ \vartheta & \mathrm{Var}W & \ldots & \vartheta \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta & \vartheta & \ldots & \mathrm{Var}W \end{bmatrix} \begin{bmatrix} m^0 \\ m^1 \\ \vdots \\ m^{I-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix} && (4.62)\\
&= m^0(m^0\mathrm{Var}W + m^1\vartheta + \ldots + m^I\vartheta) + m^1(m^0\vartheta + m^1\mathrm{Var}W + \ldots + m^I\vartheta) + && (4.63)\\
&\ldots + m^{I-1}(m^0\vartheta + m^1\vartheta + \ldots + m^{I-1}\mathrm{Var}W) \\
&= \mathrm{Var}W \sum_{i=0}^{I-1} m^{2i} + && (4.64)\\
&\vartheta \left[m^0 \left(\sum_{i=0}^{I-1} m^i - m^0 \right) + m^1 \left(\sum_{i=0}^{I-1} m^i - m^1 \right) + m^{I-1} \left(\sum_{i=0}^{I-1} m^i - m^{I-1} \right) \right] \\
&= \left(\frac{1-m^{2I}}{1-m^2} \right) \mathrm{Var}W + \left[\sum_{i=0}^{I-1} m^i \left(\frac{1-m^I}{1-m} - m^i \right) \right] \vartheta && (4.65)
\end{aligned}
$$

Using the geometric summation formulas, we get:

$$
\mathrm{Var}[S_{R,I}] = \left(\frac{1-m^{2I}}{1-m^2} \right) \mathrm{Var}W + \left[\left(\frac{1-m^I}{1-m} \right)^2 - \frac{1-m^{2I}}{1-m^2} - \right] \mathrm{Cov}(W_i, W_j) \tag{4.66}
$$

where $\mathrm{Var}W$ is given by (4.22) and $\mathrm{Cov}(W_i, W_j)$ by (4.33).

4.8 Behavior of discriminability for final potential

Let us assign $p = \dfrac{1-m^{2M}}{1-m^2}, q = \dfrac{1-m^M}{1-m}$ and substituting in the formulas,

$$
\begin{aligned}
\max(S_{R,M}) &= \frac{(1-m)(1+M) - (1-m^{M+1})}{(1-m)^2} && (4.67)\\
&= \frac{1+M}{1-m} - \frac{(1-m^{M+1})}{(1-m)^2} && (4.68)\\
&= \frac{1+M}{1-m} - \frac{q+m^M}{1-m} && (4.69)\\
&= \frac{M-q\,m}{1-m} && (4.70)
\end{aligned}
$$

$$\begin{aligned}
\mathrm{E}[S_{F,M}] &= \lambda_F\, \mu_{W_F} &(4.71)\\
&= \frac{1-m^M}{1-m}\,\frac{M+1}{2} &(4.72)\\
&= q\,\frac{M+1}{2} &(4.73)
\end{aligned}$$

$$\begin{aligned}
\mathrm{Var}[S_{R,\mathcal{N}}] &= p\mathrm{Var}W\ + (q^2-p)\,\mathrm{Cov}(W_i,W_j) &(4.74)\\
&= p\,\mu_{W_R}\left(\frac{M-1}{6}\right) + (q^2-p)\frac{\mu_{W_R}}{M-1}\left(\mu_{W_R} - \frac{2M+1}{3}\right) &(4.75)\\
&= \frac{\mu}{6(M-1)}\left(p(M-1)^2 + 6(q^2-p)\left(\mu - \frac{2M+1}{3}\right)\right) &(4.76)\\
&= \frac{M+1}{12}(p\,M - q^2) &(4.77)
\end{aligned}$$

then substituting in the formula of discriminability, max (4.70), expectation(4.73) and variance (4.77) of integration for ROC code, we get the following function,

$$\begin{aligned}
D_F(M) &= \frac{\max(S_{F,M}) - \mathrm{E}[S_{F,M}]}{\sqrt{\mathrm{Var}[S_{F,M}]}} &(4.78)\\
&= \frac{\dfrac{M-q\,m}{1-m} - q\,\dfrac{M+1}{2}}{\sqrt{\dfrac{M+1}{12}(p\,M-q^2)}} &(4.79)\\
&= \sqrt{\frac{\dfrac{(2(M-q\,m)-q(M+1)(1-m))^2}{4(1-m)^2}}{\dfrac{M+1}{12}(p\,M-q^2)}} &(4.80)\\
&= \frac{\sqrt{3}}{1-m}\sqrt{\frac{(2(M-q\,m)-q(M+1)(1-m))^2}{(M+1)(p\,M-q^2)}} &(4.81)
\end{aligned}$$

dividing numerator and denominator of the radicand by M^2 to find the tendency of the function when $M \to \infty$, we get,

$$D_F(M) = \frac{\sqrt{3}}{1-m}\sqrt{\frac{\left(2\left(1-\dfrac{q\,m}{M}\right) - q\left(1+\dfrac{1}{M}\right)(1-m)\right)^2}{\left(1+\dfrac{1}{M}\right)\left(p-\dfrac{q^2}{M}\right)}} \tag{4.82}$$

Note that $\lim_{M\to\infty} p = \dfrac{1}{1-m^2}$ and $\lim_{M\to\infty} q = \dfrac{1}{1-m}$ then we have that,

$$\lim_{M\to\infty} D_F(M) = \lim_{M\to\infty} \frac{\sqrt{3}}{1-m}\sqrt{\frac{\left(2\left(1-\frac{q\,m}{M}\right)-q\left(1+\frac{1}{M}\right)(1-m)\right)^2}{\left(1+\frac{1}{M}\right)\left(p-\frac{q^2}{M}\right)}} \tag{4.83}$$

$$= \frac{\sqrt{3}}{1-m}\sqrt{\frac{(2-q(1-m))^2}{p}} \tag{4.84}$$

$$= \frac{\sqrt{3}}{1-m}\sqrt{1-m^2} \tag{4.85}$$

therefore for $m = 0.8$ the function $Y = D_F(M)$ has an asymptote horizontal in $Y = 5.2$ given that,

$$\lim_{M\to\infty} D_F(M) = \frac{\sqrt{3}}{1-0.8}\sqrt{1-0.8^2} = 5.196152422706633 \tag{4.86}$$

5 PEARSON CORRELATION COEFFICIENT

The scores of the Ranked-NoM, NoM and ROC codes are negatively correlated and the Pearson correlation coefficient ρ is the same for the three schemes. They are not independent since permutation of a given set of values implies correlation. That is, the correlation comes from the fact that the choice for, say the last value to be put at the last position of the vector, depends upon the value which has not been put into the vector yet.

The correlation coefficient is defined by

$$\rho = \frac{\mathrm{Cov}(W_i, W_j)}{\sqrt{\mathrm{Var}W_i}\sqrt{\mathrm{Var}W_j}} \tag{5.1}$$

then we calculate ρ for each scheme. For the Ranked-NoM code,

$$\rho_H = \frac{\frac{\mu_W}{M-1}(\mu_W - \frac{2\mathcal{W}+1}{3})}{\left(\sqrt{\mu_W\frac{2\mathcal{W}+1}{3}-\mu_W^2}\right)^2} \tag{5.2}$$

$$= \frac{\frac{1}{M-1}\left(\mu_W - \frac{2\mathcal{W}+1}{3}\right)}{\frac{2\mathcal{W}+1}{3}-\mu_W} \tag{5.3}$$

$$= \frac{-1}{M-1} \tag{5.4}$$

for the ROC code,

$$\rho_R = \frac{\frac{\mu_W}{M-1}(\mu_W - \frac{2M+1}{3})}{\left(\sqrt{\frac{\mu_X}{6}(M-1)}\right)^2} \tag{5.5}$$

$$= \frac{\frac{1}{M-1}\left(\mu_W - \frac{2M+1}{3}\right)}{\frac{1}{6}(M-1)} \tag{5.6}$$

$$= \frac{2(3\mu_W - 2M - 1)}{(M-1)^2} \tag{5.7}$$

$$= \frac{-1}{M-1} \tag{5.8}$$

and for NoM, ρ is defined by,

$$\rho_F = \frac{\mu_W \left(\frac{\mathcal{W}-1}{M-1}\right) - \mu_W^2}{(\sqrt{\mu_W(1-\mu_W)})^2} \tag{5.9}$$

$$= \frac{\left(\frac{\mathcal{W}-1}{M-1}\right) - \mu_W}{1-\mu_W} \tag{5.10}$$

$$= \frac{-1 + \frac{\mathcal{W}}{M}}{(M-1)\left(1 - \frac{\mathcal{W}}{M}\right)} \tag{5.11}$$

$$= \frac{-1}{M-1} \tag{5.12}$$

therefore, $\rho_H = \rho_R = \rho_F = \frac{-1}{M-1}$ thus the correlation vanishes when $M- > \infty$, and it is because the effect of previous choice tends to zero for next choice. So, it is the fact that we make permutations of a given set of values.