FACTORIZED ADABOOST.MH ACHIEVES THE SAME CONVERGENCE RATE AS ADABOOST.MH

Anonymous authors
Paper under double-blind review

ABSTRACT

AdaBoost.MH reduces multi-class classification to a collection of binary subproblems and enjoys the classical boosting-type convergence guarantee under a weak learning condition. A more structured variant, Factorized AdaBoost.MH, uses base classifiers of the form $\boldsymbol{h}(x) = \alpha \boldsymbol{v} \varphi(x)$, where a single binary classifier φ is shared across all classes and the label dependence is carried by a vote vector $\boldsymbol{v} \in \{\pm 1\}^K$. This factorization is algorithmically attractive and achieves better performance in practice, but its convergence depends on whether one can always choose a vote vector with sufficiently large induced binary weight mass. Previous work resolved this question with a lower bound $\max\{1/n, 1/\sqrt{2K}\}$, which still leaves a dimension-dependent slowdown relative to the original AdaBoost.MH analysis. In this paper, we sharpen this combinatorial step and provide an exact characterization. For the minimax quantity $\mathfrak{W}_{n,K}$ governing the factorized edge, we prove $\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}}$, where $C_q = 1$ for $q = 1$, $C_q = q/(3q-4)$ for even $q \geq 2$, and $C_q = (q+1)/(3q-1)$ for odd $q \geq 2$. Since $C_q \downarrow 1/3$, our bounds show that $\mathfrak{W}_{n,K} = \Theta(1)$ uniformly over n and K. Consequently, Factorized AdaBoost.MH achieves the same boosting-type convergence rate as AdaBoost.MH up to a universal constant factor, removing the previously suggested additional dependence on n or K in the number of boosting rounds.

1 INTRODUCTION

Boosting is a central paradigm in supervised learning because it explains how many simple prediction rules can be combined into a highly accurate classifier and has inspired a lot on theoretical analysis and algorithm design in supervised learning (Hanneke, 2016; Montasser et al., 2019). A boosting algorithm adaptively reweights the training sample, asks for a rule that is only slightly better than random guessing under the current weights, and aggregates the returned rules into a strong predictor. This weak-to-strong principle is important both algorithmically and theoretically: it gives practical procedures that are easy to implement, and it provides a clean route from a weak learning condition to explicit training-error and margin guarantees. In the binary case, the classical AdaBoost analysis shows that, whenever every boosting round achieves a positive edge, i.e., has a better performance than random guessing, the exponential loss decreases geometrically (Schapire & Freund, 2012).

Many modern classification tasks have more than two classes, so it is natural to ask whether the same weak-to-strong mechanism remains effective when the predictor must distinguish among K labels rather than two. AdaBoost.MH extends AdaBoost by encoding each labeled example as a vector of K binary label coordinates and by maintaining a weight matrix over example–label pairs (Schapire & Singer, 1999). This construction is a foundational multi-class boosting framework: it turns a multi-class problem into a structured collection of binary decisions and minimizes a weighted Hamming loss through an exponential surrogate. In the standard analysis, the base learner can choose a separate binary decision for each class coordinate, so the empirical weak learning assumption can be applied almost coordinate by coordinate.

The original AdaBoost.MH algorithm represents the output of the weak learner as

$$\boldsymbol{h}(x) = \alpha \boldsymbol{\varphi}(x),$$

where α is a scalar coefficient and $\boldsymbol{\varphi}(x) = [\varphi_1(x), \ldots, \varphi_K(x)]^\top$ maps $\mathcal{X}$ to $\{\pm 1\}^K$. This form treats the K class coordinates as separate binary subproblems. While this independence is convenient

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

for the convergence proof, it may be unnecessarily loose as a modeling assumption: multi-class labels are mutually exclusive, and useful weak rules often capture relations among several classes at once. This motivates more structured weak hypotheses that preserve the boosting philosophy while coupling the class coordinates. In particular, Kégl (2014b) studied the factorized classifier

$$\boldsymbol{h}(x) = \alpha \boldsymbol{v} \varphi(x),$$

where α is a scalar coefficient, $\boldsymbol{v} \in \{\pm 1\}^K$ is an input-independent vote vector over classes, and $\varphi : \mathcal{X} \to \{\pm 1\}$ is a single binary classifier shared by all labels. This form can be viewed as decomposing a vector-valued weak rule into a class-side vote vector and an input-side binary split. It avoids training K unrelated weak rules at each boosting round and can encode multi-class structure through the vote vector. Empirically, such factorized weak classifiers are attractive and achieve better performance (Kégl, 2014b), but their theoretical analysis is more delicate: once $\boldsymbol{v}$ is fixed, all label coordinates must be compressed into one induced binary problem on the original examples, so the standard coordinate-wise proof for AdaBoost.MH no longer applies.

The key quantity in this compression is the total induced weight

$$w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) = \|(\boldsymbol{W} \odot \boldsymbol{Y})\boldsymbol{v}\|_1 ,$$

where $\boldsymbol{W}$ is the current normalized weight matrix and $\boldsymbol{Y}$ is the signed one-hot label matrix. If this quantity is bounded below, then the standard binary weak-learning assumption can be applied to the induced pseudo-labeled sample, yielding a positive factorized edge and hence a decrease in the exponential loss. The central obstruction is therefore purely combinatorial: for every possible weight matrix and label matrix, must there exist a vote vector $\boldsymbol{v}$ for which $w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v})$ is bounded away from zero? Kégl (2014a) posed this as an open problem.

Zou et al. (2024) answered this question positively by proving a lower bound for the associated minimax value

$$\mathfrak{W}_{n,K} = \min_{\boldsymbol{W},\boldsymbol{Y}} \max_{\boldsymbol{v}\in\{\pm 1\}^K} \|(\boldsymbol{W} \odot \boldsymbol{Y})\boldsymbol{v}\|_1 .$$

Their result shows that $\mathfrak{W}_{n,K} \geq \max\{1/n, 1/\sqrt{2K}\}$, which is enough to establish boosting-type convergence for Factorized AdaBoost.MH whenever n and K are not simultaneously large. Nevertheless, the bound leaves open the possibility that factorization intrinsically costs a factor depending on the number of examples or classes. Indeed, plugging this bound into the standard exponential loss analysis gives a sufficient number of boosting rounds larger than that of AdaBoost.MH by a factor of order $\min\{n^2, K\}$.

This paper shows that this dimension-dependent slowdown is an artifact of the previous lower bound. We prove a constant-order characterization of $\mathfrak{W}_{n,K}$. Specifically, for all $n \geq 1$ and $K \geq 2$, we show

$$\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}},$$

where

$$C_q = \begin{cases} 1, & q = 1, \\ \frac{q}{3q-4}, & q \geq 2 \text{ and } q \text{ is even}, \\ \frac{q+1}{3q-1}, & q \geq 2 \text{ and } q \text{ is odd}. \end{cases}$$

The sequence C_q decreases to $1/3$, and therefore $\mathfrak{W}_{n,K} \in (1/3, 1]$ for all $n \geq 1$ and $K \geq 2$. Our upper and lower bounds are **optimal** and **match exactly** at $C_{\min\{n+1,K\}}$.

Our proof separates the lower-bound argument into two complementary choices of vote vectors. The all-one vote vector controls the imbalance between the total weight on true classes and the total weight on false classes. Balanced or nearly balanced random vote vectors, depending on the parity of K, control the remaining regime by forcing each row to retain a constant fraction of its mass after the factorized reduction. Optimizing over the resulting one-dimensional parameter gives the constants C_K. Based on the lower bound C_K, we further find that the effective class number of the problem is actually $\min\{n + 1, K\}$ and get a better lower bound $C_{\min\{n+1,K\}}$ by more fine-grained constructions. For the upper bound, we construct explicit worst-case weight and label matrices supported on $\min\{n, K\}$ rows and columns, and show that no vote vector can exceed $C_{\min\{n,K\}}$ on this construction. Moreover, we make another construction according to the effective class number conjecture and get a complementary upper bound. Combining the above upper and lower bounds lead to the exact characterization of $\mathfrak{W}_{n,K}$.

As a consequence, Factorized AdaBoost.MH does not lose the boosting-type convergence rate of AdaBoost.MH except for a universal constant factor. Under the same empirical δ-weak learning condition used in the standard analysis, the factorized algorithm needs only $O(\log(nK)/\delta^2)$ boosting rounds to achieve 100% training accuracy, which is of the same order as the original AdaBoost.MH algorithm. Thus the structured weak classifiers studied here preserve the essential weak-to-strong behavior of AdaBoost.MH while allowing the base learner to use a more coupled multi-class form.

The remainder of this article is structured as follows: Section 2 introduces notations and setups of the task we study. Section 3 presents our main theorems and the exact characterization of $\mathfrak{W}_{n,K}$. Section 4 shows the proof sketch of the lower bounds. The conclusions are presented in the last section. Due to space limitation, related works are defered to Appendix A. Detailed proofs can be found in Appendices B, C, and D.

2 Problem Setup

Notations. For any positive integer m, we denote $[m] = \{1, \dots, m\}$. We use bold lower-case letters for vectors and bold upper-case letters for matrices. For a matrix $\boldsymbol{A}$, $\boldsymbol{A}_{ij}$ denotes its (i,j)-th entry and $\boldsymbol{A}_i$ denotes its i-th row. We use $\mathbf{1}[\cdot]$ for the indicator function. The symbol $\odot$ denotes entrywise multiplication, and $\|\cdot\|_1$ denotes the vector ℓ_1 norm.

We consider a multi-class classification problem where the input space is $\mathcal{X}$ and the output space is $\mathcal{Y} = [K]$, where $K \geq 2$. We consider n training examples with K classes. Let $\boldsymbol{Y} \in \{\pm 1\}^{n \times K}$ be the signed one-hot label matrix, where $\boldsymbol{Y}_{i,j} = +1$ if the i-th training example belongs to class j, and $\boldsymbol{Y}_{i,j} = -1$ otherwise. Formally, if the true label for the i-th training example is y_i, then for $j \in [K]$,

$$\boldsymbol{Y}_{i,j} = \begin{cases} +1 & \text{if } j = y_i \\ -1 & \text{if } j \neq y_i. \end{cases}$$

AdaBoost.MH returns a vector-valued discriminant function $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^K$ with a combined predictor $\tilde{\boldsymbol{f}} : \mathcal{X} \to \{-1, +1\}^K$ where $\tilde{\boldsymbol{f}}(x)_l = \operatorname{sign}(\boldsymbol{f}(x)_l)$ for $l \in [K]$. Here we define

$$\operatorname{sign}(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0. \end{cases}$$

Given the dataset $S = \{(x_i, \boldsymbol{y}_i)\}_{i=1}^n$ where $\boldsymbol{y}_i$ is the i-th row of $\boldsymbol{Y}$, the aim is to learn a predictor $\tilde{\boldsymbol{f}}$ such that $\tilde{\boldsymbol{f}}(x_i) = \boldsymbol{y}_i$ for all $i \in [n]$. To achieve this, AdaBoost.MH (Schapire & Singer, 1999) treats a K-class classification problem as K binary classification problems and adapts the AdaBoost (Schapire & Freund, 2012) algorithm. Specifically, let $\boldsymbol{W} \in [0,1]^{n \times K}$ be a normalized weight matrix where $\sum_{i=1}^n \sum_{j=1}^K \boldsymbol{W}_{ij} = 1$, the aim is to minimize the weighted Hamming loss:

$$R_{\mathrm{H}}(\tilde{\boldsymbol{f}}, \boldsymbol{W}) \coloneqq \sum_{i=1}^n \sum_{j=1}^K \boldsymbol{W}_{ij} \mathbf{1}\left[\tilde{\boldsymbol{f}}(x_i)_j \neq \boldsymbol{Y}_{ij}\right], \tag{2.1}$$

where $\mathbf{1}[\cdot]$ is the indicator function. It is easy to see that $\mathbf{1}\left[\tilde{\boldsymbol{f}}(x_i)_j \neq \boldsymbol{Y}_{ij}\right] = \mathbf{1}\left[\boldsymbol{f}(x_i)_j \cdot \boldsymbol{Y}_{ij} \leq 0\right] \leq \exp\left(-\boldsymbol{f}(x_i)_j \cdot \boldsymbol{Y}_{ij}\right)$, so the weighted Hamming loss is upper bounded by the weighted exponential loss

$$R_{\mathrm{EXP}}(\boldsymbol{f}, \boldsymbol{W}) \coloneqq \sum_{i=1}^n \sum_{j=1}^K \boldsymbol{W}_{ij} \exp\left(-\boldsymbol{f}(x_i)_j \cdot \boldsymbol{Y}_{ij}\right). \tag{2.2}$$

The pseudocode for AdaBoost.MH is shown in Algorithm 1. According to (Schapire & Freund, 2012, Chapter 7), AdaBoost.MH directly minimizes the weighted exponential loss and outputs a combined predictor $\boldsymbol{f}^{(T)}(\cdot)$, where $\boldsymbol{f}^{(T)}(x) = \sum_{t=1}^T \boldsymbol{h}^{(t)}(x)$ is a sum of T base predictors $\boldsymbol{h}^{(t)} : \mathcal{X} \to \mathbb{R}^K$ returned by a base learner $A(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{W}^{(t)})$, where $\boldsymbol{X}_i = x_i$ and $\boldsymbol{W}^{(t)}$ is the weight matrix for the

t-th iteration. According to the calculation in (Schapire & Freund, 2012, Proof of Theorem 3.1), given $\boldsymbol{X}$ and $\boldsymbol{Y}$, it is easy to know that:

$$R_{\mathrm{EXP}}(\boldsymbol{f}^{(T)}, \boldsymbol{W}) = \prod_{t=1}^{T} \underbrace{\left(\sum_{i=1}^{n} \sum_{j=1}^{K} \boldsymbol{W}_{ij}^{(t)} \exp\left(-\boldsymbol{h}^{(t)}(x_i)_j \cdot \boldsymbol{Y}_{ij} \right) \right)}_{:= Z(\boldsymbol{h}^{(t)}, \boldsymbol{W}^{(t)})} = \prod_{t=1}^{T} Z(\boldsymbol{h}^{(t)}, \boldsymbol{W}^{(t)}). \quad (2.3)$$

Algorithm 1: AdaBoost.MH

Input : Training data $\boldsymbol{X}, \boldsymbol{Y}$; initialization weight matrix $\boldsymbol{W}$; base learner $A(\cdot, \cdot, \cdot)$; number of iterations T;

$\boldsymbol{W}^{(1)} = \boldsymbol{W}$;
for $t \leftarrow 1$ to T **do**
  $\boldsymbol{h}^{(t)}(\cdot) \leftarrow A(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{W}^{(t)})$;
  for $i \leftarrow 1$ to n **do**
    for $j \leftarrow 1$ to K **do**
      $\boldsymbol{W}_{ij}^{(t+1)} \leftarrow \boldsymbol{W}_{ij}^{(t)} \cdot \frac{\exp\left(-\boldsymbol{h}^{(t)}(x_i)_j \cdot \boldsymbol{Y}_{ij}\right)}{\sum_{i'=1}^{n} \sum_{j'=1}^{K} \boldsymbol{W}_{i'j'}^{(t)} \exp\left(-\boldsymbol{h}^{(t)}(x_{i'})_{j'} \cdot \boldsymbol{Y}_{i'j'}\right)}$;
    end
  end
end

Output : $\boldsymbol{f}^{(T)}(\cdot) = \sum_{t=1}^{T} \boldsymbol{h}^{(t)}(\cdot)$;

Based on Equation (2.3), we know that to minimize $R_{\mathrm{EXP}}(\boldsymbol{f}^{(T)}, \boldsymbol{W})$, the base learner $A(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{W}^{(t)})$ should find a $\boldsymbol{h}^{(t)}$ to minimize $Z(\boldsymbol{h}^{(t)}, \boldsymbol{W}^{(t)})$.

In the original AdaBoost.MH algorithm, Schapire & Singer (1999) consider modeling $\boldsymbol{h}$ as the product of a scalar $\alpha \in \mathbb{R}$ and a vectorized mapping $\boldsymbol{\varphi} : \mathcal{X} \to \{-1, +1\}^K$, i.e., $\boldsymbol{h}(x) = \alpha \boldsymbol{\varphi}(x)$. It can be shown (Schapire & Singer, 1999; Zou et al., 2024) that under a standard empirically δ-weak learning condition (Assumption 2.1), AdaBoost can reach the following upper bound for the weighted exponential loss:

$$R_{\mathrm{EXP}}(\boldsymbol{f}^{(T)}, \boldsymbol{W}) \leq \exp\left(-\frac{\delta^2}{2} T \right).$$

Assumption 2.1 (empirically δ-weak learing condition, Definition 2.1 of (Zou et al., 2024))**.** *For a given binary dataset $\{(x_1, y_1), \ldots, (x_m, y_m)\}$ where $y_i \in \{-1, +1\}$, we assume that for $\delta > 0$, for any distribution $\boldsymbol{w} \in \Delta^{m-1} := \left\{ \boldsymbol{\lambda} \in \mathbb{R}^m \middle| \boldsymbol{\lambda}_i \geq 0 \, \forall i \in [m], \sum_{i=1}^{m} \boldsymbol{\lambda}_i = 1 \right\}$ over $[m]$, we can always find a binary classifier $\varphi : \mathcal{X} \to \{-1, +1\}$ such that:*

$$\gamma = \sum_{i=1}^{m} \boldsymbol{w}_i \cdot y_i \cdot \varphi(x_i) \geq \delta.$$

AdaBoost.MH reduces the multi-class problem into K binary one-against-all classifications, however, Kégl (2014b) avoids such a reduction by factorizing the vector-valued classifier $\boldsymbol{h}$ into an input-independent vector of length K and a label-independent scalar classifier. Formally, Kégl (2014b) sets

$$\boldsymbol{h}(x) = \alpha \boldsymbol{v} \varphi(x),$$

where $\alpha \in \mathbb{R}_+$ is a positive real-valued base coefficient, $\boldsymbol{v} \in \{-1, +1\}^K$ is an input-independent vote vector of length K, and $\varphi : \mathcal{X} \to \{-1, +1\}$ is a label-independent binary classifier.

Now we derive the corresponding exponential-loss upper bound. We focus on one boosting iteration and omit the superscript t for readability. For a fixed triple $(\alpha, \boldsymbol{v}, \varphi)$, define the factorized edge

$$\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W}) := \sum_{i=1}^{n} \sum_{j=1}^{K} \boldsymbol{W}_{ij} \cdot \boldsymbol{Y}_{ij} \cdot \boldsymbol{v}_j \cdot \varphi(x_i). \quad (2.4)$$

This quantity is the weighted agreement between the factorized base classifier and the signed labels: a pair (i, j) contributes positively when $\boldsymbol{v}_j\varphi(x_i)$ has the same sign as $\boldsymbol{Y}_{ij}$, and negatively otherwise. According to the similar derivation in Zou et al. (2024), since $\boldsymbol{v}_j\varphi(x_i)\boldsymbol{Y}_{ij} \in \{-1, +1\}$ and $\sum_{i,j} \boldsymbol{W}_{ij} = 1$, we have

$$\begin{aligned} Z(\boldsymbol{h}, \boldsymbol{W}) &= \sum_{i=1}^{n}\sum_{j=1}^{K} \boldsymbol{W}_{ij} \exp\left(-\alpha \boldsymbol{v}_j \varphi(x_i)\boldsymbol{Y}_{ij}\right) \\ &= \frac{1+\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W})}{2} e^{-\alpha} + \frac{1-\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W})}{2} e^{\alpha}. \end{aligned} \tag{2.5}$$

Therefore, after fixing $\boldsymbol{v}$ and φ, minimizing over α gives

$$\alpha = \frac{1}{2}\ln\left(\frac{1+\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W})}{1-\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W})}\right), \qquad Z(\boldsymbol{h}, \boldsymbol{W}) = \sqrt{1-\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W})^2}.$$

Thus the factorized base learner should find $\boldsymbol{v}$ and φ with a large positive edge. If for every iteration there is a uniform lower bound $\gamma(\boldsymbol{v}^{(t)}, \varphi^{(t)}, \boldsymbol{W}^{(t)}) \geq \delta$, then by Equation (2.3),

$$R_{\mathrm{EXP}}(\boldsymbol{f}^{(T)}, \boldsymbol{W}) \leq \prod_{t=1}^{T} \sqrt{1-\delta^2} \leq \exp\left(-\frac{\delta^2}{2}T\right). \tag{2.6}$$

The remaining difficulty is to justify such a positive lower bound on the factorized edge. In the unfactorized case, one may choose a different binary classifier for each label j. In the factorized case, the same scalar classifier φ must serve all labels, so Assumption 2.1 cannot be applied to each column of $\boldsymbol{Y}$ independently. Following the reduction described in Kégl (2014a); Zou et al. (2024), fix a vote vector $\boldsymbol{v}$ and rewrite Equation (2.4) by collecting all label contributions that belong to the same training example, we define

$$w_i^+ \coloneqq \sum_{j=1}^{K} \boldsymbol{W}_{ij}\mathbf{1}[\boldsymbol{v}_j\boldsymbol{Y}_{ij} = +1], \qquad w_i^- \coloneqq \sum_{j=1}^{K} \boldsymbol{W}_{ij}\mathbf{1}[\boldsymbol{v}_j\boldsymbol{Y}_{ij} = -1].$$

Then

$$\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W}) = \sum_{i=1}^{n}\varphi(x_i)\sum_{j=1}^{K}\boldsymbol{W}_{ij}\boldsymbol{v}_j\boldsymbol{Y}_{ij} = \sum_{i=1}^{n}\varphi(x_i)(w_i^+ - w_i^-). \tag{2.7}$$

This turns the multi-class weighted problem into an ordinary binary weighted problem on the original examples. Let

$$y_i' \coloneqq \mathrm{sign}(w_i^+ - w_i^-), \qquad w_i' \coloneqq |w_i^+ - w_i^-|, \qquad w_\Sigma' \coloneqq \sum_{i=1}^{n} w_i'.$$

Using these pseudo-labels and pseudo-weights, Equation (2.7) becomes

$$\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W}) = \sum_{i=1}^{n} w_i' y_i' \varphi(x_i). \tag{2.8}$$

However, the vector $(w_1', \ldots, w_n')$ is not necessarily a distribution because w_Σ' can be smaller than 1. If $w_\Sigma' > 0$, then after normalization, Assumption 2.1 gives a binary classifier φ such that

$$\sum_{i=1}^{n} \frac{w_i'}{w_\Sigma'} y_i' \varphi(x_i) \geq \delta.$$

Combining this with Equation (2.8) yields

$$\gamma(\boldsymbol{v}, \varphi, \boldsymbol{W}) \geq \delta w_\Sigma', \tag{2.9}$$

which leads to an upper bound for the weighted exponential loss:

$$R_{\mathrm{EXP}}(\boldsymbol{f}^{(T)}, \boldsymbol{W}) \leq \exp\left(-\frac{\delta^2 (w_\Sigma')^2}{2}T\right).$$

Consequently, to obtain the exponential-loss bound in Equation (2.6) for Factorized AdaBoost.MH, it remains to choose a vote vector $\boldsymbol{v}$ for which w'_Σ has a nontrivial lower bound. The next step is therefore to prove a lower bound on w'_Σ that holds uniformly over the current weight matrix $\boldsymbol{W}$ and label matrix $\boldsymbol{Y}$. Regarding this step, Kégl (2014a) raises an open problem:

> Does there exist a setup $(\boldsymbol{X}, \boldsymbol{W}, \boldsymbol{Y})$ such that all 2^K possible vote vectors $\boldsymbol{v} \in \{-1, +1\}^K$ lead to arbitrarily small, or even zero, w'_Σ? Or can one find a constant (independent of n) lower bound $\omega > 0$, independent of the sample size n, such that for at least one vote vector $\boldsymbol{v}$ and classifier φ, $w'_\Sigma \geq \omega$ holds?

Since w'_Σ is fully decided by $\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}$, we write $w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v})$ to explicitly show such a relationship. Zou et al. (2024) reformulate the above open problem as finding a lower bound for

$$\begin{aligned} \mathfrak{W}_{n,K} &\coloneqq \min_{\boldsymbol{W} \in \mathcal{W}_{n,K},\, \boldsymbol{Y} \in \mathcal{Y}_{n,K}} \max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \\ &= \min_{\boldsymbol{W} \in \mathcal{W}_{n,K},\, \boldsymbol{Y} \in \mathcal{Y}_{n,K}} \max_{\boldsymbol{v} \in \mathcal{V}_K} \left\| (\boldsymbol{W} \odot \boldsymbol{Y}) \boldsymbol{v} \right\|_1, \end{aligned} \tag{2.10}$$

where

$$\mathcal{W}_{n,K} \coloneqq \left\{ \boldsymbol{W} \in \mathbb{R}^{n \times K} \;\middle|\; \boldsymbol{W}_{ij} \geq 0,\ \sum_{i=1}^{n} \sum_{j=1}^{K} \boldsymbol{W}_{ij} = 1 \right\}$$

is the set of all normalized weight matrices,

$$\mathcal{Y}_{n,K} \coloneqq \left\{ \boldsymbol{Y} \in \{-1, +1\}^{n \times K} \;\middle|\; \text{each row of } \boldsymbol{Y} \text{ is a signed one-hot label vector} \right\}$$

is the set of all possible signed one-hot label matrices, and $\mathcal{V}_K \coloneqq \{-1, +1\}^K$ is the set of all vote vectors. The notation $\odot$ denotes entrywise multiplication and $\| \cdot \|_1$ denotes the vector ℓ_1 norm.

Zou et al. (2024, Theorems 3.3-3.4) solve the above open problem by proving that $\mathfrak{W}_{n,K} \geq \max\left\{ \frac{1}{n}, \frac{1}{\sqrt{2K}} \right\}$. However, it is unclear whether the lower bound shown by Zou et al. (2024) is tight. In this paper, we show that the lower bound in (Zou et al., 2024) is not tight, and provide an **exact characterization** of $\mathfrak{W}_{n,K}$.

3 Main Theorems

In this section, we present our main result, the exact characterization of $\mathfrak{W}_{n,K}$. We first take a step to provide a better (compared to those in (Zou et al., 2024)) lower bound for $\mathfrak{W}_{n,K}$, and then show the exact upper and lower bounds for $\mathfrak{W}_{n,K}$ based on our initial bounds.

Theorem 3.1 (Upper and Lower Bounds for $\mathfrak{W}_{n,K}$)**.** *For all integers $n \geq 1$ and $K \geq 2$, we have:*

$$\max\left\{ \frac{1}{n}, C_K \right\} \leq \mathfrak{W}_{n,K} \leq C_{\min\{n,K\}}, \tag{3.1}$$

where

$$C_q = \begin{cases} 1 & q = 1 \\ \frac{q}{3q-4} & q \geq 2 \text{ and } q \text{ is even}, \\ \frac{q+1}{3q-1} & q \geq 2 \text{ and } q \text{ is odd}. \end{cases} \tag{3.2}$$

Proposition 3.2 (Properties of C_q)**.** *The sequence C_q defined in Equation* (3.2) *has the following properties:*

1. *C_q is non-increasing with respect to q.*
2. *If q is odd, then $C_{q+1} = C_q$.*
3. *C_q tends to $\frac{1}{3}$ as q tends to infinity.*

Theorem 3.1 provides a lower bound $\mathfrak{W}_{n,K} \geq C_K$. When $n \gg K$, then the n examples are likely to contain all K labels. However, when $n < K$, the n examples can not use up all K labels. So we conjecture that there might be **effective class number** that really captures the intrinsic hardness caused by the labels. Based on this conjecture, we narrow down the effect of label numbers to $n+1$ (rather than K shown in Theorem 3.1) when $n+1 < K$ and get the following refined lower bound.

Theorem 3.3 (Refined Lower Bound for $\mathfrak{W}_{n,K}$)**.** *For all integers $n \geq 1$ and $K \geq 2$, we have:*

$$\mathfrak{W}_{n,K} \geq C_{\min\{n+1,K\}}. \tag{3.3}$$

Based on the **effective class number** conjecture, we further construct a new upper bound as follows, which makes up the exact upper bound together with the upper bound in Theorem 3.1.

Theorem 3.4 (Another Upper Bound for $\mathfrak{W}_{n,K}$)**.** *For integers $n \geq 1$ and $K \geq 2$, when n is even and $n+1 \leq K$, we have:*

$$\mathfrak{W}_{n,K} \leq C_{n+1}. \tag{3.4}$$

Combining Theorems 3.1, 3.3 and 3.4, we get **optimal** upper and lower bounds, leading to the following **exact characterization** of $\mathfrak{W}_{n,K}$.

Theorem 3.5 (Exact Value of $\mathfrak{W}_{n,K}$)**.** *For integers $n \geq 1$ and $K \geq 2$, we have:*

$$\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}}. \tag{3.5}$$

Remark 3.6. *Theorem 3.5 tells us that,*

$$\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}}.$$

By Proposition 3.2, we know that $C_q \leq 1$ and decreases to $\frac{1}{3}$ as q tends to infinity, it is easy to see that:

$$\frac{1}{3} < C_K \leq \mathfrak{W}_{n,K} = C_{\min\{n+1,K\}} \leq 1,$$

which means that $\mathfrak{W}_{n,K} = \Theta(1)$ and is a great improvement upon the results in (Zou et al., 2024). When $n, K \to \infty$, we have $\mathfrak{W}_{n,K} \to \frac{1}{3}$.

Let's now compare the results in (Zou et al., 2024) and those in this paper. The bounds in (Zou et al., 2024) are

$$\max\left\{\frac{1}{n}, \frac{1}{\sqrt{2K}}\right\} \leq \mathfrak{W}_{n,K} \leq 1. \tag{3.6}$$

Zou et al. (2024) claim that $\mathfrak{W}_{n,K} > 0$ is guaranteed when n and K do not tend to infinity simultaneously. However, this paper shows a much more stronger result: $\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}} \in \left(\frac{1}{3}, 1\right]$ whenever $n \geq 1$ and $K \geq 2$.

If we set the original input matrix in Algorithm 1 as $\boldsymbol{W}_{ij} = \frac{1}{nK}$ for all $i \in [n], j \in [K]$, then by making the upper bound of the exponential loss less than $\frac{1}{nK}$, we can get that

$$T > \frac{2\log(nK)}{\delta^2}.$$

This means that if we run the original AdaBoost.MH algorithm for $T_{\text{ori}} \coloneqq \left\lceil \frac{2\log(nK)}{\delta^2} \right\rceil + 1$ steps, then can make sure the combined predictor correctly classify all training data.

Similarly, in the factorized case, solving the equation $\exp\left(-\frac{\delta^2\left(w'_\Sigma\right)^2}{2}T\right) < \frac{1}{nK}$ through the lower bound given by Zou et al. (2024), we can get

$$T_{\text{fac}} \coloneqq \left\lceil \frac{2\log(nK)}{\delta^2\left(w'_\Sigma\right)^2} \right\rceil + 1. \tag{3.7}$$

Equation (3.7) shows that if we run Factorized AdaBoost.MH for T_{fac} steps, then the combined (factorized) predictor can correctly classify all training data. Combining the lower bound in (Zou et al., 2024), we show that

$$T_{\text{fac},1} \coloneqq \left\lceil \frac{2\log(nK) \cdot \min\{n^2, 2K\}}{\delta^2} \right\rceil + 1 \tag{3.8}$$

steps are sufficient to make sure the convergence of the Factorized AdaBoost.MH algorithm. Compare T_{ori} and $T_{\text{fac},1}$, we find that according to the results in (Zou et al., 2024), the iteration number of Factorized AdaBoost.MH is about $\min\{n^2, 2K\}$ times more than that of the original AdaBoost.MH algorithm. When K and n are large, the $\min\{n^2, 2K\}$ term shows that Factorized AdaBoost.MH may need a lot more steps to converge.

However, our results show that the bounds in (Zou et al., 2024) are not tight, and the additional $\min\{n^2, 2K\}$ term is not needed. Specifically, our tight lower bound for $\mathfrak{W}_{n,K}$ shows that

$$T_{\text{fac},2} \coloneqq \left\lceil \frac{18\log(nK)}{\delta^2} \right\rceil + 1 \tag{3.9}$$

steps are enough to make sure the Factorized AdaBoost.MH algorithm converges. Compared with T_{ori} and Equation (3.8), our Equation (3.9) tells us that for the AdaBoost.MH algorithm, **we can use factorized classifiers without loss of the convergence rate**.

4 Proof Sketch of the Lower Bound

The proof of the lower bound consists of two parts. The first part is the proof of the lower bound in Theorem 3.1, the second part uses a more fine-graind construction based on the effective class number conjecture. Due to space limitation, in this section, we provide the proof sketch for the lower bounds in Theorem 3.1. For complete proof details, please refer to Appendix B.

According to Theorem 3.3 in (Zou et al., 2024), we have $\mathfrak{W}_{n,K} \geq \frac{1}{n}$. So it remains to prove $\mathfrak{W}_{n,K} \geq C_K$. Before presenting the main proof steps for Theorem 3.1, we first introduce some definitions for convenience.

Fix any $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, for any $i \in [n]$, let $c_i \in [K]$ be the column unique index such that $\boldsymbol{Y}_{ic_i} = +1$. We define the row mass for the i-th row as

$$s_i = \sum_{j=1}^{K} \boldsymbol{W}_{ij}.$$

Rows with $s_i = 0$ do not affect w'_Σ, so we consider the rows with $s_i > 0$. For rows with $s_i > 0$, we define the true-class weight fraction as

$$a_i = \frac{\boldsymbol{W}_{ic_i}}{s_i}.$$

It is easy to see that $0 \leq a_i \leq 1$ and $\sum_{i=1}^{n} s_i = 1$. Then the total weight for the true classes is

$$\rho = \sum_{i=1}^{n} s_i a_i = \sum_{i=1}^{n} \boldsymbol{W}_{ic_i}.$$

Since $\sum_{i=1}^{n} s_i = 1$ and $0 \leq a_i \leq 1$ for all $i \in [n]$, we know that $0 \leq \rho \leq 1$. Fix $\boldsymbol{W}, \boldsymbol{Y}$, we define

$$Z_i(\boldsymbol{v}) = \sum_{j=1}^{K} \boldsymbol{W}_{ij}\boldsymbol{Y}_{ij}\boldsymbol{v}_j,$$

then, our goal is to make sure that make sure there exists a vector $\boldsymbol{v}$ such that $w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) = \sum_{i=1}^{n} |Z_i(\boldsymbol{v})|$ is away from zero. Our main strategy is to construct two different types of vector sets so that $Z_i(\boldsymbol{v})$ cannot simultaneously be small for the two sets.

The first type of vector we consider is the all-one vector $\boldsymbol{v}^1 = (+1, \ldots, +1)$. For $\boldsymbol{v}^1$, $|Z_i(\boldsymbol{v}^1)| = \left|\boldsymbol{W}_{ic_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij}\right|$, i.e., the difference between the true label mass and the false label mass. To make $|Z_i(\boldsymbol{v}^1)|$ small, the mass of the true label and false labels should be close, making $\rho \approx \frac{1}{2}$. Our following lemma formulates such an intuition.

Lemma 4.1 (All-one Vector). *Fix integers $n \geq 1$ and $K \geq 2$, for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq |2\rho - 1|. \tag{4.1}$$

Other than the all-one vector, we need to construct another type of vectors which can make $\sum_{i=1}^{n} |Z_i(\boldsymbol{v})|$ away from zero when the all-one vector fails to do so (i.e., when $\rho \approx \frac{1}{2}$). Since $|Z_i(\boldsymbol{v})| = \left| \boldsymbol{W}_{ic_i} \boldsymbol{v}_{c_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij} \boldsymbol{v}_j \right|$, conditioned on the fact that $\boldsymbol{W}_{ic_i} \approx \sum_{j \neq c_i} \boldsymbol{W}_{ij}$ (i.e., the fact that all-one vector fails), a natural idea to make this term away from zero is to set balanced vectors $\boldsymbol{v}$ so that the positive and negative contributions in $\sum_{j \neq c_i} \boldsymbol{W}_{ij} \boldsymbol{v}_j$ cancel each other as much as possible. The following two lemmas formulate the above intuition and construct a balanced set and a nearly balanced set based on the parity of K.

Lemma 4.2 (Balanced Vector Set for Even K)**.** *Fix integers $n \geq 1$ and $K \geq 2$, when K is **even**, then for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \rho + \frac{1-\rho}{K-1}. \tag{4.2}$$

Lemma 4.3 (Nearly Balanced Vector Set for Odd K)**.** *Fix integers $n \geq 1$ and $K \geq 3$, when K is **odd**, then for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \rho + \frac{1-\rho}{K}. \tag{4.3}$$

Let's take the even case as an example. Combining Lemma 4.1 and Lemma 4.2, we know that when ρ is away from $\frac{1}{2}$, i.e., $|2\rho - 1| > 0$, then we can get $\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq |2\rho - 1|$ according to Equation (4.1); when ρ is close to $\frac{1}{2}$, then according to Equation (4.2) we know that $\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \rho + \frac{1-\rho}{K-1} \approx \frac{1}{2} + \frac{1}{2(K-1)} > \frac{1}{2}$. So our strategy can make sure at least one of the two kinds of vectors make $w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v})$ away from zero. Putting Lemmas 4.1 and 4.2 together, we know that when K is even and $K \geq 2$,

$$\mathfrak{W}_{n,K} \geq \min_{\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}} \max \left\{ |2\rho - 1|, \rho + \frac{1-\rho}{K-1} \right\} \geq \min_{\rho \in [0,1]} \max \left\{ |2\rho - 1|, \rho + \frac{1-\rho}{K-1} \right\}.$$

Minimizing $\min_{\rho \in [0,1]} \max \left\{ |2\rho - 1|, \rho + \frac{1-\rho}{K-1} \right\}$ provides our lower bound for even K. For complete and formal arguments, please refer to Appendix B.1.

5 Conclusion

In this paper, we revisited the convergence analysis of Factorized AdaBoost.MH through the combinatorial quantity that controls the induced binary weight mass after the factorized reduction. Previous guarantees showed that this quantity is always positive, but still allowed a dimension-dependent loss in the number of boosting rounds. We proved a sharper characterization: for all $n \geq 1$ and $K \geq 2$, the minimax value $\mathfrak{W}_{n,K}$ is bounded below by $\max\{1/n, C_K\}$ and above by $C_{\min\{n,K\}}$, where the constants C_q are explicit and converge to $1/3$. Thus the relevant factorized edge is uniformly of constant order. As a consequence, Factorized AdaBoost.MH retains the boosting-type convergence rate of AdaBoost.MH up to a universal constant factor under the same empirical weak learning condition. This closes the remaining gap in the convergence analysis of factorized multi-class weak classifiers and supports their use as a structured alternative to independent one-against-all weak rules.

AI use statement

In this work, we used generative AI tools for provide critical ingredients for proving mathematical claims. Concretely, we use generative AI tools providing some proof ideas for Theorem 3.3 and Theorem 3.4, which improves from almost optimal upper and lower bounds (Theorem 3.1) to the exact value of w'_Σ (Theorem 3.5). We have not used generative AI tools for help develop theoretical models or conceptual frameworks, formulate mathematical claims, or assist in the writing of proofs. Since this paper is pure theoretical, the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for improving the readability of this paper. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Steve Hanneke. The optimal sample complexity of PAC learning. *J. Mach. Learn. Res.*, 17:38:1–38:15, 2016.

Balázs Kégl. Open problem: A (missing) boosting-type convergence result for adaboost.mh with factorized multi-class classifiers. In *COLT*, 2014a.

Balázs Kégl. The return of adaboost.mh: multi-class hamming trees. In *ICLR*, 2014b.

Omar Montasser, Steve Hanneke, and Nathan Srebro. VC classes are adversarially robustly learnable, but only improperly. In *COLT*, pp. 2512–2530, 2019.

Robert E. Schapire and Yoav Freund. *Boosting: Foundations and Algorithms*. The MIT Press, 05 2012.

Robert E. Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. *Mach. Learn.*, 37(3):297–336, 1999.

Xin Zou, Zhengyu Zhou, Jingyuan Xu, and Weiwei Liu. A boosting-type convergence result for adaboost.mh with factorized multi-class classifiers. In *NeurIPS*, 2024.

A RELATED WORKS

Binary boosting and weak-to-strong learning. Boosting originated as a constructive answer to the question of how a weak learning procedure can be converted into a strong one. AdaBoost is the most prominent representative of this idea: it repeatedly changes the distribution over training examples, calls a weak learner, and combines the returned rules with data-dependent coefficients. Its classical analysis connects three important viewpoints: adaptive reweighting, minimization of exponential loss, and geometric decay of the training error under a positive-edge condition (Schapire & Freund, 2012). This combination of a simple algorithm and a sharp theoretical guarantee is one reason boosting remains a basic tool for studying supervised learning algorithms (Hanneke, 2016; Montasser et al., 2019).

Multi-class boosting and AdaBoost.MH. The multi-class setting introduces an additional structural question: should one reduce the task to several binary problems, or should the weak learner operate directly on class-vector predictions? AdaBoost.MH follows the first route in a particularly influential way. It represents each label by a signed vector, places weights on example–label pairs, and uses an exponential surrogate for the weighted Hamming loss (Schapire & Singer, 1999). This framework keeps the binary AdaBoost proof mechanism largely intact because a vector-valued weak rule may be regarded as a collection of class-coordinate binary rules. As a result, AdaBoost.MH serves as a natural bridge from binary boosting to multi-class boosting.

Structured and factorized weak classifiers. Although the one-against-all view used in the original AdaBoost.MH algorithm is mathematically convenient, it can ignore the dependencies among class labels. Kégl (2014b) proposed to use more structured multi-class weak learners and introduced factorized rules of the form $\boldsymbol{h}(x) = \alpha \boldsymbol{v} \varphi(x)$. In this representation, the scalar classifier φ determines an input-dependent split, while the vote vector $\boldsymbol{v}$ assigns signs to the classes. This gives a compact way to share one weak decision across all labels and is closely connected to the multi-class Hamming-tree viewpoint (Kégl, 2014b). The benefit of this structure is that it provides a more coupled and potentially more efficient base learner; the cost is that the standard AdaBoost.MH convergence argument cannot be applied to the factorized weak classifiers.

Convergence of Factorized AdaBoost.MH. The missing convergence step for Factorized AdaBoost.MH was isolated by Kégl (2014a). After fixing a vote vector, the weighted example–label problem collapses to a binary problem on the original examples, but only with total induced mass $\|(\boldsymbol{W} \odot \boldsymbol{Y})\boldsymbol{v}\|_1$. Therefore the weak learning condition yields a useful factorized edge only if some vote vector preserves enough mass for every possible current weight matrix and label matrix. Zou et al. (2024) proved that such a vote vector always exists and obtained the lower bound

$\mathfrak{W}_{n,K} \geq \max\{1/n, 1/\sqrt{2K}\}$, thereby giving a boosting-type convergence theorem for Factorized AdaBoost.MH. However, the above lower bound still leaves a possible slowdown depending on the number of samples or classes. Our contribution is to sharpen this combinatorial bottleneck: we prove that the relevant minimax quantity is bounded below by a universal constant, with the explicit constants $C_{\min\{n+1,K\}}$ and matching upper bounds described in the main theorem.

B Proof of the Lower Bounds

B.1 Proof of the Lower Bound in Theorem 3.1

In this section, we provide the proof for the lower bounds in Theorem 3.1. According to Theorem 3.3 in (Zou et al., 2024), we have $\mathfrak{W}_{n,K} \geq \frac{1}{n}$. So it remains to prove $\mathfrak{W}_{n,K} \geq C_K$.

Fix any $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, for any $i \in [n]$, let $c_i \in [K]$ be the column unique index such that $\boldsymbol{Y}_{ic_i} = +1$. We define the row mass for the i-th row as

$$s_i = \sum_{j=1}^{K} \boldsymbol{W}_{ij}.$$

Rows with $s_i = 0$ do not affect w'_Σ, so we consider the rows with $s_i > 0$. For rows with $s_i > 0$, we define the true-class weight fraction as

$$a_i = \frac{\boldsymbol{W}_{ic_i}}{s_i}.$$

It is easy to see that

$$0 \leq a_i \leq 1, \quad \sum_{i=1}^{n} s_i = 1.$$

Then the total weight for the true classes is

$$\rho = \sum_{i=1}^{n} s_i a_i = \sum_{i=1}^{n} \boldsymbol{W}_{ic_i}.$$

Since $\sum_{i=1}^{n} s_i = 1$ and $0 \leq a_i \leq 1$ for all $i \in [n]$, we know that $0 \leq \rho \leq 1$. Next, given fixed $\boldsymbol{W}, \boldsymbol{Y}$, we provide three lower bounds for $\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v})$.

Lemma 4.1. *Fix integers $n \geq 1$ and $K \geq 2$, for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq |2\rho - 1|.$$

Proof of Lemma 4.1. We choose all-one vector to show this lower bound. Take $\boldsymbol{v}^1 = (+1, \ldots, +1) \in \mathcal{V}_K$, we have that, for the i-th row,

$$\left| \sum_{j=1}^{K} \boldsymbol{W}_{ij} \boldsymbol{Y}_{ij} \boldsymbol{v}_j^1 \right| = \left| \sum_{j=1}^{K} \boldsymbol{W}_{ij} \boldsymbol{Y}_{ij} \right| = \left| \boldsymbol{W}_{ic_i} - \sum_{j \neq c_i}^{K} \boldsymbol{W}_{ij} \right| \overset{(a)}{=} |s_i a_i - s_i(1 - a_i)| = s_i |2a_i - 1|,$$

where step (a) follows from the facts that $\boldsymbol{W}_{ic_i} = s_i a_i$ and $\sum_{j \neq c_i}^{K} \boldsymbol{W}_{ij} = s_i(1 - a_i)$. Then we have:

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}^1) = \sum_{i=1}^{n} s_i |2a_i - 1| \geq \left| \sum_{i=1}^{n} s_i (2a_i - 1) \right| \overset{(a)}{=} |2\rho - 1|,$$

where step (a) comes from the definition of ρ and the fact that $\sum_{i=1}^{n} s_i = 1$. □

Lemma 4.2. *Fix integers $n \geq 1$ and $K \geq 2$, when K is **even**, then for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \rho + \frac{1 - \rho}{K - 1}.$$

Proof of Lemma 4.2. Since $K \geq 2$ is even, we can define a set of balanced vectors where the numbers of -1 and $+1$ are the same:

$$\mathcal{B}_K \coloneqq \left\{ \boldsymbol{v} \in \{-1,+1\}^K \,\middle|\, \sum_{j=1}^{K} \boldsymbol{v}_j = 0 \right\} \subseteq \mathcal{V}_K.$$

Now we assume that random vector $\mathbf{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_K)$ is uniformly distributed on $\mathcal{B}_K$, then for any distinct pair $j \neq c$, we have that:

$$\mathbb{E}_{\mathbf{v}}\left[\mathrm{v}_j | \mathrm{v}_c = +1\right] \overset{(a)}{=} \frac{\binom{K}{\frac{K}{2}-1} - \binom{K}{\frac{K}{2}}}{K-1} = -\frac{1}{K-1}, \tag{B.1}$$

where step (a) follows from the fact that, condition on $\mathrm{v}_c = +1$, the remaining $K-1$ coordinates are symmetric, and $\frac{K}{2}-1$ of them are $+1$, $\frac{K}{2}$ of them are -1. Similarly, we can prove that:

$$\mathbb{E}_{\mathbf{v}}\left[\mathrm{v}_j | \mathrm{v}_c = -1\right] = \frac{\binom{K}{\frac{K}{2}} - \binom{K}{\frac{K}{2}-1}}{K-1} = \frac{1}{K-1}. \tag{B.2}$$

For fixed $\boldsymbol{W}, \boldsymbol{Y}$, we define

$$Z_i(\boldsymbol{v}) = \sum_{j=1}^{K} \boldsymbol{W}_{ij} \boldsymbol{Y}_{ij} \boldsymbol{v}_j$$

for each row i. Then we have that

$$\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = +1] = \boldsymbol{W}_{ic_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij} \mathbb{E}[\mathrm{v}_j | \mathrm{v}_{c_i} = +1] \overset{(a)}{=} \boldsymbol{W}_{ic_i} + \frac{1}{K-1} \sum_{j \neq c_i} \boldsymbol{W}_{ij},$$

where step (a) is from Equation (B.1). Since $\boldsymbol{W}_{ic_i} = s_i a_i$ and $\sum_{j \neq c_i}^{K} \boldsymbol{W}_{ij} = s_i(1-a_i)$, we have:

$$\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = +1] = s_i \left(a_i + \frac{1-a_i}{K-1} \right).$$

Similarly, we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = -1] &= -\boldsymbol{W}_{ic_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij} \mathbb{E}[\mathrm{v}_j | \mathrm{v}_{c_i} = -1] \\
&= -\boldsymbol{W}_{ic_i} - \frac{1}{K-1} \sum_{j \neq c_i} \boldsymbol{W}_{ij} = -s_i \left(a_i + \frac{1-a_i}{K-1} \right).
\end{aligned}$$

By the distribution of $\mathbf{v}$, it is obvious that $\mathbb{P}[\mathrm{v}_{c_i} = +1] = \mathbb{P}[\mathrm{v}_{c_i} = -1] = \frac{1}{2}$. Collecting the above together, we have:

$$\begin{aligned}
\mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})|\right] &= \mathbb{P}[\mathrm{v}_{c_i} = +1] \cdot \mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})| \middle| \mathrm{v}_{c_i} = +1\right] + \mathbb{P}[\mathrm{v}_{c_i} = -1] \cdot \mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})| \middle| \mathrm{v}_{c_i} = -1\right] \\
&= \frac{1}{2}\mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})| \middle| \mathrm{v}_{c_i} = +1\right] + \frac{1}{2}\mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})| \middle| \mathrm{v}_{c_i} = -1\right] \\
&\geq \frac{1}{2}\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = +1] + \frac{1}{2}\mathbb{E}_{\mathbf{v}}[-Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = -1] \\
&= \frac{1}{2} s_i \left(a_i + \frac{1-a_i}{K-1} \right) + \frac{1}{2} s_i \left(a_i + \frac{1-a_i}{K-1} \right) = s_i \left(a_i + \frac{1-a_i}{K-1} \right).
\end{aligned}$$

Then we get that

$$\mathbb{E}_{\mathbf{v}}[w'_{\Sigma}(\boldsymbol{W}, \boldsymbol{Y}, \mathbf{v})] = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})|\right] \geq \sum_{i=1}^{n} s_i \left(a_i + \frac{1-a_i}{K-1} \right) = \rho + \frac{1-\rho}{K-1}.$$

So

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_{\Sigma}(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \mathbb{E}_{\mathbf{v}}[w'_{\Sigma}(\boldsymbol{W}, \boldsymbol{Y}, \mathbf{v})] \geq \rho + \frac{1-\rho}{K-1}.$$

□

Lemma 4.3. *Fix integers $n \geq 1$ and $K \geq 3$, when K is **odd**, then for all $\boldsymbol{W} \in \mathcal{W}_{n,K}$ and all $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have that:*

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_{\Sigma}(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq \rho + \frac{1-\rho}{K}.$$

Proof of Lemma 4.3. When $K \geq 3$ and K is odd, there is no balanced vectors, we then define a set of nearly balanced vectors as:

$$\mathcal{B}'_K \coloneqq \left\{ \boldsymbol{v} \in \{-1,+1\}^K \,\middle|\, \left| \sum_{j=1}^{n} \boldsymbol{v}_j \right| = 1 \right\}.$$

Now we assume that random vector $\mathbf{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_K)$ is uniformly distributed on $\mathcal{B}'_K$. Since the elements of $\mathbf{v}$ are symmetric, and it is easy to see that $-\mathbf{v}, \mathbf{v}$ have the same distribution, we can get that:

$$\mathbb{P}(\mathrm{v}_j = +1) = \mathbb{P}(\mathrm{v}_j = -1) = \frac{1}{2} \quad \forall j \in [K].$$

Since $\mathbf{v} \in \mathcal{B}'_K$, we have $\left(\sum_{j=1}^K \mathrm{v}_j\right)^2 = 1$, so $\mathbb{E}_{\mathbf{v}}\left[\left(\sum_{j=1}^K \mathrm{v}_j\right)^2\right] = 1$. Expanding the square, we have

$$\mathbb{E}_{\mathbf{v}}\left[\left(\sum_{j=1}^{K} \mathrm{v}_j\right)^2\right] = \sum_{j=1}^{K} \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j^2] + 2 \sum_{1 \leq j < k \leq K} \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k]$$

By the symmetry of all the coordinates of $\mathbf{v}$, there exists a constant $r \in [-1, 1]$ such that for all $j \neq k$, we have $\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k] = r$. So we have that:

$$1 = \mathbb{E}_{\mathbf{v}}\left[\left(\sum_{j=1}^{K} \mathrm{v}_j\right)^2\right] = \sum_{j=1}^{K} \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j^2] + K(K-1)r \overset{(a)}{=} K + K(K-1)r,$$

where (a) is because $\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j^2] = 1$ for all $j \in [K]$. Solving the above equation, we can get that $\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k] = r = -\frac{1}{K}$ for all $j \neq k$. Then, for $j \neq k$, by the fact that

$$\begin{aligned}
\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k] &= \mathbb{P}(\mathrm{v}_k = +1) \cdot \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k | \mathrm{v}_k = +1] + \mathbb{P}(\mathrm{v}_k = -1) \cdot \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j \mathrm{v}_k | \mathrm{v}_k = -1] \\
&= \mathbb{P}(\mathrm{v}_k = +1) \cdot \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = +1] - \mathbb{P}(\mathrm{v}_k = -1) \cdot \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = -1] \\
&= \frac{1}{2}\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = +1] - \frac{1}{2}\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = -1] \\
&= \frac{1}{2}\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = +1] + \frac{1}{2}\mathbb{E}_{\mathbf{v}}[-\mathrm{v}_j | -\mathrm{v}_k = +1] \\
&\overset{(a)}{=} \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = +1] \overset{(b)}{=} -\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = -1],
\end{aligned}$$

where steps (a) and (b) follow from the fact that $\mathbf{v}$ and $-\mathbf{v}$ have the same distribution. So we have that

$$\mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = +1] = -\frac{1}{K}, \quad \mathbb{E}_{\mathbf{v}}[\mathrm{v}_j | \mathrm{v}_k = -1] = \frac{1}{K}. \tag{B.3}$$

Then by similar arguments shown in the proof of Lemma 4.2, we have

$$\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = +1] = \boldsymbol{W}_{ic_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij} \mathbb{E}[\mathrm{v}_j | \mathrm{v}_{c_i} = +1] = \boldsymbol{W}_{ic_i} + \frac{1}{K} \sum_{j \neq c_i} \boldsymbol{W}_{ij} = s_i \left(a_i + \frac{1-a_i}{K}\right),$$

$$\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = -1] = -\boldsymbol{W}_{ic_i} - \sum_{j \neq c_i} \boldsymbol{W}_{ij} \mathbb{E}[\mathrm{v}_j | \mathrm{v}_{c_i} = -1] = -\boldsymbol{W}_{ic_i} - \frac{1}{K} \sum_{j \neq c_i} \boldsymbol{W}_{ij} = -s_i \left(a_i + \frac{1-a_i}{K}\right).$$

By similar arguments in the proof of Lemma 4.2 again, we can obtain that

$$\begin{aligned}
\mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})|\right] &\geq \frac{1}{2}\mathbb{E}_{\mathbf{v}}[Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = +1] + \frac{1}{2}\mathbb{E}_{\mathbf{v}}[-Z_i(\mathbf{v}) | \mathrm{v}_{c_i} = -1] \\
&= \frac{1}{2} s_i \left(a_i + \frac{1-a_i}{K}\right) + \frac{1}{2} s_i \left(a_i + \frac{1-a_i}{K}\right) = s_i \left(a_i + \frac{1-a_i}{K}\right).
\end{aligned}$$

Then

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \mathbb{E}_{\mathbf{v}}[w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\mathbf{v})] = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{v}}\left[|Z_i(\mathbf{v})|\right] \geq \sum_{i=1}^{n} s_i \left(a_i + \frac{1-a_i}{K}\right) = \rho + \frac{1-\rho}{K}.$$

□

B.1.1 Lower Bound for Even K

Putting Lemmas 4.1 and 4.2 together, we know that when K is even and $K \geq 2$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K-1}\right\}, \tag{B.4}$$

where $\rho \in [0,1]$ and depends on $\boldsymbol{W},\boldsymbol{Y}$. Minimize the right side of Equation (B.4) over $\rho \in [0,1]$, then for any $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \min_{\rho\in[0,1]} \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K-1}\right\}.$$

If $2\rho - 1 \leq 0$, i.e., $0 \leq \rho \leq \frac{1}{2}$, then $|2\rho-1| = 1-2\rho$, which decreases with ρ. It is easy to see that $\rho + \frac{1-\rho}{K-1} = \frac{K-2}{K-1}\rho + \frac{1}{K-1}$, which increases with ρ. So when $1-2\rho = \rho + \frac{1-\rho}{K-1}$, we achieve $\min_{\rho\in\left[0,\frac{1}{2}\right]} \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K-1}\right\}$. Solving the equation $1-2\rho = \rho + \frac{1-\rho}{K-1}$ yields

$$\rho^*_{\text{even}} = \frac{K-2}{3K-4} < \frac{1}{3}.$$

The corresponding minimal value is

$$1 - 2\rho^*_{\text{even}} = \frac{K}{3K-4}.$$

If $\rho \geq \frac{1}{2}$, we have:

$$\begin{aligned}
&\min_{\rho\in\left[\frac{1}{2},1\right]} \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K-1}\right\} \geq \min_{\rho\in\left[\frac{1}{2},1\right]} \left(\rho + \frac{1-\rho}{K-1}\right) \\
&\overset{(a)}{=} \rho + \frac{1-\rho}{K-1}\bigg|_{\rho=\frac{1}{2}} = \frac{K}{2K-2} \overset{(b)}{\geq} \frac{K}{3K-4},
\end{aligned}$$

where step (a) is from the fact that $\rho + \frac{1-\rho}{K-1}$ increases with ρ; and step (b) is because $K \geq 2$. Putting all the above together, we get that when $K \geq 2$ and is even, for any $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \frac{K}{3K-4}.$$

Minimizing over $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have

$$\mathfrak{W}_{n,K} \geq \frac{K}{3K-4} = C_K.$$

B.1.2 Lower Bound for Odd K

In this section, we take similar steps as Section B.1.1 to prove the result. Putting Lemmas 4.1 and 4.3 together, we know that when K is odd and $K \geq 3$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K}\right\}, \tag{B.5}$$

where $\rho \in [0,1]$ and depends on $\boldsymbol{W},\boldsymbol{Y}$. Minimize the right side of Equation (B.5) over $\rho \in [0,1]$, then for any $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W},\boldsymbol{Y},\boldsymbol{v}) \geq \min_{\rho\in[0,1]} \max\left\{|2\rho-1|, \rho + \frac{1-\rho}{K}\right\}.$$

When $0 \le \rho \le \frac{1}{2}$, $|2\rho - 1| = 1 - 2\rho$ decreases with ρ, and $\rho + \frac{1-\rho}{K} = \frac{K-1}{K}\rho + \frac{1}{K}$ increases as ρ. So when $1 - 2\rho = \rho + \frac{1-\rho}{K}$, we achieve $\min_{\rho \in \left[0, \frac{1}{2}\right]} \max\left\{|2\rho - 1|, \rho + \frac{1-\rho}{K}\right\}$. Solving the equation $1 - 2\rho = \rho + \frac{1-\rho}{K}$ yields

$$\rho^*_{\text{odd}} = \frac{K-1}{3K-1} < \frac{1}{3}.$$

The corresponding minimal value is

$$1 - 2\rho^*_{\text{odd}} = \frac{K+1}{3K-1}.$$

When $\rho \ge \frac{1}{2}$,

$$\begin{aligned}
&\min_{\rho \in \left[\frac{1}{2}, 1\right]} \max\left\{|2\rho - 1|, \rho + \frac{1-\rho}{K}\right\} \ge \min_{\rho \in \left[\frac{1}{2}, 1\right]} \left(\rho + \frac{1-\rho}{K}\right) \\
&\overset{(a)}{=} \rho + \frac{1-\rho}{K}\bigg|_{\rho = \frac{1}{2}} = \frac{K+1}{2K} \overset{(b)}{\ge} \frac{K+1}{3K-1},
\end{aligned}$$

where step (a) is from the fact that $\rho + \frac{1-\rho}{K}$ increases with ρ; and step (b) is because $K \ge 3$. Putting all the above together, we get that when $K \ge 3$ and is odd, for any $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$,

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \ge \frac{K+1}{3K-1}.$$

Minimizing over $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we have

$$\mathfrak{W}_{n,K} \ge \frac{K+1}{3K-1} = C_K.$$

B.2 Proof of the Refined Lower Bound

In this section, we prove the refined lower bound, i.e., Theorem 3.3. The proof of Theorem 3.3 is based on the lower bound in Theorem 3.1.

For given $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we define c_i as the true label for the i-th example, i.e., $\boldsymbol{Y}_{i,c_i} = 1$ for all $i \in [n]$. We define

$$I_+ \coloneqq \left\{ i \in [n] : \sum_{j=1}^{K} \boldsymbol{W}_{ij} > 0 \right\}$$

as the set of indices whose row-weight summation is non-zero. We only need to consider the rows with indices in I_+ because the behavior of the rows outside of I_+ does not affect the value of $\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v})$. Define the set of all labels as

$$S \coloneqq \{c_i : i \in I_+\}.$$

Let $r = |S| \le K$, by the above definitions, it is easy to get that $r \le |I_+| \le n$. If $r = K$, then $K = r \le n < n + 1$, then $\min\{n+1, K\} = K$. Then by the lower bound in Theorem 3.1, we know that

$$\mathfrak{W}_{n,K} \ge C_K = C_{\min\{n+1,K\}}.$$

Then we only need to consider the case that $r < K$. When $r < K$, let $q = r + 1$. We enumerate the elements of S and define

$$S = \{s_1, \ldots, s_r\}, \qquad U \coloneqq [K] \backslash S \neq \emptyset,$$

where $U \neq \emptyset$ because $r = |S| < K$ by our assumption.

For the given $\boldsymbol{W}, \boldsymbol{Y}$, we then construct $\bar{\boldsymbol{W}} \in \mathcal{W}_{n,q}$ and $\bar{\boldsymbol{Y}} \in \mathcal{Y}_{n,q}$ so that the following inequation holds:

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \ge \max_{\boldsymbol{v} \in \mathcal{V}_q} w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{v}).$$

For all $i \in [n]$, define

$$\bar{\boldsymbol{W}}_{ij} \coloneqq \boldsymbol{W}_{is_j} \text{ for } 1 \le j \le r, \quad \bar{\boldsymbol{W}}_{iq} \coloneqq \sum_{j \in U} \boldsymbol{W}_{ij},$$

i.e., we put the weights of the rows inside U in the q-th row of $\bar{\boldsymbol{W}}$. It is obvious that $\bar{\boldsymbol{W}}_{ij} \ge 0$ for all $i \in [n], j \in [q]$. Moreover,

$$\sum_{i=1}^{n}\sum_{j=1}^{q} \bar{\boldsymbol{W}}_{ij} = \sum_{i=1}^{n}\left(\sum_{j=1}^{r} \boldsymbol{W}_{is_j} + \sum_{j \in U} \boldsymbol{W}_{ij}\right) = \sum_{i=1}^{n}\sum_{j=1}^{K} \boldsymbol{W}_{ij} = 1.$$

So we verified that $\bar{\boldsymbol{W}} \in \mathcal{W}_{n,q}$. Then we construct $\bar{\boldsymbol{Y}}$.

For any $i \in I_+$, we have $c_i \in S$, then by the definition of S, there exists an unique index $l(i) \in [r]$ such that $c_i = s_{l(i)}$. Then for all $i \in I_+$, we define

$$\bar{\boldsymbol{Y}}_{i,l(i)} = +1, \quad \bar{\boldsymbol{Y}}_{ij} = -1 \text{ for } j \in [r+1] \backslash \{l(i)\}.$$

For $i \notin I_+$, the behavior of the i-th row does not affect the result, we can set $\bar{\boldsymbol{Y}}_i$ to be any q-dimensional signed one-hot vector. It is easy to see that $\bar{\boldsymbol{Y}} \in \mathcal{Y}_{n,q}$.

Fix the vector $\boldsymbol{u} = (u_1, \ldots, u_q) \in \mathcal{V}_q$, we construct a K-dimensional vector $\boldsymbol{v} \in \mathcal{V}_K$ as follows:

$$\boldsymbol{v}_{s_j}(\boldsymbol{u}) = u_j \text{ for } 1 \le j \le r, \quad \boldsymbol{v}_j(\boldsymbol{u}) = u_q \text{ for } j \in U.$$

For any $i \in I_+$, since the true label is c_i, then $\boldsymbol{Y}_{ic_i} = +1$ and $\boldsymbol{Y}_{ij} = -1$ for $j \in U$. Then we have:

$$\begin{aligned}
\sum_{j=1}^{K} \boldsymbol{W}_{ij}\boldsymbol{Y}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) &= \boldsymbol{W}_{ic_i}\boldsymbol{v}_{c_i}(\boldsymbol{u}) - \sum_{1 \le j \le K, j \ne c_i} \boldsymbol{W}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) \\
&\overset{(a)}{=} \boldsymbol{W}_{ic_i}\boldsymbol{u}_{l(i)} - \sum_{1 \le j \le K, j \ne c_i} \boldsymbol{W}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) \\
&= \boldsymbol{W}_{ic_i}\boldsymbol{u}_{l(i)} - \sum_{j \in S \backslash \{c_i\}} \boldsymbol{W}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) - \sum_{j \in U} \boldsymbol{W}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) \\
&\overset{(b)}{=} \boldsymbol{W}_{ic_i}\boldsymbol{u}_{l(i)} - \sum_{1 \le l \le r, l \ne l(i)} \boldsymbol{W}_{is_l}\boldsymbol{v}_{s_l}(\boldsymbol{u}) - \sum_{j \in U} \boldsymbol{W}_{ij}\boldsymbol{v}_j(\boldsymbol{u}) \\
&\overset{(c)}{=} \boldsymbol{W}_{is_{l(i)}}\boldsymbol{u}_{l(i)} - \sum_{1 \le l \le r, l \ne l(i)} \boldsymbol{W}_{is_l}\boldsymbol{u}_l - \sum_{j \in U} \boldsymbol{W}_{ij}\boldsymbol{u}_q \\
&\overset{(d)}{=} \bar{\boldsymbol{W}}_{il(i)}\bar{\boldsymbol{Y}}_{il(i)}\boldsymbol{u}_{l(i)} + \sum_{1 \le l \le r, l \ne l(i)} \bar{\boldsymbol{W}}_{il}\bar{\boldsymbol{Y}}_{il}\boldsymbol{u}_l + \bar{\boldsymbol{W}}_{iq}\bar{\boldsymbol{Y}}_{iq}\boldsymbol{u}_q \\
&= \sum_{l=1}^{q} \bar{\boldsymbol{W}}_{il}\bar{\boldsymbol{Y}}_{il}\boldsymbol{u}_l,
\end{aligned}$$

where step (a) is from the definition of $\boldsymbol{v}(\boldsymbol{u})$ and the fact that $c_i = s_{l(i)}$; (b) changes the indices of the sum over $S \backslash \{c_i\}$; (c) is from the definition of $\boldsymbol{v}_{s_j}(\boldsymbol{u})$ and the fact that $c_i = s_{l(i)}$; and (d) is from the definition of $\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}$. If $i \notin I_+$, both sides of the above equation equals to zero. So for any $i \in [n]$, we have:

$$\left|\sum_{j=1}^{K} \boldsymbol{W}_{ij}\boldsymbol{Y}_{ij}\boldsymbol{v}_j(\boldsymbol{u})\right| = \left|\sum_{l=1}^{q} \bar{\boldsymbol{W}}_{il}\bar{\boldsymbol{Y}}_{il}\boldsymbol{u}_l\right|.$$

Sum over $i \in [n]$ yields

$$w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}(\boldsymbol{u})) = w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{u}).$$

Assume $\boldsymbol{u}^* = \arg\max_{\boldsymbol{u} \in \mathcal{V}_q} w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{u})$, then we have:

$$\max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \ge w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}(\boldsymbol{u}^*)) = w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{u}^*) = \max_{\boldsymbol{u} \in \mathcal{V}_q} w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{u}).$$

By the proof of the lower bounds of Theorem 3.1, we have $\max_{\boldsymbol{u}\in\mathcal{V}_q} w'_\Sigma(\bar{\boldsymbol{W}}, \bar{\boldsymbol{Y}}, \boldsymbol{u}) \geq C_q$. So we can conclude that when $r < K$,

$$\max_{\boldsymbol{v}\in\mathcal{V}_K} w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) \geq C_q \overset{(a)}{\geq} C_{\min\{n+1,K\}},$$

where step (a) is from the fact that $q \leq \min\{n+1, K\}$(because $q = r+1 \leq n+1$ and $q = r+1 < K+1$) and the monotonicity property in Proposition 3.2. Minimize over $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$, we get

$$\mathfrak{W}_{n,K} \geq C_{\min\{n+1,K\}}.$$

Putting the cases $r < K$ and $r \geq K$ together, we conclude that for any integers $n \geq 1$ and $K \geq 2$,

$$\mathfrak{W}_{n,K} \geq C_{\min\{n+1,K\}}.$$

C Proof of the Upper Bounds

C.1 Proof of the Upper Bound in Theorem 3.1

Let $M \coloneqq \min\{n, K\}$, we choose any $\boldsymbol{Y}' \in \mathcal{Y}_{n,K}$ such that:

$$\boldsymbol{Y}'_{ii} = +1 \qquad \forall i \in [M],$$

i.e., we set the true label of the i-th example as i for $i \in [M]$. For any fixed parameter $a \in [0, 1]$, we construct the weight matrix $\boldsymbol{W}^a$ as follows:

1. We set $\boldsymbol{W}^a_{ii} \coloneqq \frac{a}{M}$ for all $i \in [M]$.
2. For $i \in [M]$ and $j \in [M]\backslash\{i\}$, we set $\boldsymbol{W}^a_{ij} \coloneqq \frac{\frac{1}{M} - \boldsymbol{W}^a_{ii}}{M-1} = \frac{1-a}{M(M-1)}$.
3. When $i > M$ or $j > M$, we set $\boldsymbol{W}^a_{ij} = 0$.

From the above construction, it is easy to verify that $\boldsymbol{W}^a \in \mathcal{W}_{n,K}$ and $s_i = \sum_{j=1}^M \boldsymbol{W}^a_{ij} = \frac{1}{M}$. When $M = 1$, it is easy to see that $\boldsymbol{W}^a_{11} = a$ and $w'_\Sigma(\boldsymbol{W}^a, \boldsymbol{Y}', \boldsymbol{v}) = 1$ for any $\boldsymbol{v} \in \mathcal{V}_K$, which directly leads to the upper bound $\mathfrak{W}_{n,K} \leq C_1$. Without loss of generality, we assume that $M \geq 2$ in the remaining arguments.

Next, we calculate $w'_\Sigma(\boldsymbol{W}^a, \boldsymbol{Y}', \boldsymbol{v})$ for a given $\boldsymbol{v}$. Fix $\boldsymbol{v} \in \mathcal{V}_K$, since $\boldsymbol{v}_{M+1}, \ldots, \boldsymbol{v}_K$ do not effect the value of $w'_\Sigma(\boldsymbol{W}^a, \boldsymbol{Y}', \boldsymbol{v})$, define

$$V \coloneqq \sum_{j=1}^M \boldsymbol{v}_j, \quad b \coloneqq \frac{1-a}{M-1}.$$

Then

$$\begin{aligned} w'_\Sigma(\boldsymbol{W}^a, \boldsymbol{Y}', \boldsymbol{v}) &= \sum_{i=1}^n \left| \sum_{j=1}^K \boldsymbol{W}^a_{ij} \cdot \boldsymbol{Y}' \cdot \boldsymbol{v}_j \right| \overset{(a)}{=} \sum_{i=1}^M \left| \boldsymbol{W}^a_{ii} \boldsymbol{v}_i - \sum_{j=1, j\neq i}^M \boldsymbol{W}^a_{ij} \boldsymbol{v}_j \right| \\ &\overset{(b)}{=} \frac{1}{M} \sum_{i=1}^M \left| a\boldsymbol{v}_i - \sum_{j=1, j\neq i}^M b\boldsymbol{v}_j \right| \overset{(c)}{=} \frac{1}{M} \sum_{i=1}^M |a\boldsymbol{v}_i - b(V - \boldsymbol{v}_i)| \\ &= \frac{1}{M} \sum_{i=1}^M |(a+b)\boldsymbol{v}_i - bV|, \end{aligned}$$

where steps (a) and (b) are from the definition of $\boldsymbol{W}^a, \boldsymbol{Y}'$; step (c) is from the fact that $\sum_{j=1, j\neq i}^M \boldsymbol{v}_j = V - \boldsymbol{v}_i$. Assume that among $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_M$, there are p of them have value $+1$, and $M - p$ of them have value -1. Then, solving the equation $p - (M - p) = V$ gives

$$p = \frac{M + V}{2}, \qquad M - p = \frac{M - V}{2}.$$

Then,

$$w'_\Sigma(\boldsymbol{W}^a, \boldsymbol{Y}', \boldsymbol{v}) = \frac{1}{M}\left[p|(a+b) - bV| + (M-p)|(a+b) + bV|\right] \coloneqq f(V),$$

where the possible value set of V is

$$D_V \coloneqq \{-M, -M+2, \ldots, M-2, M\}.$$

Fix a, b, M, the next step is to figure out the maximal value of $f(V)$ over D_V. By the definition of $f(V)$, it is easy to see that $f(V) = f(-V)$ for any $V \in D_V$, so it suffices to consider $0 \le V \le M$. Since $a, b, V \ge 0$, we have $f(V) = \frac{1}{M}\left[p|(a+b) - bV| + (M-p)\left((a+b) + bV\right)\right]$.

When $(a+b) - bV \ge 0$, i.e., $0 \le V \le \frac{a+b}{b}$, we have

$$f(V) = \frac{1}{M}\left[p\left((a+b) - bV\right) + (M-p)\left((a+b) + bV\right)\right] = (a+b) + \frac{bV}{M}(M-2p) \overset{(a)}{=} (a+b) - \frac{bV^2}{M},$$

where step (a) is because $p = \frac{M+V}{2}$.

When $(a+b) - bV \le 0$, i.e., $\frac{a+b}{b} \le V \le M$, we have

$$f(V) = \frac{1}{M}\left[p\left(bV - (a+b)\right) + (M-p)\left((a+b) + bV\right)\right] = bV + \frac{a+b}{M}(M-2p) = V\left(b - \frac{a+b}{M}\right).$$

Collecting the above, we get

$$f(V) = \begin{cases} (a+b) - \frac{bV^2}{M} & \text{if } 0 \le V \le \frac{a+b}{b} \\ V\left(b - \frac{a+b}{M}\right) & \text{if } \frac{a+b}{b} \le V \le M. \end{cases} \tag{C.1}$$

C.1.1 Upper Bound for Even M

When $M \ge 2$ and is even, take $a = a^e \coloneqq \frac{M-2}{3M-4}$, then we have

$$b = \frac{1-a}{M-1} = \frac{2}{3M-4}, \qquad a+b = \frac{M}{3M-4}, \qquad \frac{a+b}{b} = \frac{M}{2}.$$

According to Equation (C.1), when $0 \le V \le \frac{M}{2}$,

$$f(V) = (a+b) - \frac{bV^2}{M} \le a+b = \frac{M}{3M-4};$$

when $\frac{M}{2} \le V \le M$,

$$b - \frac{a+b}{M} = \frac{1}{3M-4} > 0,$$

so we have

$$f(V) \le f(M) = M \cdot \frac{1}{3M-4} = \frac{M}{3M-4}.$$

In conclusion, $f(V) \le \frac{M}{3M-4}$ on D_V, which means that

$$\mathfrak{W}_{n,K} \le \max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}^{a^e}, \boldsymbol{Y}', \boldsymbol{v}) \le \frac{M}{3M-4}.$$

C.1.2 Upper Bound for Odd M

When $M \ge 3$ and is odd, take $a = a^o \coloneqq \frac{M-1}{3M-1}$, then we have

$$b = \frac{1-a}{M-1} = \frac{2M}{(M-1)(3M-1)}, \qquad a+b = \frac{M^2+1}{(M-1)(3M-1)}, \qquad \frac{a+b}{b} = \frac{M^2+1}{2M}.$$

Since M is odd, V can not have even value, then we consider $V \in \{1, 3, 5, \ldots, M\}$. According to Equation (C.1), when $1 \le V \le \frac{a+b}{b}$, $f(V)$ decreases with V, so we have $f(V) \le f(1) = \frac{M+1}{3M-1}$.

When $\frac{a+b}{b} \le V \le M$,

$$b - \frac{a+b}{M} = \frac{M+1}{M(3M-1)} > 0,$$

so we have

$$f(V) \le f(M) = M \cdot \frac{M+1}{M(3M-1)} = \frac{M+1}{3M-1}.$$

In conclusion, $f(V) \le \frac{M+1}{3M-1}$ on D_V, which means that

$$\mathfrak{W}_{n,K} \le \max_{\boldsymbol{v} \in \mathcal{V}_K} w'_\Sigma(\boldsymbol{W}^{a^o}, \boldsymbol{Y}', \boldsymbol{v}) \le \frac{M+1}{3M-1}.$$

Putting the results in Sections C.1.1 and C.1.2 together proves the upper bound.

C.2 Proof of the Upper Bound in Theorem 3.4

If n is even, then $n+1$ is odd and $C_{n+1} = \frac{(n+1)+1}{3(n+1)-1} = \frac{n+2}{3n+2}$. For simplicity, define $D = 3n+2$, then we construct $\boldsymbol{W} \in \mathcal{W}_{n,K}, \boldsymbol{Y} \in \mathcal{Y}_{n,K}$.

Since $n+1 \le K$, we define

$$\boldsymbol{Y}_{ii} = +1 \text{ for } i \in [n], \quad \boldsymbol{Y}_{ij} = -1 \text{ for all } j \ne i.$$

It is easy to verify that $\boldsymbol{Y} \in \mathcal{Y}_{n,K}$.

For $\boldsymbol{W}$, we set the weights for the j-th column to be 0 for all $j \ge n+2$ and define

$$\begin{aligned}
\boldsymbol{W}_{ii} &= \frac{1}{D} \quad \text{for } \ i \in [n], \\
\boldsymbol{W}_{ij} &= \frac{2}{nD} \quad \text{for } \ 1 \le j \le n \text{ and } \ j \ne i, \\
\boldsymbol{W}_{i,n+1} &= \frac{4}{nD} \quad \text{for } \ i \in [n], \\
\boldsymbol{W}_{ij} &= 0 \quad \text{for } \ j \ge n+2.
\end{aligned}$$

It is obvious that $\boldsymbol{W}_{ij} \ge 0$ for $i \in [n], j \in [K]$. For any $i \in [n]$, we have:

$$\sum_{j=1}^{K} \boldsymbol{W}_{ij} = \frac{1}{D} + (n-1)\frac{2}{nD} + \frac{4}{nD} = \frac{3n+2}{nD} = \frac{1}{n}.$$

So the weights for each row are equal and $\sum_{i=1}^{n}\sum_{j=1}^{K} \boldsymbol{W}_{ij} = 1$, so $\boldsymbol{W} \in \mathcal{W}_{n,K}$.

For any $\boldsymbol{v} \in \mathcal{V}_K$, let

$$V \coloneqq \sum_{j=1}^{n} v_j, \quad u = v_{n+1},$$

then we have:

$$\begin{aligned}
Z_i(\boldsymbol{v}) &\coloneqq \sum_{j=1}^{K} \boldsymbol{W}_{ij}\boldsymbol{Y}_{ij}v_j = \sum_{j=1}^{n+1} \boldsymbol{W}_{ij}\boldsymbol{Y}_{ij}v_j \\
&\overset{(a)}{=} \frac{1}{D}v_i - \frac{2}{nD}\sum_{1 \le j \le n, j \ne i} v_j - \frac{4}{nD}u \\
&\overset{(b)}{=} \frac{1}{D}v_i - \frac{2}{nD}(V - v_i) - \frac{4}{nD}u \\
&= \frac{(n+1)v_i - 2V - 4u}{nD},
\end{aligned}$$

where (a) is from the definitions of $\boldsymbol{W}$ and $\boldsymbol{Y}$; and (b) is from the definition of V. So

$$w'_\Sigma(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) = \sum_{i=1}^{n} |Z_i(\boldsymbol{v})| = \frac{1}{nD}\sum_{i=1}^{n} \left|(n+2)v_i - 2V - 4u\right|.$$

For simplicity, let $A = n + 2$ and $B = 2V + 4u$, let p be the number of $+1$ in $v_1, \ldots, v_n$, then by the definition of V, we have $p - (n - p) = V$, which means that

$$p = \frac{n + V}{2}.$$

Define the function

$$f(B) \coloneqq \sum_{i=1}^{n} |Av_i - B| = p|A - B| + (n - p)|A + B|,$$

where the last equation is from the definition of p. As functions of $\boldsymbol{v}$, we can easily show that $V(-\boldsymbol{v}) = -V(\boldsymbol{v})$, $B(-\boldsymbol{v}) = -B(\boldsymbol{v})$, and that $f(B(-\boldsymbol{v})) = f(B(\boldsymbol{v}))$. To get the maximal value of $f(B(\boldsymbol{v}))$, we can assume with not loss of generality that $B \geq 0$. Since $A = n + 2$, then we have $A + B > 0$, so

$$f(B) = p|A - B| + (n - p)(A + B), \quad B \geq 0.$$

If $0 \leq B \leq A$, then $|A - B| = (A - B)$, then

$$f(B) = p(A - B) + (n - p)(A + B) = nA + (n - 2p)B.$$

Since $2p = n + V$, then $n - 2p = -V$, then we have:

$$f(B) = nA - VB.$$

Since n is even, then by the definition of V, it is obvious that V is even. By the definition of B and the assumption that $B \geq 0$, we have $B = 2(V + 2u) \geq 0$, i.e., $V + 2u \geq 0$. We consider the value of u.

- If $u = +1$, then we have $V + 2 \geq 0$, so $V \geq -2$. Since V is even, then either $V = -2$ or $V \geq 0$. When $V = -2$, then $B = 0$, so $BV = 0$. When $V \geq 0$, since $B \geq 0$, we have $BV \geq 0$. So we have $BV \geq 0$.
- If $u = -1$, then we have $V - 2 \geq 0$, i.e., $V \geq 2$. Since $B \geq 0$, we have $BV \geq 0$.

So $BV \geq 0$ regardless of the value of u. Then we have:

$$f(B) = nA - VB \leq nA = n(n + 2).$$

If $B \geq A$, then

$$f(B) = p(B - A) + (n - p)(A + B) = nB + (n - 2p)A = nB - VA.$$

Since $B = 2V + 4u$ and $A = n + 2$, we can obtain that:

$$f(B) = n(2V + 4u) - (n + 2)V = (n - 2)V + 4nu.$$

Since $u \leq 1$ and $V \leq n$, we have:

$$f(B) = (n - 2)V + 4nu \leq n(n - 2) + 4n = n(n + 2).$$

So we have $f(B) \leq n(n + 2)$ for $B \geq 0$. Then

$$w'_{\Sigma}(\boldsymbol{W}, \boldsymbol{Y}, \boldsymbol{v}) = \frac{f(B(\boldsymbol{v}))}{nD} \leq \frac{n(n + 2)}{nD} = \frac{n + 2}{3n + 2} = C_{n+1}.$$

Minimize over $\boldsymbol{W} \in \mathcal{W}, \boldsymbol{Y} \in \mathcal{Y}$, we have:

$$\mathfrak{W}_{n,K} \leq C_{n+1}.$$

D OTHER MISSING PROOFS

Proof of Proposition 3.2. It is easy to verify that $C_1 = C_2 = 1$, so we consider $q \geq 2$.

If q is even, then

$$\begin{aligned} C_{q+1} - C_q &= \frac{(q+1)+1}{3(q+1)-1} - \frac{q}{3q-4} = \frac{q+2}{3q+2} - \frac{q}{3q-4} \\ &= \frac{(q+2)(3q-4) - q(3q+2)}{(3q+2)(3q-4)} = \frac{-8}{(3q+2)(3q-4)} < 0. \end{aligned}$$

If q is odd, then

$$C_{q+1} - C_q = \frac{(q+1)}{3(q+1)-4} - \frac{q+1}{3q-1} = \frac{q+1}{3q-1} - \frac{q+1}{3q-1} = 0.$$

So $C_{q+1} - C_q \leq 0$ for all $q \in \mathbb{N}_+$, which means that C_q is non-increasing with respect to q.

It is easy to see that $\lim_{q\to\infty} \frac{q}{3q-4} = \frac{1}{3}$ and $\lim_{q\to\infty} \frac{q+1}{3q-1} = \frac{1}{3}$, so $\lim_{q\to\infty} C_q = \frac{1}{3}$. □

Proof of Theorem 3.5. Define $M = \min\{n+1, K\}$, according to Theorem 3.3, we know that $\mathfrak{W}_{n,K} \geq C_M$. For the upper bound, we consider there cases.

- When $K \leq n$, then $M = \min\{n+1, K\} = K = \min\{n, K\}$. Then by the upper bound in Theorem 3.1, we know that $\mathfrak{W}_{n,K} \leq C_{\min\{n,K\}} = C_M$.
- When $K \geq n+1$ and n is odd, $M = \min\{n+1, K\} = n+1$ and $\min\{n, K\} = n$. Then by the upper bound in Theorem 3.1, we have $\mathfrak{W}_{n,K} \leq C_{\min\{n,K\}} = C_n$. By the proof of Proposition 3.2, when n is odd, $C_{n+1} = C_n$. So $\mathfrak{W}_{n,K} \leq C_n = C_{n+1} = C_M$.
- When $K \geq n+1$ and n is even, $M = \min\{n+1, K\} = n+1$, by Theorem 3.4, $\mathfrak{W}_{n,K} \leq C_{n+1} = C_M$.

In conclusion, for all integers $n \geq 1$ and $K \geq 2$, we have $\mathfrak{W}_{n,K} = C_{\min\{n+1,K\}}$. □