Iterative Channel Estimation for Discrete Denoising under Channel Uncertainty

Hongjoon Ahn[1] **and Taesup Moon**[1,2]
[1] Department of Artificial Intelligence, [2]Department of Electrical and Computer Engineering,
Sungkyunkwan University, Suwon, Korea 16419
{hong0805,tsmoon}@skku.edu

Abstract

We propose a novel iterative channel estimation (ICE) algorithm that essentially removes the critical known noisy channel assumption for universal discrete denoising problem. Our algorithm is based on Neural DUDE (N-DUDE), a recently proposed neural network-based discrete denoiser, and it estimates the channel transition matrix as well as the neural network parameters in an alternating manner until convergence. While we do not make any probabilistic assumption on the underlying clean data, our ICE resembles Expectation-Maximization (EM) with variational approximation, and it takes advantage of the property that N-DUDE can always induce a marginal posterior distribution of the clean data. We carefully validate the channel estimation quality of ICE, and with extensive experiments on several radically different types of data, we show the ICE equipped neural network-based denoisers can perform *universally* well regardless of the uncertainties in both the channel and the clean source. Moreover, we show ICE becomes extremely robust to its hyperparameters, and show the denoisers with ICE significantly outperform the strong baseline that can handle the channel uncertainties for denoising, the widely used Baum-Welch algorithm for hidden Markov models (HMM).

1 Introduction

Denoising, which focuses on estimating the clean source data based on its noisy observation, is one of the most studied topics in machine learning and signal processing. In particular, *discrete denoising* focuses on denoising the data that take finite-alphabet values. Such setting covers several applications in various domains, *e.g.*, image denoising [16, 15], DNA sequence denoising [11, 12], and channel decoding [17], etc. Recently, utilizing quantized measurements from low-power sensors [18], DNA sequencing devices [7] and biometric recognition systems based on quantized images [13] are getting more prevalent, thus, denoising such data is gaining more attention.

The Bayesian and supervised learning frameworks are the two typical approaches for denoising. They both infuse some side information about the clean source one way or the other; namely, the Bayesian approach imposes some reasonable stochastic models, *e.g.*, Markov models, on the clean data, whereas the supervised learning approach collects representative clean and noisy data from which a denoiser can *learn* a mapping from noisy to clean. While the two approaches are effective in practice to some extent, they also possess limitations as well; namely, when the assumed model or the collected data mismatches with the characteristics of the actual data subject to denoising, serious performance degradation may follow.

To that end, [20] first considered a *universal* approach for discrete denoising; *i.e.*, nothing about the underlying clean source data was assumed nor collected, and a denoising rule was *adaptively* obtained solely from the observed noisy data. They devised a sliding-window algorithm called DUDE (Discrete Universal DEnoiser), which is rooted in information-theoretic universal data compression/prediction schemes, and demonstrated powerful theoretical guarantees as well as empirical performance. Despite the strong results, however, DUDE suffered from a couple of shortcomings as well; the performance of the algorithm deteriorates as the alphabet size grows and is quite sensitive to the choice of a hyperparameter, the window size k. To overcome such limitations, [14] recently proposed Neural DUDE (N-DUDE), a neural network-based sliding window denoiser that implicitly aggregates similar contexts via shared network parameters. Their algorithm maintained the robustness with respect to both k and the alphabet size, and as a result, N-DUDE achieved significantly better performance than DUDE. The main

Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR volume 124, 2020.

gist of N-DUDE was to devise "pseudo-labels" solely based on the noisy data, hence, the unsupervised training (without any clean data) of the neural network denoiser was still possible.

Although both DUDE and N-DUDE did not utilize any assumptions on the underlying clean data, one critical assumption they both made is that the statistical characteristics of the noise mechanism (*i.e.*, channel) is *known* to the denoiser. That is, the noise is modeled to be a Discrete Memoryless Channel (DMC), and the channel transition matrix was assumed to be completely known to the denoiser. While such an assumption makes sense in some applications, *e.g.*, when the noisy channel can be reliably estimated with known reference sequences, it can become a major weakness in competing with other fully unsupervised methods that do not require such assumption. For example, the Baum-Welch (BW) algorithm [1] combined with forward-backward (FB) recursion for hidden Markov models (HMM) [3] can both estimate the channel (*i.e.*, the emission probability) and the underlying clean data (*i.e.*, the latent states) as long as the noisy observation can be modeled as an HMM.

In this paper, we aim to remove the known noise assumption of N-DUDE. Namely, the *only* assumption we make is that the noise mechanism is a DMC (like in HMM), but neither the channel transition matrix nor characteristics of the clean data (such as Markovity) are assumed to be known. Thus, our setting is a much more challenging one than that of [20, 14] as we impose uncertainty on the noise model in addition to on the clean data[1]. We propose a novel unsupervised, iterative channel estimation (ICE) algorithm such that learning the channel transition matrix and the neural network parameters can be done in an alternating manner. The key component of our algorithm is to approximate the marginal posterior distribution of the clean data with a posterior *induced* from the N-DUDE's output and carry out the Expectation-Maximization (EM)-like variational inference.

In our experimental results with various types of data (*e.g.*, images or DNA sequences), we show the effectiveness of our ICE by showing that denoising with the *estimated* channel achieves almost identical denoising performance as with the *true* channel. We employ two neural network-based denoisers to evaluate the denoising performance with ICE; N-DUDE and CUDE (Context-aggregated Universal DEnoiser)[19], in which the former is what ICE is based on, and the latter is another recently developed universal denoiser that is shown to outperform N-DUDE. Both algorithms that plug-in the estimated channel by ICE are shown to outperform the widely used BW with FB recursion, which models the noisy data as an HMM regardless of it being true. In addition, we show ICE is much more robust with respect to its hyperparameters and initializations compared to BW, which is sensitive to the initial transition and channel models. Finally, we give thorough experimental analyses on the channel estimation errors as well as the convergence property of ICE.

[1] Such setting was initially considered in [5, 6], but mainly with a theoretical motivation.

2 Notations and Related Work

To be self-contained, we introduce notations that mainly follow [14]. Throughout the paper, an n-tuple sequence is denoted as, *e.g.*, $a^n = (a_1, \ldots, a_n)$, and a_i^j refers to the subsequence $(a_i, \ldots, a_j)$. We denote the uppercase letters as random variables and the lowercase letters as either the realizations of the random variables or the individual symbols. We denote Δ^d as the probability simplex in $\mathbb{R}^d$. We will denote the clean, underlying source data by x^n as we make no stochastic assumption on its distribution $p_{\mathcal{X}}(x^n)$ in our universal setting. We assume each component x_i takes a value in some finite set $\mathcal{X}$. For example, for binary data, $\mathcal{X} = \{0, 1\}$, and for DNA data, $\mathcal{X} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$.

We assume x^n is corrupted by a DMC, namely, the index-independent noise, and results in the noisy data, Z^n, of which each Z_i takes a value in, again, a finite set $\mathcal{Z}$. The DMC is characterized by the channel transition matrix $\mathbf{\Pi} \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{Z}|}$, and the (x, z)-th element of $\mathbf{\Pi}$ stands for $\Pr(Z = z|x)$. A natural assumption we make is that $\mathbf{\Pi}$ is of the *full row rank*, which holds for most practical settings. We also denote $\mathbf{\Pi}^\dagger = \mathbf{\Pi}^\top(\mathbf{\Pi}\mathbf{\Pi}^\top)^{-1}$ as the Moore-Penrose pseudoinverse of $\mathbf{\Pi}$. Now, upon observing the entire noisy data Z^n, a discrete denoiser reconstructs the original data with $\hat{X}^n = (\hat{X}_1(Z^n), \ldots, \hat{X}_n(Z^n))$, where each reconstructed symbol $\hat{X}_i(Z^n)$ takes its value in a finite set $\hat{\mathcal{X}}$. The goodness of the reconstruction is measured by the average denoising loss,

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{\Lambda}(x_i, \hat{X}_i(Z^n)),$$

where the per-symbol loss $\mathbf{\Lambda}(x_i, \hat{x}_i)$ measures the loss incurred by estimating x_i with $\hat{x}_i$. The loss is fully represented with a loss matrix $\mathbf{\Lambda} \in \mathbb{R}^{|\mathcal{X}|\times|\hat{\mathcal{X}}|}$.

The k-th order sliding window denoisers are the denoisers that are defined by a time-invariant mapping $s_k : \mathcal{Z}^{2k+1} \to \hat{\mathcal{X}}$. That is, $\hat{X}_i(Z^n) = s_k(Z_{i-k}^{i+k})$. We also denote the tuple $(Z_{i-k}^{i-1}, Z_{i+1}^{i+k}) \triangleq \mathbf{C}_i$ as the k-th order double-sided context around the noisy symbol Z_i, and we let $\mathbf{C}[k]$ as the set of all such contexts. We also denote $\mathcal{S} \triangleq \{s : \mathcal{Z} \to \hat{\mathcal{X}}\}$ as the set of *single-symbol denoisers* that are sliding window denoisers with $k = 0$. Note $|\mathcal{S}| = |\hat{\mathcal{X}}|^{|\mathcal{Z}|}$. Then, an alternative view of of $s_k(\cdot)$ is that

$s_k(\mathbf{C}_i, \cdot) \in \mathcal{S}$ is a single symbol denoiser defined by $\mathbf{C}_i$ and applied to Z_i.

When $\mathbf{\Pi}$ is known, as in [14, Section 3.1], we can devise an *unbiased esimate* of the true loss $\mathbf{\Lambda}$ as

$$\mathbf{L} = \mathbf{\Pi}^{\dagger}\boldsymbol{\rho} \in \mathbb{R}^{|\mathcal{Z}|\times|\mathcal{S}|}, \tag{1}$$

in which $\boldsymbol{\rho} \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{S}|}$ with the (x, s)-th element is $\mathbb{E}_{Z|x}\mathbf{\Lambda}(x, s(Z))$, and $\mathbb{E}_{Z|x}(\cdot)$ stands for the expectation with respect to the distribution defined by the x-th row of $\mathbf{\Pi}$. Then, as shown in [14, 21], $\mathbf{L}$ has the unbiased property, $\mathbb{E}_{Z|x}\mathbf{L}(Z, s) = \mathbb{E}_{Z|x}\mathbf{\Lambda}(x, s(Z))$.

2.1 Related work

DUDE [20] is a two-pass, sliding-window denoiser. For reconstruction at location i, DUDE takes $\mathbf{C}_i \in \mathbf{C}[k]$ and $Z_i \in \mathcal{Z}$ as input, and applies the rule

$$\begin{aligned}&\hat{X}_{i,\texttt{DUDE}}(\mathbf{C}_i, Z_i)\\ &= \arg\min_{\hat{x}\in\hat{\mathcal{X}}} \hat{\mathbf{p}}_{\text{emp},Z}(\cdot|\mathbf{C}_i)^{\top}\mathbf{\Pi}^{\dagger}[\mathbf{\Lambda}_{\hat{x}} \odot \boldsymbol{\pi}_{Z_i}],\end{aligned} \tag{2}$$

in which $\hat{\mathbf{p}}_{\text{emp}}(\cdot|\mathbf{C}_i) \in \mathbb{R}^{|\mathcal{Z}|}$ is an empirical probability vector on Z_i given the context vector $\mathbf{C}_i$, obtained from the entire noisy sequence Z^n. That is, for a context $\mathbf{C} \in \mathbf{C}[k]$, the z-th element becomes

$$\hat{\mathbf{p}}_{\text{emp}}(z|\mathbf{C}) = \frac{|\{j : \mathbf{C}_j = \mathbf{C}, Z_j = z\}|}{|\{j : \mathbf{C}_j = \mathbf{C}\}|}. \tag{3}$$

Moreover, the $\mathbf{\Lambda}_{\hat{x}}$ and $\boldsymbol{\pi}_{Z_i}$ in (2) stand for the $\hat{x}$-th and Z_i-th column of $\mathbf{\Lambda}$ and $\mathbf{\Pi}$, respectively. Note the rule (2) solely depends on Z^n and the knowledge of $\mathbf{\Pi}$ is required. The main intuition for obtaining (2) is to show that the following approximation to the true posterior distribution

$$p(x|\mathbf{C}_i, Z_i) \approx \left(\boldsymbol{\pi}_{Z_i} \odot [\mathbf{\Pi}^{\dagger\top}\hat{\mathbf{p}}_{\text{emp},Z}(\cdot|\mathbf{C}_i)]\right)_x \tag{4}$$

holds with high probability with large n [20, Section IV.B] and compute the Bayes response with respect to $\boldsymbol{\pi}_{Z_i} \odot [\mathbf{\Pi}^{\dagger\top}\hat{\mathbf{p}}_{\text{emp},Z}(\cdot|\mathbf{C}_i)]$. [20] showed (2) can universally attain the optimum denoising performance for *any* underlying stationary $p_{\mathcal{X}}(x^n)$.

Neural DUDE [14] identifies that the limitation of DUDE follows from the empirical count (3), which happens totally separately for each context $\mathbf{C}$. To that end, N-DUDE implements a *single* neural network-based sliding-window denoiser such that the information among similar contexts can be shared through the network parameters. Namely, N-DUDE defines $\mathbf{p}^k_{\texttt{N-DUDE}}(\mathbf{w}, \cdot) : \mathcal{Z}^{2k} \to \Delta^{|\mathcal{S}|}$, in which $\mathbf{w}$ stands for the parameters in the network; from the alternative view on the sliding-window denoiser mentioned above, $\mathbf{p}^k_{\texttt{N-DUDE}}(\mathbf{w}, \cdot)$ takes the context $\mathbf{C}_i$ and outputs a probability distribution on the single-symbol denoisers to apply to Z_i, for each i.

To train $\mathbf{w}$, the non-negative matrix $\mathbf{L}_{\text{new}} \in \mathbb{R}^{|\mathcal{Z}|\times|\mathcal{S}|}$, based on $\mathbf{L}$, is defined as

$$\mathbf{L}_{\text{new}} \triangleq -\mathbf{L} + L_{\max}\mathbf{1}_{|\mathcal{Z}|}\mathbf{1}^{\top}_{|\mathcal{S}|}, \tag{5}$$

in which $L_{\max} \triangleq \max_{z,s}\mathbf{L}(z, s)$, and $\mathbf{1}_{|\mathcal{Z}|}$ and $\mathbf{1}_{|\mathcal{S}|}$ stand for the all-1 vectors with $|\mathcal{Z}|$ and $|\mathcal{S}|$ dimensions, respectively. By design, $\mathbf{L}_{\text{new}}$ can be computed solely with $\mathbf{\Pi}$ and $\mathbf{\Lambda}$, and N-DUDE treats $\mathbf{L}^{\top}_{\text{new}}\mathbb{1}_{Z_i} \in \mathbb{R}^{|\mathcal{S}|}$ as the target "pseudo-label" vector for the mapping to apply at location i. Note $\mathbf{L}^{\top}_{\text{new}}\mathbb{1}_{Z_i}$ is not necessarily a one-hot vector, but from (1) and (5), we can observe that the mapping s with larger pseudo-label value should have smaller "true" loss *in expectation*. Then, the objective function of N-DUDE to train $\mathbf{w}$ becomes

$$\mathcal{L}(\mathbf{w}, Z^n; \mathbf{\Pi}) \triangleq \frac{1}{n}\sum_{i=1}^{n} \mathcal{C}\Big(\mathbf{L}^{\top}_{\text{new}}\mathbb{1}_{Z_i}, \mathbf{p}^k_{\texttt{N-DUDE}}(\mathbf{w}, \mathbf{C}_i)\Big), \tag{6}$$

in which $\mathcal{C}(\mathbf{g}, \mathbf{p})$ stands for the (unnormalized) cross-entropy. Note we highlighted the dependency of the objective function on Z^n and $\mathbf{\Pi}$, and the training of $\mathbf{w}$ is done via usual stochastic gradient descent.

Once (6) is minimized after sufficient number of iterations, the converged parameter is denoted as $\mathbf{w}^{\star}$. Then, the single-letter mapping defined by N-DUDE for the context $\mathbf{C} \in \mathbf{C}[k]$ is expressed as $s_{k,\texttt{N-DUDE}}(\mathbf{C}, \cdot) = \arg\max_{s\in\mathcal{S}} \mathbf{p}^k_{\texttt{N-DUDE}}(\mathbf{w}^{\star}, \mathbf{C})_s$, and the reconstruction at location i becomes

$$\hat{X}_{i,\text{N-DUDE}}(\mathbf{C}_i, Z_i) = s_{k,\text{N-DUDE}}(\mathbf{C}_i, Z_i). \tag{7}$$

In summary, N-DUDE is trained in an unsupervised manner, and [14] shows it significantly outperforms DUDE, has more robustness with respect to k, and achieves the *optimum* denoising performance for stationary sources.

CUDE [19] takes an alternative and simpler approach for using neural network to extend DUDE. Namely, instead of (3), CUDE learns a network $\mathbf{p}_{\texttt{CUDE}}(\mathbf{w}, \cdot) : \mathcal{Z}^{2k} \to \Delta^{|\mathcal{Z}|}$, which takes the context $\mathbf{C}_i$ as input and outputs a prediction for Z_i, by minimizing

$$\frac{1}{n}\sum_{i=1}^{n} \mathcal{C}(\mathbb{1}_{Z_i}, \mathbf{p}_{\texttt{CUDE}}(\mathbf{w}, \mathbf{C}_i)).$$

Note the difference between (6). Once the minimizer $\mathbf{w}^*$ is obtained, CUDE simply plugs in $\mathbf{p}_{\texttt{CUDE}}(\mathbf{w}^*, \mathbf{C}_i)$ in place of $\hat{\mathbf{p}}_{\text{emp}}(\cdot|\mathbf{C}_i)$ in (2). [19] shows CUDE outperforms N-DUDE primarily due to the reduced output size, *i.e.*, $|\mathcal{Z}|$ vs. $|\mathcal{S}| = |\hat{\mathcal{X}}|^{|\mathcal{Z}|}$.

Baum-Welch (BW) algorithm [1] for HMM From the equations (2) and (6), we confirm that all of above three schemes require the exact knowledge on the channel $\mathbf{\Pi}$. In contrast, as mentioned in Introduction, the Baum-Welch (BW) algorithm combined with forward-backward (FB) recursion for HMM is a powerful method that can denoise Z^n without requiring such knowledge on the channel. Despite the strength and being widely used in practice [10, 8], we note the BW based on HMM has some drawbacks, too. Firstly, the Markov assumption on the clean x^n may not be accurate; *i.e.*, x^n may not have generated from a Markov source or the assumed order could have a mismatch from the true model. In such cases, the resulting BW and FB recursion based denoising will have poor performance. Secondly, the BW-based channel estimation may suffer from instability with respect to the initialization of the algorithm. In the later sections, we convincingly show that our proposed ICE can reliably estimate the channel, and N-DUDE or CUDE that plugs in the estimated channel can overcome the drawbacks of BW and achieve significantly better denoising performance in realistic discrete data.

3 Iterative Channel Estimation (ICE)

We first give a succinct description of our ICE algorithm, then elaborate its theoretical motivation and intuition.

3.1 Description of ICE

The ICE alternates between the following two steps to estimate $\mathbf{\Pi}$ and learn $\mathbf{w}$ jointly until the objective (6) converges. The algorithm starts with randomly initialized $\mathbf{\Pi}^{(0)}$ and $\mathbf{w}^{(0)}$.

(1) Approximate E-step (update w): Assuming the t-th estimate of $\mathbf{\Pi}$, $\mathbf{\Pi}^{(t)}$, is given, the network parameter of N-DUDE is then updated by obtaining

$$\mathbf{w}^{(t+1)} = \arg\min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, Z^n; \mathbf{\Pi}^{(t)}). \tag{8}$$

When carrying out the minimization in (8), we always do a warm-start from the weight of the previous iteration, $\mathbf{w}^{(t)}$, except for the first iteration. Now, using $\mathbf{w}^{(t+1)}$, we obtain an induced posterior

$$q(x_i|Z_{i-k}^{i+k}; \mathbf{w}^{(t+1)}) \triangleq \sum_{s:s(Z_i)=x_i} \mathbf{p}^k_{\texttt{N-DUDE}}(\mathbf{w}^{(t+1)}, \mathbf{C}_i)_s \tag{9}$$

for each index i. Note (9) is a partial sum of the N-DUDE output probabilities and is a valid posterior distribution of x_i since it is nonnegative and sums up to 1.

(2) M-step (update $\mathbf{\Pi}$): Using $\mathbf{w}^{(t+1)}$ and (9), the (j, ℓ)-th element of $\mathbf{\Pi}^{(t+1)}$ is obtained by computing

$$\mathbf{\Pi}^{(t+1)}(j, \ell) = \frac{\sum_{i=1}^n \mathbb{1}_{\{Z_i=\ell\}} q(x_i = j|Z_{i-k}^{i+k}; \mathbf{w}^{(t+1)})}{\sum_{i=1}^n q(x_i = j|Z_{i-k}^{i+k}; \mathbf{w}^{(t+1)})}. \tag{10}$$

Note this step looks similar to the M-step of BW for HMM, and we elaborate more about the update steps in the intuition section below.

Once the iteration converges, we can do a final weight update (8) with the estimated channel $\hat{\mathbf{\Pi}}$ and obtain the final parameters of N-DUDE, $\hat{\mathbf{w}}$. In our experimental results, we show the objective (6) with true $\mathbf{\Pi}$ nicely converges as the iterative updates for $\mathbf{\Pi}^{(t)}$ and $\mathbf{w}^{(t)}$ continues.

3.2 Theoretical motivation of ICE

The derivation of the ICE algorithm is based on extending the usual maximum likelihood estimation and variational inference arguments. First, denote $p(x^n, Z^n; \mathbf{\Pi}^{(t)})$ as the joint distribution of (x^n, Z^n) induced from an unknown source distribution $p_{\mathcal{X}}(x^n)$ and the t-th channel estimate $\mathbf{\Pi}^{(t)}$, *i.e.*,

$$\begin{aligned} p(x^n, Z^n; \mathbf{\Pi}^{(t)}) &= p_{\mathcal{X}}(x^n) p(Z^n|x^n; \mathbf{\Pi}^{(t)}) \\ &= p_{\mathcal{X}}(x^n) \prod_{i=1}^n \mathbf{\Pi}^{(t)}(x_i, Z_i). \end{aligned} \tag{11}$$

Now, the standard evidence lower bound (ELBO) on the log-likelihood of Z^n with $\mathbf{\Pi}^{(t)}$ becomes

$$\begin{aligned} &\log p(Z^n; \mathbf{\Pi}^{(t)}) \geq \sum_{x^n} Q(x^n) \log \frac{p(x^n, Z^n; \mathbf{\Pi}^{(t)})}{Q(x^n)} \\ &= \log p(Z^n; \mathbf{\Pi}^{(t)}) - D\big(Q(x^n) \| p(x^n|Z^n; \mathbf{\Pi}^{(t)})\big), \end{aligned} \tag{12}$$

in which $Q(x^n)$ stands for an arbitrary probability distribution on x^n, and $D(\cdot\|\cdot)$ is the Kullback-Leibler divergence. In the standard Expectation-Maximization (EM), for a fixed $\mathbf{\Pi}^{(t)}$, $Q(x^n)$ that maximizes (12) obtained by the E-step is

$$Q(x^n; \mathbf{\Pi}^{(t)}) = p(x^n|Z^n; \mathbf{\Pi}^{(t)}), \tag{13}$$

the posterior of x^n given Z^n derived from the joint distribution $p(x^n, Z^n; \mathbf{\Pi}^{(t)})$. Then, the standard M-step obtains

$$\mathbf{\Pi}^{(t+1)} = \arg\max_{\mathbf{\Pi}} \text{ELBO}(\mathbf{\Pi}^{(t)}, \mathbf{\Pi}), \tag{14}$$

in which

$$\text{ELBO}(\mathbf{\Pi}_1, \mathbf{\Pi}_2) \triangleq \sum_{x^n} Q(x^n; \mathbf{\Pi}_1) \log \frac{p(x^n, Z^n; \mathbf{\Pi}_2)}{Q(x^n; \mathbf{\Pi}_1)}. \tag{15}$$

Such iteration results in monotonically increasing ELBO for the log-likelihood and is shown to converge. Now,

for certain source distribution $p_{\mathcal{X}}(x^n)$ that has particular structures, *e.g.*, Markovity, then, computing (13) and $\mathbf{\Pi}^{(t+1)}$ become computationally efficient, as in the BW algorithm for HMM.

Unlike the case of HMM, in our universal setting in which no distributional assumption on $p_{\mathcal{X}}(x^n)$ is made, exactly carrying out the E- and M-steps become intractable. To that end, ICE carries out an approximate ELBO maximization by the above described iterative scheme. Before elaborating on our intuition used for the approximation, which is built from the result of N-DUDE, we first give the following Lemma.

Lemma 1 *Suppose $Q(x^n;\mathbf{\Pi}^{(t)})$ has the following form,*

$$Q(x^n;\mathbf{\Pi}^{(t)}) \triangleq \prod_{i=1}^{n} Q(x_i|Z^n;\mathbf{\Pi}^{(t)}), \tag{16}$$

namely, it can be factored into the product of the marginal posteriors. Then, the M-step results in $\mathbf{\Pi}^{(t+1)}$ with

$$\mathbf{\Pi}^{(t+1)}(j,\ell) = \frac{\sum_{i=1}^{n} \mathbb{1}_{\{Z_i=\ell\}} Q(x_i=j|Z^n;\mathbf{\Pi}^{(t)})}{\sum_{i=1}^{n} Q(x_i=j|Z^n;\mathbf{\Pi}^{(t)})}. \tag{17}$$

Proof sketch: The full proof is given in the Supplementary Material, and we only give the sketch of the proof here. Namely, we simplify the maximization problem of (14) by exploiting the memoryless property of the channel, $\mathbf{\Pi}$. Then, we consider the Lagrangian dual for the maximization problem, obtain the expression of the dual variable in terms of $Q(x^n,\mathbf{\Pi}^{(t)})$, and exploit the factorizing assumption (16) to obtain the expression (17). ■

Motivated by the lemma, we set $Q(x^n;\mathbf{\Pi}^{(t)})$ for the *approximate E-step* in ICE as following:

$$Q(x^n;\mathbf{\Pi}^{(t)}) \triangleq \prod_{i=1}^{n} q(x_i|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)}), \tag{18}$$

in which $q(x_i|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)})$ is defined in (8),(9) and is in the form (16). Then, our M-step result (10) follows from Lemma 1. Now, using (15), we make the following assumption, of which intuition is given below.

Assumption 1 *For $\mathbf{\Pi}^{(t+1)}$ in (10), we assume*

$$\textit{ELBO}(\mathbf{\Pi}^{(t+1)},\mathbf{\Pi}^{(t+1)}) \geq \textit{ELBO}(\mathbf{\Pi}^{(t)},\mathbf{\Pi}^{(t+1)}). \tag{19}$$

We believe (19) intuitively makes sense since it implies

$$\begin{aligned} &D(Q(x^n;\mathbf{\Pi}^{(t)})||p(x^n|Z^n;\mathbf{\Pi}^{(t+1)})) \\ \geq &D(Q(x^n;\mathbf{\Pi}^{(t+1)})||p(x^n|Z^n;\mathbf{\Pi}^{(t+1)})). \end{aligned} \tag{20}$$

Namely, note $Q(x^n;\mathbf{\Pi}^{(t+1)})$ is defined by the induced marginal posteriors from the N-DUDE that is trained with the pseudo-labels computed with $\mathbf{\Pi}^{(t+1)}$. The inequality (20) asserts that $Q(x^n;\mathbf{\Pi}^{(t+1)})$ is closer to the posterior distribution $p(x^n|Z^n;\mathbf{\Pi}^{(t+1)})$ in the KL-sense than $Q(x^n;\mathbf{\Pi}^{(t)})$, which is obtained by training N-DUDE with the *mismtched* pseudo-labels computed with $\mathbf{\Pi}^{(t)}$. We believe this makes sense because N-DUDE trained with true $\mathbf{\Pi}$ is empirically shown (in [14]) to attain the optimum denoising performance for stationary sources; namley, since achieving the optimum performance requires the knowledge of $p(x_i|Z^n;\mathbf{\Pi})$, we can expect the induced posterior $q(x_i|Z_{i-k}^{i+k};\mathbf{w}^\star)$ of N-DUDE, which is learned with pseudo-labels computed with $\mathbf{\Pi}$, approximates $p(x_i|Z^n;\mathbf{\Pi})$ well for sufficiently large k.

Combining Assumption 1 with the M-step, we then derive

$$\text{ELBO}(\mathbf{\Pi}^{(t+1)},\mathbf{\Pi}^{(t+1)}) \geq \text{ELBO}(\mathbf{\Pi}^{(t)},\mathbf{\Pi}^{(t)}), \tag{21}$$

which shows that our ICE iteration also results in the monotonically increasing ELBO, guaranteeing the convergence of the iteration. In our experiments, we convincingly show the convergence of ICE on various source data, which advocates the validity of our Assumption 1.

Algorithm 1 ICE algorithm with N-DUDE

Require: Noisy data Z^n

Ensure: Channel estimate $\hat{\mathbf{\Pi}}$, Network parameters $\hat{\mathbf{w}}$

Initialize $\mathbf{\Pi}^{(0)}$ and $\mathbf{w}^{(0)}$ and fix window size k

Set $t \leftarrow 0$, $\epsilon = 10^{-3}$

while $|\mathcal{L}(\mathbf{w},Z^n;\mathbf{\Pi}^{(t)}) - \mathcal{L}(\mathbf{w},Z^n;\mathbf{\Pi}^{(t-1)})| > \epsilon$ **do**

Compute $\mathbf{L}_{\text{new}}$ in (5) using $\mathbf{\Pi}^{(t)}$.

`/* Approx. E-step (Update` $\mathbf{w}$`) */`

Using (8) and (9), obtain $\mathbf{w}^{(t+1)}$ and the induced posterior for each location i, $q(x_i|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)})$, as:

$$\mathbf{w}^{(t+1)} = \arg\min_{\mathbf{w}} \mathcal{L}(\mathbf{w},Z^n;\mathbf{\Pi}^{(t)})$$

$$q(x_i|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)}) \triangleq \sum_{s:s(Z_i)=x_i} \mathbf{p}^k(\mathbf{w}^{(t+1)},\mathbf{C}_i)_s$$

`/* M-step (Update` $\mathbf{\Pi}$`) */`

Using (10), obtain $\mathbf{\Pi}^{(t+1)}$ as

$$\mathbf{\Pi}^{(t+1)}(j,k) = \frac{\sum_{i=1}^{n} \mathbb{1}_{\{Z_i=k\}} q(x_i=j|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)})}{\sum_{i=1}^{n} q(x_i=j|Z_{i-k}^{i+k};\mathbf{w}^{(t+1)})}.$$

$t \leftarrow t+1$

end while

$\hat{\mathbf{\Pi}} = \mathbf{\Pi}^{(t)}$

Do a final update (8) with $\hat{\mathbf{\Pi}}$, $\mathbf{w}^{(t)}$ and obtain $\hat{\mathbf{w}}$

3.3 Algorithm summary and remarks on ICE

Algorithm 1 summarizes our ICE algorithm with a concrete stopping criterion we used for the experiments.

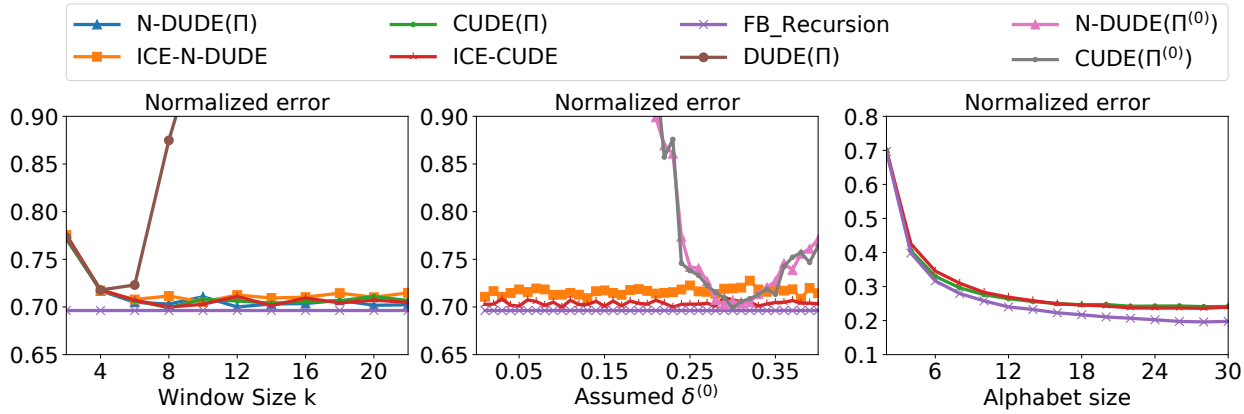


Figure 1: Denosing results for HMM with respect to window size k (left), assumed $\delta^{(0)}$ (center), and the alphabet size $|\mathcal{X}|$ (right). The vertical axes for three figures correspond to the normalized error rates.

Remark 1: One important point to make is that it is *not* possible to devise an iterative channel estimation scheme like ICE based on CUDE [19]. The reason is that in CUDE, the channel $\mathbf{\Pi}$ only occurs in the final denoising rule (2), and learning the neural network has nothing to do with $\mathbf{\Pi}$. In N-DUDE, however, $\mathbf{\Pi}$ is used to compute the pseudo-labels (5) for training the network, hence, an updated channel would lead to an updated network. Moreover, from the updated network, N-DUDE can naturally induce the marginal posterior (9) to carry out the M-step to update $\mathbf{\Pi}$. While the channel estimation cannot be done based on CUDE, the ICE-estimated (using N-DUDE) channel can be *plugged-in* to CUDE to carry out the denoising. We denote such a scheme as ICE-CUDE and show strong performance in the experimental results.

Remark 2: Another important detail for implementing our algorithm is to address one limitation of the original N-DUDE. Namely, the N-DUDE has the output size of $|\mathcal{S}| = |\hat{\mathcal{X}}|^{|\mathcal{Z}|}$, hence, it grows quickly as the alphabet size of data grows, which makes the induced posterior (9) suffer from high variance. Therefore, to make ICE scalable with respect to large alphabets, we reduced the output dimension of the network from $|\mathcal{S}| = |\hat{\mathcal{X}}|^{|\mathcal{Z}|}$ to $|\mathcal{S}| = |\hat{\mathcal{X}}| + 1$ by simplifying the denoising to either "saying-what-you-see" (i.e., $s(Z_i) = Z_i$) or "saying-one-in-$\hat{\mathcal{X}}$. As a result, the reduced output grows only linearly as the alphabet size grows, and the channel estimation as well as denoising becomes more accurate. We give a more detailed explanation on the output dimension reduction in Supplementary Materials.

4 Experimental Results

Data and training details We carry out extensive experiments using synthetic data, real binary images, and Oxford nanopore MinION DNA sequence data [2] to show the effectiveness and robustness of our ICE algorithm. All the experiments were done with Python 3.6 and Keras with Tensorflow backend. For the *approximate E-step* in (8), we used the Adam optimizer [9] with default setting to minimize the objective function. The number of epochs for each iteration was set to 10 for synthetic and real binary image data and 20 for DNA data. The initial learning rate for the first iteration was 10^{-3}, then from the second iteration, we used 10^{-4}. As shown below in Figure 3, the objective function quickly converges after a few iterations, hence, we stopped the estimation process after the third iteration in all of our experiments. For the network architecture, we used 3 fully connected layers with 40 hidden nodes for synthetic/binary image data and used 3 layers of 1-dimensional 1×1 convolution layer with 160 channels for DNA data[2]. For all our implementation, we used the model with the reduced output dimensions as described in above Remark 2.

4.1 Synthetic data

First, we carry out experiment on synthetic data to validate the performance of ICE. Following [14], we generated the clean binary data from a binary symmetric Markov chain (BSMC) with transition probability $\alpha = 0.1$. The data was corrupted by a binary symmetric channel (BSC) $\mathbf{\Pi}$ with cross-over probability $\delta = 0.3$ to result in the noisy sequence Z^n, which becomes a hidden Markov process. The length of the sequence was set to $n = 10^6$, and the Hamming loss was used to set the Bit Error Rate (BER) as $\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{x_i \neq \hat{x}_i\}$. We report the normalized error, obtained by dividing BER with δ. The denoising results on this data are given in the left and center plots

[2]Note the fully-connected architecture can be equivalently implemented with 1×1 convolutions.

Table 1: Denoising results for real binary images.

Noise level	$\delta = 0.1$		$\delta = 0.2$		$\delta = 0.3$	
Methods \Dataset	PASCAL	Standard	PASCAL	Standard	PASCAL	Standard
BW_1st	0.4294	0.5469	0.3508	0.4726	0.5088	0.5716
BW_2nd	0.4770	0.6342	0.4025	0.5149	0.3829	0.5020
BW_3rd	0.5996	1.0943	0.5619	0.7523	0.5277	0.7415
ICE-N-DUDE	0.3516	0.4120	0.3223	0.3771	**0.3429**	0.4286
ICE-CUDE	**0.3512**	**0.4038**	**0.3205**	**0.3712**	0.3438	**0.4266**
N-DUDE ($\mathbf{\Pi}$)	0.3540	0.3981	0.3300	0.3826	0.3494	0.4524
CUDE ($\mathbf{\Pi}$)	0.3259	0.3748	0.3171	0.3684	0.3396	0.4245

of Figure 1, in which the left shows with respect to the window size k and the center shows with respect to the initially assumed $\mathbf{\Pi}^{(0)}$. Note since Z^n is a hidden Markov process in this case, the FB-Recursion that *knows* $\mathbf{\Pi}$ and the state transition probability α can achieve the optimum denoising performance, shown as purple lines (*i.e.*, lower bounds) in the Figure 1.

In Figure 1 (left), DUDE($\mathbf{\Pi}$), N-DUDE($\mathbf{\Pi}$) and CUDE($\mathbf{\Pi}$) stand for the results of the three schemes that exactly know the true channel $\mathbf{\Pi}$. We can confirm the universality of those methods since they almost achieve the optimum performance, while not knowing the source is a Markov. Also, N-DUDE($\mathbf{\Pi}$) and CUDE($\mathbf{\Pi}$) are much more robust with respect to k than DUDE($\mathbf{\Pi}$). For our ICE, we initialized $\mathbf{\Pi}^{(0)}$ as BSC with crossover probability $\delta^{(0)} = 0.1$, and show the results of two variants, ICE-N-DUDE and ICE-CUDE. Both *plug-in* the estimated channel by ICE to N-DUDE and CUDE, respectively, and we observe they work very well and essentially achieve the same performances as their counterparts that know $\mathbf{\Pi}$. We stress that this is a nontrivial result since ICE just observes Z^n and provides an accurate enough estimation of $\mathbf{\Pi}$ to achieve the optimum performance, only with the independent noise assumption.

In Figure 1 (center), we show the robustness of ICE with respect to varying initially assumed $\delta^{(0)}$, while fixing the window size $k = 16$. In the figure, N-DUDE($\mathbf{\Pi}^{(0)}$) and CUDE($\mathbf{\Pi}^{(0)}$) are the schemes that run with $\mathbf{\Pi}^{(0)}$, which can be potentially mismatched with the true $\mathbf{\Pi}$, and we clearly see they become very sensitive to the mismatch of the assumed $\mathbf{\Pi}^{(0)}$. In contrast, ICE becomes extremely robust to the initial $\mathbf{\Pi}^{(0)}$ such that both ICE-N-DUDE and ICE-CUDE almost achieve the optimum performance regardless of the initially assumed $\delta^{(0)}$.

In Figure 1 (right), we also investigate the performance of ICE with respect to the alphabet size of data. We increased the alphabet size of the Markov source, $|\mathcal{X}|$ (and $|\mathcal{Z}|$), from 2 to 30, and for each case, the transition probabilities from a state to others were set to be uniform as $0.1/(|\mathcal{X}| - 1)$. n was 5×10^6, and the true $\mathbf{\Pi}$ was set such that $\mathbf{\Pi}(i,i) = 0.7$ and $\mathbf{\Pi}(i,j) = 0.3/(|\mathcal{Z}| - 1)$ for $i \neq j$. The initial $\mathbf{\Pi}^{(0)}$ for ICE was set such that $\mathbf{\Pi}^{(0)}(i,i) = 0.9$ and $\mathbf{\Pi}^{(0)}(i,j) = 0.1/(|\mathcal{Z}| - 1)$ for $i \neq j$. We compared CUDE($\mathbf{\Pi}$), ICE-CUDE and FB-Recursion, and again, ICE-CUDE performs almost as well as CUDE($\mathbf{\Pi}$), robustly over the alphabet sizes. The gap from FB-Recursion is primarily due to fixing n, and we believe it will close as n grows with the alphabet size.

Table 2: Denoising results on randomized channels.

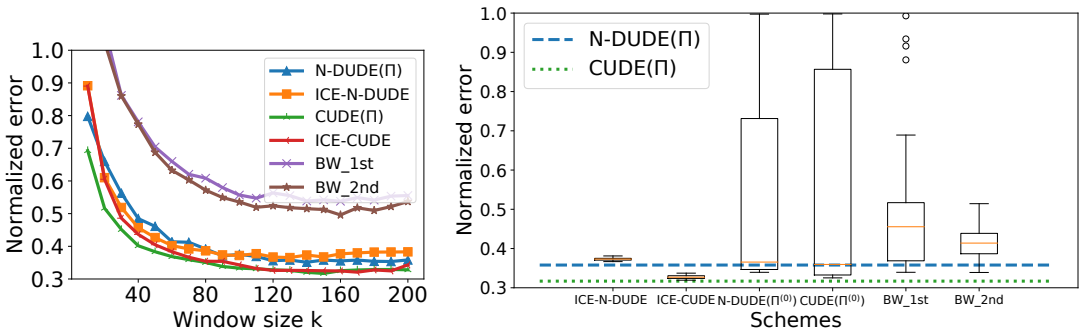
Method \ $\lvert\mathcal{X}\rvert$	2	4	6	12	18	24	30
CUDE ($\mathbf{\Pi}$)	0.619	0.353	0.283	0.231	0.227	0.213	0.212
ICE-CUDE	0.635	0.402	0.319	0.251	0.226	0.220	0.230

In Table 2, we show the results for more challenging channels beyond the symmetric channels used in Figure 1. Namely, we again compare the denoising performance of CUDE($\mathbf{\Pi}$) and ICE-CUDE, for the same Markov source (with increasing alphabet size) as in Figure 1(right), but with randomized channels — we selected each diagonal element of a channel matrix from $[0.7, 0.8]$, then randomized the off-diagonal elements. The initial $\mathbf{\Pi}^{(0)}$ for ICE was the same as we used in Figure 1(right), and the table shows the average of 5 independent runs for each alphabet size. From the table, due to the channel irregularity, we observe slightly larger gaps between CUDE($\mathbf{\Pi}$) and ICE-CUDE than in the symmetric channel case of Figure 1, but they are still sufficiently small.

4.2 Binary images

Now, we move on to the experiments using more realistic binary images as clean data. We tested on two datasets: PASCAL and Standard. PASCAL consists of 50 binarized grayscale images that we obtained from PASCAL VOC 2012 dataset [4], and Standard consists of 8 binarized standard images that are widely used in image processing, {`Barbara`, `Boat`, `C.man`, `Couple`, `Einstein`, `fruit`, `Lena`, `Peppers`}. We tested with three noise levels and applied non-symmetric channels with average noise levels of 0.1, 0.2, and 0.3, and the exact $\mathbf{\Pi}$'s are given in the Supplementary Material. As in [14], we raster scanned the images and converted them to 1-D sequences.

In Table 1, we compare the normalized errors of ICE-N-DUDE and ICE-CUDE with Baum-Welch (BW) that assume the images are Markov. BW_1st, BW_2nd, and



(a) Error rate with respect to window size k

(b) Boxplots of error rates with varying initial $\delta^{(0)}$.

Figure 2: Denosing results for real DNA data with respect to window size k & initial $\mathbf{\Pi}^{(0)}$.

BW_3rd correspond to BW with various Markov order assumptions. N-DUDE($\mathbf{\Pi}$) and CUDE($\mathbf{\Pi}$), which are known to achieve the state-of-the-art for binary image denoising with *known* channel, are shown as lower bounds. We fixed the window size to $k = 50$ for all neural network based schemes. For both ICE and BW, we set the initial $\mathbf{\Pi}^{(0)}$ as BSC with $\delta^{(0)} = 0.1$ for all noise levels. The exact procedure of running ICE and training denoisers with multiple images are given in the Supplementary Material.

In the table, we see that both ICE-N-DUDE and ICE-CUDE significantly outperform all three BW methods for all noise levels and datasets, and they get very close to N-DUDE($\mathbf{\Pi}$) and CUDE($\mathbf{\Pi}$), respectively. We also confirm the superiority of CUDE over N-DUDE, as claimed in [19]. Moreover, note the BW schemes become sensitive depending on the noise level and dataset; *i.e.*, for $\delta = 0.1$ and 0.2, BW_1st performs the best, while for $\delta = 0.3$, BW_2nd is superior. Such difficulty of accurately determining the best order of HMM for a given dataset is one of the main drawbacks of BW method. On the contrary, ICE-N-DUDE and ICE-CUDE work universally well for all sources and noise levels, and neither the clean source modeling nor the true channel $\mathbf{\Pi}$ was necessary.

4.3 DNA sequence

We now apply ICE to DNA sequence denoising and mainly follow the experimental setting of [14, Section 5.3]; namely, we obtained 16S rDNA reference sequences for 20 species and randomly generated noiseless template reads x^n of length $n = 2,469,111$. Then, we used the same $\mathbf{\Pi}$ in [14], which had 20.375% average error rate, to simulate the Oxford Nanopore sequencer and corrupt x^n to obtain Z^n. The true (asymmetric) $\mathbf{\Pi}$ is given in the Supplementary Material. For ICE, the initial $\mathbf{\Pi}^{(0)}$ was assumed to be $\mathbf{\Pi}^{(0)}(i,i) = 0.6$ and $\mathbf{\Pi}^{(0)}(i,j) = 0.4/(|\mathcal{Z}| - 1)$ for $i \neq j$. We also abuse the notation and define $\delta^{(0)} \triangleq \sum_{j \neq i} \mathbf{\Pi}^{(0)}(i,j)$ which becomes 0.4 for all i. Note as shown in [12, 14], DUDE($\mathbf{\Pi}$) and N-DUDE($\mathbf{\Pi}$) can achieve the state-of-the-art for DNA sequence denoising as well.

Figure 2(a) shows the denoising results with varying window size k, and Figure 2(b) shows the boxplots of normalized errors with window size $k = 150$ and varying initial $\delta^{(0)}$'s within the range $0.01 \sim 0.40$ (40 samples). BW_1st and BW_2nd completely failed for this experiment, resulting in the normalized error rate of 1.440 and 3.153, respectively. Thus, we could not include them in the figure, and we instead included the results of a hybrid method, i.e, running N-DUDE with BW estimated channels. BW_1st and BW_2nd in the figure stand for such a hybrid of BW with N-DUDE.

Paralleling the results in the previous sections, we observe from Figure 2(a) that ICE-N-DUDE and ICE-CUDE get very close to N-DUDE($\mathbf{\Pi}$) and CUDE($\mathbf{\Pi}$), respectively, and significantly outperform the BW hybrid methods, as k increases. This shows the accuracy and effectiveness of ICE; its channel estimation quality is much better than BW when the underlying x^n is far from being a Markov and while it is based on N-DUDE, the estimated channel can be readily plugged-in to other schemes like CUDE. We believe this is quite a strong result since ICE-CUDE can remove almost 70% of noise solely based on Z^n and with no other information on the noise and the clean source.

Moreover, in Figure 2(b), we see that ICE-N-DUDE and ICE-CUDE are extremely robust with respect to the initial $\delta^{(0)}$, while the mismatched N-DUDE($\mathbf{\Pi}^{(0)}$) and CUDE($\mathbf{\Pi}^{(0)}$) completely fails for wrong initializations. Additionally, the BW hybrid methods also show large variance, mainly due to the sensitivity of the channel estimation quality of BW with respect to the initial $\delta^{(0)}$.

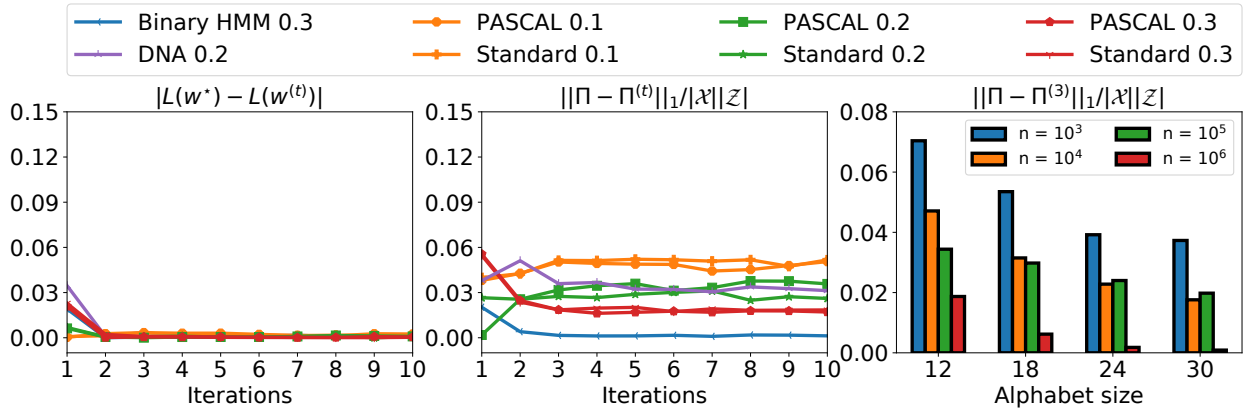


Figure 3: Convergence and estimation performance analysis with respect to iteration t

4.4 Convergence & estimation analyses

We now give a closer analysis on the channel estimation performance of ICE for all of our experiments given in the above sections. Figure 3 shows the following two metrics;

- $|\mathcal{L}(\mathbf{w}^\star, Z^n; \mathbf{\Pi}) - \mathcal{L}(\mathbf{w}^{(t)}, Z^n; \mathbf{\Pi})|$
- $\|\mathbf{\Pi} - \mathbf{\Pi}^{(t)}\|_1/|\mathcal{X}||\mathcal{Z}|$

The first metric shows the difference between the value of the objective function (6) for N-DUDE($\mathbf{\Pi}$) parameter $\mathbf{w}^\star$ and for the model $\mathbf{w}^{(t)}$ after each approximate E-step of ICE. Note (6) is computed with the true $\mathbf{\Pi}$. The second metric is the normalized L_1-norm of $\mathbf{\Pi} - \mathbf{\Pi}^{(t)}$, which directly measures the channel estimation accuracy.

The first two figures in Figure 3 show the two metrics for all experiments with respect to the iteration t of ICE, respectively, except for the large alphabet Markov source case. For the first metric, we observe that the difference becomes very small after just a few iterations for all cases. The result suggests that $\mathbf{w}^{(t)}$ of ICE and $\mathbf{w}^\star$ become indistinguishable from the perspective of objective function value, hence, it justifies the good performance of ICE-N-DUDE in denoising experiments. For the second metric, we observe the channel estimation errors also become small and stable with respect to the iteration t. This result empirically confirms the convergence of $\mathbf{\Pi}^{(t)}$, as elaborated in the theoretical motivation section above, and advocates the validity of our Assumption 1. Moreover, the excellent denoising performance of ICE-CUDE, which just plugs-in the estimated channel to CUDE, confirms that the level of the estimation error in the figure is tolerable and has a negligible effect in denoising. Moreover, the estimation errors seem to get smaller for larger noise levels. The third figure shows the second metric at iteration 3 for the synthetic Markov source data, with respect to the alphabet size $|\mathcal{X}| = 12 \sim 18$ and the sequence length $n = 10^3 \sim 10^6$. We clearly observe that the estimation errors become smaller as the sequence length increases, and for sufficiently large n, the estimation error approaches 0 for $|\mathcal{X}| = 18 \sim 30$.

5 Discussion and Concluding Remarks

In this paper, we proposed a novel iterative channel estimation method for removing the known channel assumption of the recently developed N-DUDE. The resulting ICE-N-DUDE and ICE-CUDE achieved excellent denoising performance for various types of data, without *any* knowledge on the channel and the clean source, except for the DMC matrix being invertible. Some limitations also exist. First, the computational cost for ICE is relatively high since it requires multiple model training. It is a necessary price to pay for the channel uncertainly, however, as in Figure 3, the number of iterations tends to be not large. Second, as shown in various experiments, ICE tends to perform better for higher noise rates than for the low noise regime. For future work, we plan to extend ICE to more general settings, e.g., to general state estimation beyond denoising and to the continuous-alphabet case.

Acknowledgments

This work is supported in part by IITP grant funded by the Korea government [No.2016-0-00563, Research on adaptive machine learning technology development for intelligent autonomous digital companion], [No.2019-0-00421, AI Graduate School Support Program], [No.2019-0-01396, Development of framework for analyzing, detecting, mitigating of bias in AI model and training data], and [IITP-2020-2018-0-01798, ITRC Support Program]. The authors also thank Tsachy Weissman for his helpful discussions on BW-hybrid schemes.

References

[1] L. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(164-171), 1970.

[2] A. Benítez-Páez, K. J. Portune, and Y. Sanz. Species-level resolution of 16s rrna gene amplicons sequenced through the minion™ portable nanopore sequencer. *Gigascience*, 5(1):4, 2016.

[3] Y. Ephraim and N. Merhav. Hidden markov processes. *IEEE Trans. Inform. Theory*, 48(6):1518–1569, 2002.

[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. `http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html`.

[5] G. Gemelos, S. Sigurjonsson, and T. Weissman. Algorithms for discrete denoising under channel uncertainty. *IEEE Trans. Signal Process.*, 54(6):2263–2276, 2006.

[6] G. Gemelos, S. Sigurjonsson, and T. Weissman. Universal minimax discrete denoising under channel uncertainty. *IEEE Trans. Inform. Theory*, 52:3476–3497, 2006.

[7] S. Goodwin, J. Gurtowski, S. Ethe-Sayers, P. Deshpande, M. Schatz, and W. McCombie. Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome. *Genome Res.*, 2015.

[8] X. Huang, Y. Ariki, and M. Jack. *Hidden Markov Models for Speech Recognition*. Edinburgh University Press, Edinburgh, 1990.

[9] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[10] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler. Hidden Markov models in computational biology: Applications to protein modeling. *Journal of Molecular Biology*, 235:1501–1531, Feb. 1994.

[11] D. Laehnemann, A. Borkhardt, and A. McHardy. Denoising DNA deep sequencing data– high-throughput sequencing errors and their corrections. *Brief Bioinform*, 17(1):154–179, 2016.

[12] B. Lee, T. Moon, S. Yoon, and T. Weissman. DUDE-Seq: Fast, flexible, and robust denoising for targeted amplicon sequencing. *PLoS ONE, 12(7):e0181463*, 2017.

[13] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.

[14] T. Moon, S. Min, B. Lee, and S. Yoon. Neural universal discrete denosier. In *Neural Information Processing Systems (NIPS)*, 2016.

[15] G. Motta, E. Ordentlich, I. Ramirez, G. Seroussi, and M. J. Weinberger. The iDUDE framework for grayscale image denoising. *IEEE Trans. Image Processing*, 20:1–21, 2011.

[16] E. Ordentlich, G. Seroussi, S. Verdú, M. Weinberger, and T. Weissman. A universal discrete image denoiser and its application to binary images. In *IEEE ICIP*, 2003.

[17] E. Ordentlich, G. Seroussi, S. Verdú, and K. Viswanathan. Universal algorithms for channel decoding of uncompressed sources. *IEEE Trans. Inform. Theory*, 54(5):2243–2262, 2008.

[18] D. Romero, S.-J. Kim, G. B. Giannakis, and R. Lopez-Valcarce. Learning power spectrum maps from quantized power measurements. *IEEE Trans. Signal Process.*, 2547 - 2560, 2017.

[19] J. Ryu and Y.-H. Kim. Conditional distribution learning with neural networks and its application to universal image denoising. pages 3214–3218, 10 2018. doi: 10.1109/ICIP.2018.8451573.

[20] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M. Weinberger. Universal discrete denoising: Known channel. *IEEE Trans. Inform. Theory*, 51(1): 5–28, 2005.

[21] T. Weissman, E. Ordentlich, M. Weinberger, A. Somekh-Baruch, and N. Merhav. Universal filtering via prediction. *IEEE Trans. Inform. Theory*, 53 (4):1253–1264, 2007.