An ETF view of Dropout regularization

Dor Bank
dorbank@gmail.com
Raja Giryes
raja@tauex.tau.ac.il

School of Electrical Engineering
The Iby and Aladar Fleischman Faculty of Engineering
Tel Aviv University
Tel Aviv, Israel

Abstract

Dropout is a popular regularization technique in deep learning. Yet, the reason for its success is still not fully understood. This paper provides a new interpretation of Dropout from a frame theory perspective. By drawing a connection to recent developments in analog channel coding, we suggest that for a certain family of autoencoders with a linear encoder, optimizing the encoder with dropout regularization leads to an equiangular tight frame (ETF). Since this optimization is non-convex, we add another regularization that promotes such structures by minimizing the cross-correlation between filters in the network. We demonstrate its applicability in convolutional and fully connected layers in both feed-forward and recurrent networks. All these results suggest that there is indeed a relationship between dropout and ETF structure of the regularized linear operations.

1 Introduction

Deep neural networks are powerful computational models that have been used extensively for solving problems in computer vision, speech recognition, natural language processing, and many other areas [30, 32, 33, 53, 61]. The parameters of these architectures are learned from a given training set. Thus, regularization techniques for preventing overfitting of the data are very much required [22, 35]. Such methods include Batch Normalization [31], Weight decay [34], ℓ_1 regularization on the weights [48, 62] and Jacobian regularization [45, 50].

One of the most popular strategies is *Dropout*, which randomly drops hidden nodes along with their connections at training time [29, 51]. During training, in each batch, nodes are kept with a probability p, which causes them to be eliminated with probability $q = 1 - p$ (with their corresponding input and output weights). The weights of the remaining nodes are trained by back-propagation regularly. At inference time, the outputs of the layer(s) on which Dropout was applied are multiplied by p. Though very useful, Dropouts explicit regularization is not fully understood yet. Such an understanding is required to exploit the full potential of Dropout, and to deepen our knowledge in neural networks.

This work approaches Dropout from a signal processing and information theory perspective. It draws a connection between Dropout in a denoising autoencoder (DAE) and signal recovery from erasures in the analog domain (see Fig. 1). In this "analog coding" problem, a signal passes through an encoder A and then disrupted by an additive noise and part of its values are nullified. Once received, it is recovered by passing through a decoder B. The goal is to find the pair (A, B), which recovers the input signal with a minimal ℓ_2 error.

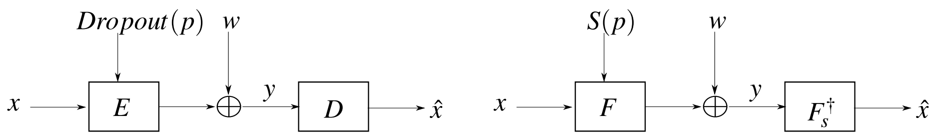

Figure 1: (left) a DAE variant (see [2]) with a linear encoder; (right) a signal encoding scheme in an analog channel with a decoder that performs least squares based inversion. $S(p)$ is a sampling pattern with a sampling ratio p and $w \in \mathbb{R}^{p \cdot n}$ is an additive noise.

.

To draw a connection to Dropout, we make the following steps. First, we examine a specific case, where the encoding A is performed by a (linear) matrix multiplication F, and the recovery is done by solving a least squares problem with the given measurements and F_s, the subset of columns from the matrix F corresponding to the kept measurements. It has been suggested in a recent work that frames with MANOVA distribution [17], minimize the expected ℓ_2 error in this setting [23]. Though not proven formally, various empirical measurements lead to the conjecture that ETFs have MANOVA distribution in their sub-matrices, and thus minimize the ℓ_2 error in the above setup [24, 25].

Next, we draw a relationship to DAE (briefly illustrated in Fig. 1). Considering an autoencoder with a linear encoder and a Dropout regularization applied on it, we get a very similar structure to the analog coding problem. Thus, if the decoder solves the least squares problem, then an ETF is likely to be a global minimum in the encoder optimization.

Last, we notice that the representation learned by autoencoders may be used for classification, e.g., in a semi-supervised learning setup, where the learned encoder serves as a feature extractor. This leads to the conjecture that promoting structure of an ETF in some layers of the network might turn useful for classification tasks as well. This provides a first step towards using frame theory for understanding and improving neural networks.

We support our claim by experiments done on various data-sets for image classification and word level prediction. We measure the effect of the ETF regularization when used as a sole regularizer, and when combined with Dropout. For fully connected (FC) layers, we promote an ETF structure for the weight matrix directly by reducing the correlation between its rows. We demonstrate this regularization for both feed-forward and recurrent (LSTM) networks. For convolutional layers, we do not use their corresponding Toeplitz matrix. Instead, for simplicity, the coherence between the convolution kernels is minimized.

2 Related works

This section discusses some previous works that analyze dropout. A detailed description of dropout and autoencoders appears in the sup. mat.

One disadvantage of Dropout, is that it slows down the training time. Wang and Manning, have implied that Dropout makes a Monte Carlo assessment of the layers output and thus reduced the training time [57]. Frazier-Logue and Hanson suggest that Dropout is just a special case of a stochastic delta rule, where each weight is parameterized as a random variable with a mean and variance of its own. Their method leads to faster convergence than using Dropout [18]. Hara et al. compared training with Dropout to ensemble learning, where several sub-networks are learned independently, and then the final result is an aggregation of all of them [27]. Baldi et al. [3, 41] introduced a general formalism for studying Dropout in networks with the sigmoid activation function. They showed that for a shallow network the expected output of a network with dropout can be approximated via the weighted geometric mean of the network outputs. Wager et al. analyzed Dropout applied to the logistic loss for generalized linear models (GLM) [54]. They claim that Dropout is similar to applying ℓ_2 regularization, where each squared weight is normalized using the Fisher information matrix.

Helmond and Lond derived a sufficient condition to guarantee a unique minimizer for a loss function that uses Dropout [28]. To differentiate between the bias induced by Dropout and ℓ_2 regularization, they provide examples for input data distributions for which the error achieved by Dropout is lower than the one of ℓ_2, and examples for the opposite case. Wager et al. showed for a generative Poisson topic model with long documents that Dropout training improves the exponent in the generalization bound for empirical risk minimization [55]. Cavazza et al. discussed the equivalence between Dropout and a fully deterministic model for Matrix Factorization in which the factors are regularized by the sum of products of the squared Euclidean norms of the columns of the matrix [9]. Pal et al. showed equivalence between Dropout and DropConnect [56], and that for single hidden-layer linear networks, DropBlock [21] induces spectral k-support norm regularization, and promotes solutions that are low-rank and have factors with equal norm [42]. Tang et al. proposed DisOut, a method for feature distortion based on the network layers empirical Rademacher complexity [52].

Gal and Ghahramani use Dropout to measure the uncertainty of a network. They approximate the likelihood functions with Monte Carlo sampling done via Dropout [19].

The two methods most related to our work are the one by Mianjy et al. [40] and DeCov [11] described in detail in the sup. mat. The first studies the implicit bias of Dropout [40]. It focuses on the case of a shallow autoencoder with a single hidden layer. It draws a relationship between the norms in the encoder and the decoder showing that they need to be equalized. The second by Cogswell et al. [11] uses the fact that Dropout leads to less correlated features and thus suggests to regularize the covariance of the features with respect to the training data. Hereafter, we compare our theory to the one of Mianjy et al. [40] and show that our proposed regularization method enforces jointly equalized matrices when performed in a linear autoencoder, and mention the connection between our work and Decov.

3 Signal reconstruction from a frame representation

We now address a notorious problem in information theory: Signal reconstruction from a frame representation with erasures, as illustrated in Fig. 1. Later on, we shall use its resemblance to autoencoders. Consider the signal vector $x \in \mathbb{R}^m$ and a frame F. First, the vector is encoded by F, i.e., yielding xF, which is then transmitted in an analog channel. In the channel, part of the values are nullified with probability p, and then the remaining values are disrupted by an additive white Gaussian noise (AWGN).

Notice that nullifying the values in xF with probability p is equivalent to removing columns from F with probability p and then multiplying it with x. Denote by $S(p)$ the pattern that defines which vectors of F are used, with respect to the probability p, and by F_s the sub-matrix of F with the vectors corresponding to $S(p)$. Then the resulted vector after the addition of the AWGN w is defined as

$$y = xF_s + w. \quad (1)$$

In order to recover the input from y, one may use the least square solution

$$\hat{x} = \arg\min_{\tilde{x}} \|y - \tilde{x}F_s\|_2^2 = yF_s^{\dagger}, \tag{2}$$

where $F_s^{\dagger}$ is the pseudo-inverse of F_s. Thus, if one wishes to optimize F for minimizing the reconstruction error in the ℓ_2 sense, the target objective is:

$$\arg\min_F \mathbb{E}\|x - \hat{x}\|_2^2 = \arg\min_F \mathbb{E}\left\|x - yF_s^{\dagger}\right\|_2^2, \tag{3}$$

where the expectation is with respect to the noise variable w, the distribution of the input variable x, and the sampling vector $S(p)$.

Frames for signal encoding. A number of works have studied the problem of reconstruction from erasures in the setup presented in Fig. 1 (see for example [4, 6, 7, 36]). As part of it, the usage of frames as encoders was vastly explored. Frames, or overcomplete bases, are $m \times n$ matrices with rank m, where $n > m$. They are widely used in various applications of communication, signal processing, and harmonic analysis [5, 8, 10, 26]. For example, they are often used for sampling techniques to analyze and digitize signals and images when they are represented as vectors or functions in a Hilbert space [16].

There is also a great interest in finding frames with favorable properties that hold for random subsets of their columns [46]. One popular type of frames is tight frames. A frame F of dimensions $m \times n$ is a tight frame iff $FF^T = c \cdot I_m$ for some constant c. In [12], they have been shown to be useful for quantization.

Equiangular tight frames. An interesting sub-group of tight frames are ETFs. The Gram matrix of a frame F is defined by $G_F = F^T F$ and contains outside its diagonal the cross-correlation values between the columns of the frame F, i.e., $G_{i,j}$ contains the cross-correlation value between the ith and jth columns of F. The Welch bound [58] provides a universal lower bound on the mean and maximal absolute value of the cross-correlations between the frame vectors. A frame that achieves the Welch lower bound on the maximal absolute cross-correlation value is an ETF. The Gram matrix G_{ETF} of a $m \times n$ ETF satisfies:

$$\left|(G_{ETF})_{i,j}\right| = \begin{cases} 1 & i = j \\ \frac{n-m}{(n-1)m} & else. \end{cases} \tag{4}$$

Intuitively, the n vectors of an ETF are spread uniformly across an m dimensional space with an angle $\theta = \arccos\sqrt{\frac{n-m}{(n-1)m}}$ between them.The maximal off-diagonal value in the Gram matrix is denoted the mutual coherence [13] or simply the coherence value.

It was demonstrated that frames that reach the Welch bound (also known as Equiangular Tight Frames(ETF)), have MANOVA distribution [24]. The eigenvalue distribution of the submatrices of an ETF is shown empirically to resemble the MANOVA distribution. We provide a brief intuition here and more details in the sup. mat. Note that minimizing the estimation error at the decoder output is equivalent to minimizing $\mathbb{E}[Tr(F_s^T F_s)^{-1}]$ because

$$\arg\min_F \mathbb{E}\|x - \hat{x}\|_2^2 = \arg\min_F \mathbb{E}\left\|x - F_s^{\dagger}Fx + F_s^{\dagger}w\right\|_2^2 = \arg\min_F \mathbb{E}\left\|F_s^{\dagger}w\right\|_2 \tag{5}$$
$$= \arg\min_F \mathbb{E}(Tr(F_s^{\dagger T}F_s^{\dagger}ww^T)) = \arg\min_F \sigma_w^2 \cdot \mathbb{E}(Tr(F_s^T F_s)^{-1}) = \arg\min_F \mathbb{E}(Tr(F_s^T F_s)^{-1}).$$

Assuming the frame columns are normalized and F_s has k columns, then $\mathbb{E}[Tr(F_s^T F_s)]$ is independent of S and equals to $= \sum_{i=1}^{k} \lambda_i$, where λ_k is the k^{th} eigenvalue of F. Thus, the minimization objective in Eq. (5) becomes $\mathbb{E}\left[\sum_{i=1}^{k} \frac{1}{\lambda_i}\right]$. In this case, it is clear that the best possible distribution is $p(\lambda) = \delta(\lambda)$, i.e., each sub-matrix is unitary. Yet, this is impossible to maintain for over-complete frames for which all the sub-matrices cannot be unitary.

To assess that Manova is the optimal choice, various popular random matrices with known distributions were tested [24]. These include Low pass frames with Vandermonde distribution [47], Gaussian frames that obeys the Marchenko-Pastur distribution [24], and frames whose distribution resembles the MANOVA distribution such as ETF, Random Fourier, and Haar [24]. MANOVA was shown to be the distribution closest to $\delta(\lambda)$. Thus, overall we get the conjecture that ETFs are the global minimum for the settings of Eq. (3).

4 An ETF perspective of Dropout

Having the problem of reconstruction of a signal with erasures stated, we turn to draw a relationship between it and optimizing a neural network with dropout. In particular, we focus mainly on the relationship to autoencoders.

4.1 The relationship between Dropout and ETF

Notice the great resemblance between a denoising autoencoder (DAE) with linear encoder and Dropout applied on it and the analog coding problem, as illustrated in Fig. 1. Though in standard DAE the noise is added at the input, in the DAE we present here, we put the noise at the output of the encoder, as suggested in [2]. There is a similarity between the two models in the linear case as adding noise at the output of the decoder is equivalent to adding noise at its input with a covariance matrix equal to the pseduo-inverse of the decoder.

Given the above information, in the case that the encoder is linear and the decoder calculates the least squares solution, we conjecture that the global minimum of training with Gaussian distributed data and noise, and Dropout on the encoder should be an ETF for the encoder (or very close to it if the setting slightly changes).

Notice that for a given Dropout/erasure pattern, the decoder is a linear operation. Since the encoder is also linear, the autoencoder with the fixed pattern becomes a shallow linear autoencoder as used in [40] (See sup. mat.). In that work it is claimed that Dropout induces the matrices of such a shallow linear autoencoder to be jointly equalized. In our case, the optimal encoder is claimed to be an ETF and thus the linear encoder and decoder in the linear autoencoder induced by the Dropout are a sub-matrix of an ETF and its pseudo-inverse, respectively. Interestingly, it turns out that this pair is indeed jointly equalized, which corresponds with the theory derived in [40]. Notice that this is not exactly the result derived in that work, since unlike their assumption that the decoder is linear, here it is non-linear (it is linear only given a specific erasure pattern). Thus, this relationship requires further study.

To examine the relationship between Dropout and ETFs, we set an experiment with an autoencoder that has a similar structure to the analog coding problem setup described in Section 3. The encoder A in this network is a linear one, represented by a randomly initialized matrix. Specifically, we use a matrix A of size 75×150.

For the decoder, we do not use $A_s^{\dagger}$ since it is hard to calculate its derivative with respect to A during training. Instead, we use the fact that the pseudo inverse is the least squares solution and perform ten iterations of gradient descent $\hat{x}^{i+1} = \hat{x}^i - \mu A_s^T(A_s\hat{x}^i - y)$, where $\hat{x} = \hat{x}^9$ and $x = \hat{x}^0$. The learning rate μ is the inverse of the largest eigenvalue of the Gram matrix $A_s^T A_s$ as in [37]. For the sample pattern $S(p)$, we simply apply Dropout on the encoder.

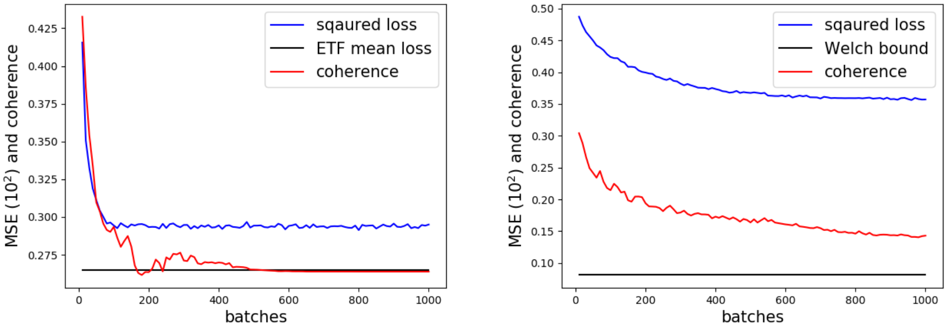

Figure 2: Training linear DAE with infinite data: plots of the coherence and the squared error as a function of batches. The error is scaled by 100 to fit with coherence in the same plot. Left: optimizing over the MSE. Right: optimizing over the Encoder coherence.

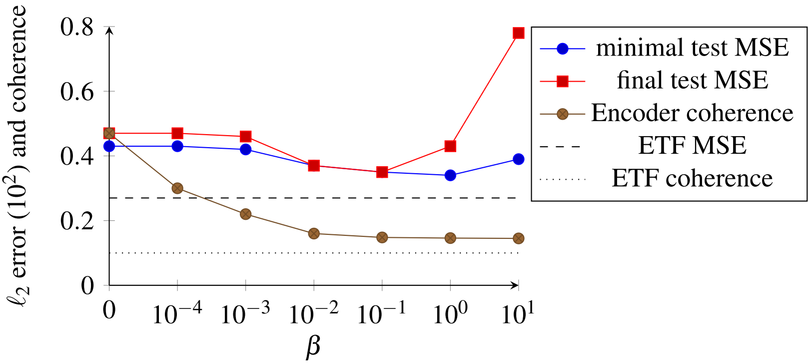

Figure 3: Training linear DAE with finite data: Plots of the squared error of a DAE and the coherence of its encoder as a function of β. The error is scaled by 100 to put it with the coherence in the same plot. For each β, the minimal and final MSE is measured and compared to the one of an ETF with a pseudo inverse decoder.

The input signals are generated as i.i.d. Gaussian vectors with a standard deviation of 1 and the noise is generated with the same distribution but with a standard deviation of 0.001.

The experiment is performed in two different settings: the first includes an infinite amount of data, and the second deals with a finite and limited one. In the infinite data case, we seek to find correspondence between the encoder coherence and the squared loss. We use an "online" learning setup with 100 signals per batch. First, we optimize over the squared loss and measure the coherence. Second, we optimize over the following "Coherence loss":

$$CL = \|A^T A - |G_{ETF}|\|_\infty, \tag{6}$$

where $|\cdot|$ is an element-wise absolute value, and $\|\cdot\|_\infty$ returns the maximum absolute value in the matrix. As can be seen in Fig. 2, the coherence and the reconstruction loss are closely related. Notice that coherence minimization induces a MSE reduction and vice versa. This validates our claim on the relationship between the two. Indeed, the coherence does not reach the Welch bound, and the error is much higher than the one of an ETF. We conjecture that this is mainly due to the non-convexity of the problem.

For limited data, regularization of the coherence is considered such that the new loss is

$$L = MSE + \beta \cdot CL, \tag{7}$$

where β is the regularization coefficient. Note that this term encourages getting an ETF-like structure. We train the autoencoder with this new regularization over several values of β. We use a training set of 100 signals, where the training phase includes randomizing the noise vector and the sampling pattern in each batch. The test set size is chosen as 5000 to accurately measure the test error. We train the model for 300 epochs in which the minimal and final test errors are measured, along with the final coherence.

It is known that regularization techniques increase the bias of a model. If successful, they reduce the models variance such that the total error is reduced, and thus prevent overfitting. Therefore, we expect low regularization coefficients to have little effect on the performance, large ones to perform poorly due to high bias, and for a specific range to increase the training error while decreasing the test error. Fig. 3 shows that until a certain value, both the error and the coherence diminish as β increases. High β values (in this case - higher than 0.1), result with optimization difficulties. This demonstrates that adding this term indeed helps in improving the convergence of the encoding frame closer to the desired ”global minimum”.

4.2 Promoting an ETF structure in general neural networks

Recalling the setting of Section 3, notice that the encoding part is exactly equivalent to a FC layer in a neural network, where the frame F plays the roll of the weight matrix, and the nullification with probability p acts as Dropout. Though the specific setup discussed here is more relevant to autoencoders, we believe that the new understandings of Dropout may be carried also to more general neural networks. Inspired by the usage of autoencoders for classification, we conjecture that ETFs may be helpful for regularization also in other tasks beyond signal recovery, e.g. for classification as we demonstrate hereafter.

Since there are infinitely many ETFs, we do not want to regularize a layer towards a specific one. Moreover, we do not always have an ETF construction for every combination of m and n. Yet, the structure of the Gram matrix is easily accessible and is the same for all ETFs that have the same value of m and n.

For these reasons, and the ones specified in Section 4.1, we adopt the "ETF similarity" term presented in Eq. (7) also for general neural networks and in particular for ones performing classification tasks. Notice that in the case where $m > n$, all vectors can be independent, and we penalize the distance from I_m, which is the same as reducing the magnitude of the off-diagonal entries of $A^T A$. In the case of convolution, we regularize the coherence between the convolution kernels (we justify this selection in the sup. mat.).

In an LSTM cell, we have four different FC gates: One to create a new state vector; one to create a $forget$ vector, which decides how much to keep from the old state; One for an $input$ vector, which decides how much to keep from the new state; and one for the cell's $output$. We promote ETF-like matrices on each one of them separately, since we do not want to impose low coherence between the vectors of the different FC layers (We may still want that the same filter will be used in the different gates.).

Interestingly, our proposed coherence based regularization technique may also be motivated by the sparse coding theory, where it is well known that it is easier to recover the sparse

Table 1: Comparison of ETF optimization criteria on LeNet5 FC layer and Fashion MNIST

Regularization	None	**ETF max (ℓ_∞)**	ETF sum (ℓ_1)	ETF squared (ℓ_2)
Test Accuracy	88.36%	**90.89%**	88.89%	89.22%

representation of a vector from a matrix that has a low coherence [13, 15]. In a recent work, it has been shown that the layers of a convolutional neural network may be viewed as stages for reconstructing the sparse representation of the input [44]. Moreover, recovery guarantees have been developed based on the coherence showing that a smaller coherence leads to better reconstruction of the sparse representation of the input by the network [43]. While that work focuses mainly on convolutional layers, it definitely provides another motivation for our new regularization technique. This is especially true since in classical sparse coding the coherence is also used with regular matrices (equivalent to the weights in the FC layers).

Practically, there are few ways to promote a matrix A to be an ETF-like, i.e., making its coherence as small as possible. We focus on three of them: minimizing the sum of squares of $\left|A^T A\right| - |G_{ETF}|$, the sum of absolute values and the maximal value, which is equivalent to minimizing the coherence of A as in (7). Notice that minimizing the sum of absolute values is similar to the approach used in [15] for minimizing the coherence in a dictionary by reducing the average absolute value of the cross-correlations between its columns. Another approach proposed in [14] relies on a spectral decomposition of A. Though it is shown to be more effective than the one in [15], it is too computationally demanding for using it with a neural network training and thus we focus only on techniques that minimize the coherence directly. In addition, by assuming that the inputs are Gaussian, it is possible to minimize the cross-correlations of the columns, which partially coincides with the DeCov method [11] that penalizes the activation cross correlations.

We compare the three regularization options above with a classification network for the Fashion MNIST dataset. We regularize the FC layer in a LeNet5 type network (the exact settings are detailed in Section 5). Table 1 presents the classification accuracy on the test set. We select for each regularization strategy its own optimal parameter β. This table suggests that minimizing the coherence directly, i.e. the maximal value (ℓ_∞) of $\left|A^T A\right| - |G_{ETF}|$ as appears in Eq. (7), should be the preferred option.

An intuition behind the usage of the ℓ_∞ norm in the optimization is related to the concept of hard example mining. The loss focuses on the two columns that cause the Gram matrix to be the farthest from the one of an ETF. It is known that in non-convex optimization, one may achieve improvement when focusing on the optimization of the harder examples, which improves the convergence and results [20, 38].

5 Experiments

We turn to evaluate our method apart of and on top of Dropout in the classification regime. Our analysis above applies to the regression (auto-encoder) case, and therefore it does not necessarily imply that in the classification case Dropout will encourage an ETF structure. Thus, we check here whether adding such a regularization can help also for classification. We emphasize that we do not try to compete with Dropout, nor we try to reach state-of-the-art results. In addition, the training time using the coherence is much higher than using Dropout.

Table 2: FC layer regularization effect on test accuracy

Regularization	Fashion MNIST	CIFAR-10	Tiny ImageNet (top1)	Tiny ImageNet (top5)
None	88.36%	84.41%	39.92%	65.29%
Dropout	90.16%	86.16%	48.35%	73.13%
ETF	90.89%	86.14%	44.21%	69.34%
Dropout+ETF	**91.91%**	**86.94%**	**49.78%**	**73.55%**

We simply aim at demonstrating the impact of ETF regularization with and without Dropout. The value of β is chosen by cross-validation. To isolate the effect of the two methods, no other regularization techniques are used. We demonstrate our proposed strategy on FC layers, convolutional layers, and LSTM. Four known datasets (Fashion MNIST, CIFAR-10, Tiny-ImageNet, Penn tree bank) are used with their appropriate architectures.

Fashion MNIST [59] is a dataset similar to MNIST but with fashion related classes that are harder to classify compared to the standard MNIST. We use for it a LeNet5 based model.

CIFAR-10 is composed of 10 classes of 32×32 RGB natural images with 50,000 training images, and 10,000 testing images. The architecture used is also based on a variant of Lenet5.

Tiny Imagenet is composed of 200 classes of natural images with 500 training examples per class, and 10,000 images for validation. Each image is an RGB image of size 64×64. It is tested by top-1 and top-5 accuracy. The architecture we use is an adaptation of the VGG-16 model [49] to the Tiny Imagenet dataset [1].

We perform word level prediction experiments on the Penn Tree Bank data set [39]. It consists of 929,000 training words, 73,000 validation words, and 82,000 test words. The vocabulary has 10,000 words. In this dataset, we measure the results by the attained perplexity, which we aim at reducing. The architecture used is as in [60]. Two models are considered, where all of them involve LSTMs with two-layer, which are unrolled for 35 steps. The *small* model includes 200 hidden units, and the *medium* includes 650.

The full implementation details appear in the sup. mat. and the code is available at: https://github.com/dorbank/An-ETF-view-of-Dropout-Regularization.

Fully connected layers. We start by applying our ETF regularization on the FC layers on three image classification datasets: Fashion MNIST , CIFAR-10 and Tiny ImageNet. As can be seen in Table 2, the ETF regularization improves the test accuracy, with and without Dropout. Note that it always improves the results of Dropout when combined together with it and that on the Fashion MNIST data it gets better performance also when it is used alone.

Convolutional layers. Next, we apply our ETF regularization on the convolutional layers (Table 3). It can be observed that the ETF regularization has less effect on the convolutional layers compared to the FC ones, both when applied with and without Dropout. We conjecture that for classification tasks, the kernels of the different channels have already lower coherence than the columns of a FC weight matrix. It might be also that a regularization of the coherence of the stride matrix may lead to better results.

LSTM. Lastly, we apply our ETF regularization on LSTM cells. We test it for both the small sized model and the medium sized one (Table 4). Notice that in this case, we also see the positive effect of the ETF regularization mainly when combined with Dropout. When applied alone, its effect is weaker in the medium model compared to the small one though it always leads to improvement. We believe that this difference should be further investigated.

Table 3: Convolution layer regularization effect on test accuracy

Regularization	Fashion MNIST	CIFAR-10	Tiny ImageNet (top1)	Tiny ImageNet (top5)
None	88.36%	84.41%	39.92%	65.29%
Dropout	91.14%	85.75%	43.44%	69.03%
ETF	90.30%	85.15%	42.13%	67.05%
Dropout+ETF	**91.58%**	**86.36%**	**45.55%**	**69.80%**

Table 4: LSTM layer regularization effect on test accuracy - Penn Tree Bank dataset

Regularization	small model (Val Perp.)	small model (Test Perp.)	medium model (Val Perp.)	medium model (Test Perp.)
None	121.39	115.91	123.012	122.853
Dropout	98.260	93.927	87.059	83.059
ETF	104.425	99.398	115.868	111.956
Dropout + ETF	**93.998**	**90.139**	**85.267**	**81.646**

6 Conclusions

This work provides a novel interpretation of the role of Dropout by bringing together two, similar but "unacquainted", research fields, namely, deep learning and frame theory. This combination provides the understanding that Dropout promotes an ETF structure when applied on a linear encoder in an autoencoder model. We have shown that adding a regularization that encourages an ETF structure improves the performance in these networks. The fact that in semi-supervised learning, the encoder also serves many times as a feature extractor for classification tasks, has led us to the usage of this ETF regularization also in standard neural networks, e.g., for classification, along with Dropout. This combination has shown improvement in different tasks and network types. It showed that the bias induced by the proposed regularization is related to the one of Dropout. We believe that the relationship that this work draws between the two (both theoretically and empirically) should be further explored.

It appears that the study of frames can help to gain a better understanding of the Dropout regularization. We believe that this paper makes the first steps in this direction by studying the optimal frame created by Dropout in an autoencoder architecture that has a linear encoder. The improvement demonstrated in this work by the ETF regularization together with Dropout, for various tasks such as classification, suggests that the role of ETF in neural network optimization should be more deeply analyzed in these contexts.

7 Acknowledgments

We thank Prof. Ram Zamir and Marina Haikin for fruitful discussion and for introducing us to the analog coding setup. This work is supported by the ERC-STG SPADE grant.

References

[1] VGG code for tiny imagenet, 2017. https://github.com/pat-coady/tiny_imagenet.

[2] A. Achille and S. Soatto. Information dropout: Learning optimal representations through noisy computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[3] Pierre Baldi and Peter J Sadowski. Understanding dropout. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2814–2822. Curran Associates, Inc., 2013.

[4] B. G. Bodmann and V. I. Paulsen. Frames, graphs and erasures. *Linear Algebra and its Applications*, 404:118 – 146, 2005.

[5] Bernhard G. Bodmann, Pete Casazza, and Radu Balan. Frames for linear reconstruction without phase. *The 42nd Annual Conference on Information Sciences and Systems*, pages 721–726, 2008.

[6] Pete G. Casazza and Gitta Kutyniok. Robustness of fusion frames under erasures of subspaces and of local frame vectors. *Contemporary Mathematics*, 25:114–132, 2008.

[7] Peter G. Casazza and Jelena Kovačević. Equal-norm tight frames with erasures. *Advances in Computational Mathematics*, 18(2):387–430, Feb 2003.

[8] Peter G Casazza, Gitta Kutyniok, and Shidong Li. Fusion frames and distributed processing. *Applied and Computational Harmonic Analysis*, 25:114–132, 2008.

[9] Jacopo Cavazza, Pietro Morerio, Benjamin Haeffele, Connor Lane, Vittorio Murino, and Rene Vidal. Dropout as a low-rank regularizer for matrix factorization. *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 84:435–444, 09–11 Apr 2018.

[10] Ole Christensen. An introduction to frames and riesz bases. *Birkhauser*, 2003.

[11] Michael Cogswell, Faruk Ahmed, Ross B. Girshick, Larry Zitnick, and Dhruv Batra. Reducing overfitting in deep networks by decorrelating representations. *CoRR*, 2015.

[12] Nicolae Cotfas and Jean Pierre Gazeau. Finite tight frames and some applications. *Journal of Physics A: Mathematical and Theoretical*, 43(19):193001, 2010. URL http://stacks.iop.org/1751-8121/43/i=19/a=193001.

[13] D.L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. *Proc. Nat. Aca. Sci.*, 100(5):2197–2202, March 2003.

[14] J. M. Duarte-Carvajalino and G. Sapiro. Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization. *IEEE Transactions on Image Processing*, 18(7):1395–1408, July 2009.

[15] M. Elad. Optimized projections for compressed sensing. *IEEE Transactions on Signal Processing*, 55(12):5695–5702, Dec 2007.

[16] Y. C. Eldar. *Sampling Theory: Beyond Bandlimited Systems*. Cambridge University Press, 2015.

[17] L. Erdős and B. Farrell. Local Eigenvalue Density for General MANOVA Matrices. *Journal of Statistical Physics*, 152:1003–1032, September 2013. doi: 10.1007/s10955-013-0807-8.

[18] Noah Frazier-Logue and Stephen José Hanson. Dropout is a special case of the stochastic delta rule: faster and more accurate deep learning. *CoRR*, abs/1808.03578, 2018.

[19] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 1050–1059, 2016.

[20] Weifeng Ge, Weilin Huang, Dengke Dong, and Matthew R. Scott. Deep metric learning with hierarchical triplet loss. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 272–288, Cham, 2018. Springer International Publishing.

[21] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le. Dropblock: A regularization method for convolutional networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 10750–10760, Red Hook, NY, USA, 2018. Curran Associates Inc.

[22] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[23] M. Haikin and R. Zamir. Analog coding of a source with erasures. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pages 2074–2078, July 2016.

[24] Marina Haikin, Ram Zamir, and Matan Gavish. Random subsets of structured deterministic frames have MANOVA spectra. *Proceedings of the National Academy of Sciences*, 114(26):E5024–E5033, 2017.

[25] Marina Haikin, Ram Zamir, and Matan Gavish. Frame moments and welch bound with erasures. *CoRR*, abs/1801.04548, 2018.

[26] Deguang Han and David R. Larson. Frames, bases and group representations. *American Mathematical Society,*, 697, 2000.

[27] Kazuyuki Hara, Daisuke Saitoh, and Hayaru Shouno. Analysis of dropout learning regarded as ensemble learning. *CoRR*, abs/1706.06859, 2017.

[28] David P. Helmbold and Philip M. Long. On the inductive bias of dropout. *Journal of Machine Learning Research*, 16:3403–3454, 2015.

[29] G. E. Hinton, N. Srivatava, A. Krizhvsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *NIPS*, 2012.

[30] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-Rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara Sainath, and Brian Kingbury. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[31] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 448–456. JMLR.org, 2015.

[32] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751, 2014.

[33] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances In Neural Information Processing Systems*, pages 1–9, 2012.

[34] Anders Krogh and John A. Hertz. A simple weight decay can improve generalization. In *Advances in Neural Information Processing Systems*, pages 950–957, 1992.

[35] J. Kukačka, V. Golkov, and D. Cremers. Regularization for deep learning: A taxonomy. In *ArXiv preprint*, 2017.

[36] David Larson and Sam Scholze. Signal reconstruction from frame and sampling erasures. *Journal of Fourier Analysis and Applications*, 21(5):1146–1167, Oct 2015.

[37] Yann LeCun, Patrice Y. Simard, and Barak Pearlmutter. Automatic learning rate maximization by on-line estimation of the hessian's eigenvectors. In S. J. Hanson, J. D. Cowan, and C. L. Giles, editors, *Advances in Neural Information Processing Systems 5*, pages 156–163. Morgan-Kaufmann, 1993.

[38] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017.

[39] M. P. Marcus, M. A. Marcinkiewicz, and Santorini B. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313—-330, 1993.

[40] Poorya Mianjy, Raman Arora, and Rene Vidal. On the implicit bias of dropout. In *ICML*, 2018.

[41] P. Sadowski P. Baldi. The dropout learning algorithm. *Artifical Intelligence*, 210:78–122, 2014.

[42] Ambar Pal, Connor Lane, René Vidal, and Benjamin D. Haeffele. On the regularization properties of structured dropout. *CVPR*, 2020.

[43] V. Papyan, Y. Romano, and M. Elad. Convolutional neural networks analyzed via convolutional sparse coding. *Journal of Machine Learning Research (JMLR)*, (18): 1–52, 2017.

[44] V. Papyan, Y. Romano, J. Sulam, and M. Elad. Theoretical foundations of deep learning via sparse representations: A multilayer sparse model and its connection to convolutional neural networks. *IEEE Signal Processing Magazine*, 35(4):72–89, July 2018. ISSN 1053-5888. doi: 10.1109/MSP.2018.2820224.

[45] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *International Conference on International Conference on Machine Learning (ICML)*, pages 833–840, 2011.

[46] M. Rupf and J. L. Massey. Optimum sequence multisets for synchronous code-division multiple-access channels. *IEEE Trans. Inf. Theor.*, 40(4):1261–1266, September 2006. ISSN 0018-9448. doi: 10.1109/18.335940. URL https://doi.org/10.1109/18.335940.

[47] Øyvind Ryan and Mérouane Debbah. Asymptotic behaviour of random vandermonde matrices with entries on the unit circle. volume 55, pages 3115–3148, 2009.

[48] Simone Scardapane, Danilo Comminiello, Amir Hussain, and Aurelio Uncini. Group sparse regularization for deep neural networks. *Neurocomputing*, 241:81 – 89, 2017. ISSN 0925-2312.

[49] Karen Simonian and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, 2015.

[50] J. Sokolic, R. Giryes, G. Sapiro, and M. R. D. Rodrigues. Robust large margin deep neural networks. *IEEE Transactions on Signal Processing*, 65(16):4265–4280, 2017.

[51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15, 2014. 1929–1958.

[52] Yehui Tang, Yunhe Wang, Yixing Xu, Boxin Shi, Chao Xu, Chunjing Xu, and Chang Xu. Beyond dropout: Feature map distortion to regularize deep neural networks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 5964–5971. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6057.

[53] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018.

[54] Stefan Wager, Sida Wang, and Percy S Liang. Dropout training as adaptive regularization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 351–359, 2013.

[55] Stefan Wager, William Fithian, Sida Wang, and Percy S Liang. Altitude training: Strong bounds for single-layer dropout. In *Advances in Neural Information Processing Systems (NIPS)*, pages 100–108, 2014.

[56] Li Wan, Matthew Zeiler, Sixin Zhang, Yann LeCun, and Rob Fergus. Regularization of neural networks using dropconnect. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1058–III–1066. JMLR.org, 2013.

[57] Sida Wang and Christopher Manning. Fast dropout training. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 118–126, 2013.

[58] L. Welch. Lower bounds on the maximum cross correlation of signals. *IEEE Transactions on Information theory*, 20, 1974.

[59] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

[60] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization, 2014. URL https://arxiv.org/abs/1409.2329.

[61] Ying Zhang, Mohammad Pezeshki, Philemon Brakel, Saizheng Zhang, César Laurent, Yoshua Bengio, and Aaron C Courville. Towards end-to-end speech recognition with deep convolutional neural networks. In *Interspeech*, pages 410–414, 2016.

[62] Yuchen Zhang, Jason D. Lee, and Michael I. Jordan. l1-regularized neural networks are improperly learnable in polynomial time. *CoRR*, abs/1510.03528, 2015.