Politehnica University of Bucharest
Faculty of Automatic Control and Computers
Department of Automatic Control and Systems Engineering

PhD Thesis Abstract

Adaptive Strategies for Separable Dictionary Learning and Applications to Anomaly Detection

Andra-Elena Băltoiu

Advisor:

Prof. dr. ing. Bogdan Dumitrescu

2021

1 Introduction

Dictionary Learning for sparse representations (DL) is a class of signal processing techniques that approximate signals by a linear combination of few elements of a basis, called the dictionary. The learning problem requires that both the dictionary and the sparse representation are learned from the data. The model is used for solving general signal reconstruction problems, for classification, various image-related tasks and in the field of compressed sensing. When it comes to 2D signals, such as images, however, the model requires that the data is vectorized, an operation that potentially breaks the correlations present in the second dimension. A solution to this drawback is the so called Separable Dictionary Learning (SDL) model, in which the dictionary is structured as the Kronecker product of two smaller (therefore more computationally efficient) dictionaries.

The primary focus of the thesis concerns the SDL model and, in particular, it deals with determining the optimal values of the two underlying heuristics: the sparsity of the representation and the dictionary size. Signal sparsity is the main assumption of the DL model and while it is rigorously proven only for some types of signals, practice has shown the suitability of such a representation in numerous (other) applications. The question of choosing the target sparsity of the model therefore remains open: there are limited possibilities of inferring the real sparsity of the data. The second heuristic is related to the number of elements in the dictionary (called the atoms), of which few are chosen to represent each signal. Numerical experiments [1] have shown that while larger dictionaries produce better signal approximations, this improvement levels out beyond a point and does not justify the additional computational demand.

In this work, we also adapt the SDL model to the problem of identifying graph anomalies. We are concerned with detecting abnormal topologies, i.e. (sub-)graph structures that stand out from the neighboring connectivity patterns. In such a case, the graph signals can be represented by their corresponding Laplacian matrices and we expect the dictionary atoms to describe elementary relational configurations. Our contribution also includes two other solutions for the anomaly detection problem, that take into consideration the requirements of most applications: the appropriateness of the method for unsupervised and online scenarios.

The dictionary learning problem is formalized as

$$\boldsymbol{Y} = \boldsymbol{D}\boldsymbol{X} + \boldsymbol{V}. \tag{1}$$

where $\boldsymbol{Y} \in \mathbb{R}^{m \times N}$ are the N signals, $\boldsymbol{D} \in \mathbb{R}^{m \times n}$ is the dictionary, having n atoms, $\boldsymbol{X} \in \mathbb{R}^{n \times N}$ is the representation matrix and $\boldsymbol{V} \in \mathbb{R}^{m \times N}$ represents the noise, which in most cases is considered to be Gaussian.

Both the dictionary and the representation are to be learned from the signals, leading to the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{D},\boldsymbol{X}} \ & \|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2 \\ s.t. \ & \|\boldsymbol{x}_l\|_0 \leq s, l = 1:N \\ & \|\boldsymbol{d}_j\|_2 = 1. \end{aligned} \tag{2}$$

Note that while $\boldsymbol{D}$ can also be designed in advance and kept fixed, we do not treat this case and instead consider only the case in which the dictionary is learned. Such an approach has the advantage of producing a better suited dictionary for approximating the signals. The above problem is usually solved via an alternate minimization scheme, in which $\boldsymbol{D}$ and $\boldsymbol{X}$ are iteratively refined until a certain condition is met. The procedure starts by initializing a random dictionary and computing the representation, a step called sparse coding. In the next iteration, $\boldsymbol{X}$ is kept fixed and the dictionary is updated. A common stopping condition is the number of learning iterations, as most algorithms for updating $\boldsymbol{D}$ are known to achieve good approximation performance after sufficient iterations.

In separable dictionary learning, the dictionary is formed as $\boldsymbol{D} = \boldsymbol{D}_2 \otimes \boldsymbol{D}_1$, with $\boldsymbol{D}_1 \in \mathbb{R}^{m_1 \times n_1}$ and $\boldsymbol{D}_2 \in \mathbb{R}^{m_2 \times n_2}$. The model becomes

$$\boldsymbol{Y} = \boldsymbol{D}_1 \boldsymbol{X} \boldsymbol{D}_2^\top + \boldsymbol{V} \tag{3}$$

and it is known to be equivalent to (1), since

$$\text{vec}(\boldsymbol{D}_1 \boldsymbol{X} \boldsymbol{D}_2^\top) = (\boldsymbol{D}_2 \otimes \boldsymbol{D}_1)\text{vec}(\boldsymbol{X}). \tag{4}$$

2 Sparsity Bayesian Learning for Separable DL

The first constraint in (2) assumes the target sparsity s is known in advance, which is rarely the case in practice. Performance is highly dependent on the choice of sparsity, especially in some classes of applications, such as compressed sensing. We show that misestimating sparsity leads to either overfitting or underfitting. The experiment is run on a synthetic dataset generated where the real sparsity level is known and evaluates several models (i.e. models having different target sparsity values) by assessing the ratio between train and test errors. Results show that understimating sparsity leads to similar test and train errors, suggesting improper model training, while overestimating s leads to overfitting.

Existing solutions alleviate these problems by either estimating s in a preliminary stage, by using a small set of signals to infer sparsity bounds, or employ adaptive strategies for inferring optimal sparsity as learning progresses. A third approach involves casting (2) as a Sparse Bayesian Learning (SBL) problem, which, in its hierarchical formulation, leads to a sparse solution without the need for explicitly setting a sparsifying prior. Consider the case where we are dealing with a single signal $\boldsymbol{y}$, instead of the entire signal matrix $\boldsymbol{Y}$, and its

corresponding representation $\boldsymbol{x}$. The basic, non-hierarchical, SBL approach considers the problem of learning $\boldsymbol{x}$ as

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{x}|\boldsymbol{y}) = \arg\max_{\boldsymbol{x}} p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}). \tag{5}$$

This formulation leads to the Maximum a Posteriori (MAP) solution, which can be improved if instead of working directly on the elements of $\boldsymbol{x}$, a hyperparameter $\boldsymbol{\gamma}$ controlling the variance of each element is introduced

$$\hat{\boldsymbol{\gamma}} = \arg\max_{\boldsymbol{\gamma}} p(\boldsymbol{\gamma}|\boldsymbol{y}, \sigma^2) = \arg\max_{\boldsymbol{\gamma}} p(\boldsymbol{y}|\boldsymbol{\gamma}, \sigma^2)p(\boldsymbol{\gamma}). \tag{6}$$

As [2] has shown, a non-informative prior $p(\boldsymbol{\gamma})$ results in a sparse solution, therefore knowledge on the real sparsity is no longer required. Optimizing for $\boldsymbol{\gamma}$ leads to an estimation of the (Gaussian) posterior $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\gamma}, \sigma^2)$, having mean $\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}_x\boldsymbol{D}^\top\boldsymbol{y}$ and variance

$$\boldsymbol{\Sigma}_x = \left(\sigma^2\boldsymbol{D}^\top\boldsymbol{D} + \boldsymbol{\Gamma}^{-1}\right)^{-1}. \tag{7}$$

where $\boldsymbol{\Gamma}$ is a diagonal matrix containing the hyperparameters. The covariance matrix above is computationally demanding, because of its dimension. An alternative formulation, in terms of the smaller $\boldsymbol{\Sigma}_y$ is given in [3]

$$\boldsymbol{\Sigma}_y = \sigma^2\boldsymbol{I} + \boldsymbol{D}\boldsymbol{\Gamma}\boldsymbol{D}^\top. \tag{8}$$

The mean now becomes

$$\boldsymbol{\mu} = \boldsymbol{\Gamma}\boldsymbol{D}^\top\boldsymbol{\Sigma}_y^{-1}\boldsymbol{y}. \tag{9}$$

Estimation of $\boldsymbol{\gamma}$ can be performed following different approaches [2, 3]. We use the Expectation-Maximization (EM) formulation (k denotes the current iteration number):

$$\text{E step: } \mathbb{E}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\gamma}^{(k)}}[x_i^2] = (\Sigma_x)_{i,i} + \mu_i^2, \tag{10}$$

$$\text{M step: } \gamma_i^{(k+1)} = \mathbb{E}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\gamma}^{(k)}}[x_i^2], \tag{11}$$

$$(\sigma^2)^{(k+1)} = \frac{1}{m}\left(\|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\mu}\|^2 + (\sigma^2)^{(k)}\sum_{i=1}^{n}\left(1 - \frac{\Sigma_{x_{i,i}}}{\gamma_i}\right)\right)^{1/2}. \tag{12}$$

Our contribution (SBL-2D) [4] adapts the SBL framework to the separable DL problem and proposes a two-stage approach. First, the representation support is learned in a computationally efficient way, by changing the way the hyperparameters control the representation elements. Since we are dealing with two dictionaries, an element of $\boldsymbol{X}$ is influenced by an atom from $\boldsymbol{D}_1$ (namely a column of the dictionary) and an atom from $\boldsymbol{D}_2$ (a row of $\boldsymbol{D}_2^\top$). We propose that the variance of the rows and columns of $\boldsymbol{X}$ are controlled independently by two hyperparameters, $\boldsymbol{\beta}^{(1)} \in \mathbb{R}^{n_1}$ for rows and $\boldsymbol{\beta}^{(2)} \in \mathbb{R}^{n_2}$ for columns.

To adapt relations (7-8), we note that the matrix $\boldsymbol{\Gamma}$ is replaced by two matrices $\boldsymbol{B}^{(d)}$, one for each SBL process. Each $\boldsymbol{B}^{(d)}$ can be expressed as a Kronecker product, denoted generically $\boldsymbol{B}^{(d)} = \boldsymbol{B}_2^{(d)} \otimes \boldsymbol{B}_1^{(d)}$, with the particular forms

$$\begin{aligned} \boldsymbol{B}^{(1)} &= \boldsymbol{I}_{n_2} \otimes \mathrm{diag}(\boldsymbol{\beta}^{(1)}), \\ \boldsymbol{B}^{(2)} &= \mathrm{diag}(\boldsymbol{\beta}^{(2)}) \otimes \boldsymbol{I}_{n_1}. \end{aligned} \tag{13}$$

Here $\boldsymbol{I}_{n_d}$ denotes the identity matrix of size n_d; the first above relation simply says that all elements of row i of $\boldsymbol{X}$ are associated with $\beta_i^{(1)}$. We introduce the notation $\boldsymbol{\beta}^{(d)}$, $d \in \{1, 2\}$ for referencing both dimensions and extend it to all other variables that have both row and column values.

Using known Kronecker product properties, adapting (8) for the 2D case reads

$$\boldsymbol{\Sigma}_y^{(d)} = (\sigma^{(d)})^2 \boldsymbol{I} + (\boldsymbol{D}_2 \boldsymbol{B}_2^{(d)} \boldsymbol{D}_2^\top) \otimes (\boldsymbol{D}_1 \boldsymbol{B}_1^{(d)} \boldsymbol{D}_1^\top). \tag{14}$$

Structured covariances arise in several multivariate contexts. Consequently, computing the inverse of covariances such as (14) is known [5] to be efficiently obtained using the SVD decomposition, in our case

$$\begin{aligned} \boldsymbol{D}_1 \boldsymbol{B}_1^{(d)} \boldsymbol{D}_1^\top &= \boldsymbol{U}_1^{(d)} \boldsymbol{S}_1^{(d)} \boldsymbol{U}_1^{(d)\top}, \\ \boldsymbol{D}_2 \boldsymbol{B}_2^{(d)} \boldsymbol{D}_2^\top &= \boldsymbol{U}_2^{(d)} \boldsymbol{S}_2^{(d)} \boldsymbol{U}_2^{(d)\top}. \end{aligned} \tag{15}$$

By simply using the orthogonality of the matrices from the above relations, it results that

$$\left(\boldsymbol{\Sigma}_y^{(d)}\right)^{-1} = (\boldsymbol{U}_2^{(d)} \otimes \boldsymbol{U}_1^{(d)}) \left((\sigma^{(d)})^2 \boldsymbol{I} + \boldsymbol{S}_2^{(d)} \otimes \boldsymbol{S}_1^{(d)}\right)^{-1} (\boldsymbol{U}_2^{(d)} \otimes \boldsymbol{U}_1^{(d)})^\top, \tag{16}$$

where now the matrix to be inverted is diagonal. We denote

$$\left((\sigma^{(d)})^2 \boldsymbol{I} + \boldsymbol{S}_2^{(d)} \otimes \boldsymbol{S}_1^{(d)}\right)^{-1} = \mathrm{diag}(\boldsymbol{t}^{(d)}),$$

with $\boldsymbol{t}^{(d)} \in \mathbb{R}^m$. Using the above expression and denoting

$$\boldsymbol{E}_1^{(d)} = \boldsymbol{B}_1^{(d)} \boldsymbol{D}_1^\top \boldsymbol{U}_1^{(d)}, \quad \boldsymbol{E}_2^{(d)} = \boldsymbol{B}_2^{(d)} \boldsymbol{D}_2^\top \boldsymbol{U}_2^{(d)}, \tag{17}$$

the solution estimation (9) becomes

$$\boldsymbol{\mu}^{(d)} = (\boldsymbol{E}_2^{(d)} \otimes \boldsymbol{E}_1^{(d)}) \cdot \mathrm{diag}(\boldsymbol{t}^{(d)}) \cdot (\boldsymbol{U}_2^{(d)} \otimes \boldsymbol{U}_1^{(d)})^\top \boldsymbol{y}. \tag{18}$$

The E step is completed by transforming the expression of the posterior covariances (7) into

$$\boldsymbol{\Sigma}_x^{(d)} = \boldsymbol{B}_2^{(d)} \otimes \boldsymbol{B}_1^{(d)} - (\boldsymbol{E}_2^{(d)} \otimes \boldsymbol{E}_1^{(d)}) \cdot \mathrm{diag}(\boldsymbol{t}^{(d)}) \cdot (\boldsymbol{E}_2^{(d)} \otimes \boldsymbol{E}_1^{(d)})^\top. \tag{19}$$

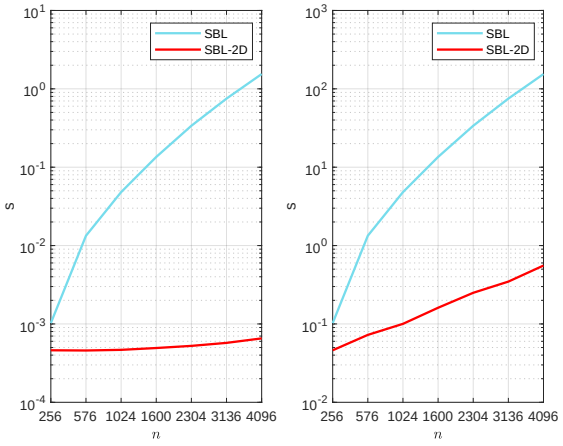

Figure 1: Execution time (in seconds) for SBL and SBL-2D. Left: time per iteration. Right: total time.

Until now, we have presented the ingredients for solving the E step of the EM algorithm. The M step amounts to updating the hyperparameter values and noise estimation. We adapt the step to our separable problem as

$$\beta_i^{(d)} \leftarrow \frac{1}{n_{3-d}} \sum_{\substack{\ell=1 \\ c_d(\ell)=i}}^{n} \left((\Sigma_x)_{\ell,\ell}^{(d)} + (\mu_\ell^{(d)})^2 \right). \tag{20}$$

Finally, the noise estimation (12) is performed separately for the row and column cases with

$$(\sigma^{(d)})_{(k+1)}^2 = \frac{1}{m_1 m_2} \left(\|\boldsymbol{y} - (\boldsymbol{D}_2 \otimes \boldsymbol{D}_1)\boldsymbol{\mu}^{(d)}\|^2 + (\sigma^{(d)})_{(k)}^2 \sum_{\ell=1}^{n} \left(1 - \frac{(\Sigma_x)_{\ell,\ell}^{(d)}}{\beta_{c_d(\ell)}^{(d)}} \right) \right)^{1/2}. \tag{21}$$

With identifying the representation support in this manner, we complete the first stage of our SBL-2D algorithm. The second step involves using the regular SBL method on the reduced support in order to estimate the representation. The approach improves the computational costs, since the problem is now considerably smaller. Figure 1 shows execution times for both algorithms. Testing is done using a synthetic dataset, where the real sparsity is $s = 5$ and signals are corrupted by noise having signal-noise ratio SNR $= 40$.

Results show that accuracy is comparable to regular SBL and often considerably higher than other sparse representation solutions adapted for the 2D case. We test performance in a set of experiments on synthetic data as well as real images. We vary the real sparsity level and noise values and measure the signal approximation error (RMSE) and representation error (by comparing the

true representation with the SBL-2D estimate). The contribution also includes convergence tests - showing the reduction of the representation support at each iteration, the evolution of the two error metrics, as well as false negative and false positive counts for the identified support.

3 Dictionary Size Adaptation for Separable DL

When sparsity is unknown and not estimated by one of the strategies mentioned above, it is usually set with respect to dictionary size. However, while at least for some types of signals, an informed choice for sparsity is possible, in what the dimension of the dictionary is concerned, such prior information hardly exists. In turn, common practice is to set the size of $\boldsymbol{D}$ with respect to signal dimension. The alternative is to apply adaptive strategies for determining the optimal size.

Regardless of the optimality criterion used, all strategies involve starting with either a small dictionary or a large one and adding and/or trimming atoms as learning progresses, in order to obtain a better suited model (i.e. dictionary). More precisely, size optimality is evaluated once every few iterations and the dimension of the dictionary is modified accordingly. However, the size adaptation strategy can interfere with learning, since newly added atoms require some iterations to align to meaningful directions. In an experiment on synthetic data, we investigate this effect by evaluating signal approximation error at each model evaluation step and the corresponding optimal size, which we in turn compare to the true dictionary size.

One solution for evaluating the model is the Minimum Description Length (MDL) principle of model selection, which assumes the correlations present in the signals allow for a parsimonious representation. As such, an optimal model is one that can compress the data, while not hindering on signal approximation. The framework includes several Information-theoretic Criteria (ITC) that seek a compromise between model performance and model complexity. In particular, the Renormalized Maximum Likelihood (RNML) criterion is known to permit a formulation (called Extended-RNML) for the problem of determining the optimal size of the dictionary [6].

Our contribution involves the adaptation of ERNML to the separable model and corresponding size adjustment scheme [7]. Because of the equivalence of the original DL model (1) and the separable counterpart (3), the ERNML criterion does not require any essential reformulation in order to fit the separable case. The only necessary adaptations concern dimension adjustments and, given we are working with two dictionaries instead of one, parameter recount. In particular, signal dimension is $T_{2D} = m_1 m_2 N$ and the number of parameters is $\mathrm{NoP}_{2D} = sN + (m_1 - 1)n_1 + (m_2 - 1)n_2$, since we are counting the degrees of freedom for each dictionary separately. Adapted to the 2D case, the criterion reads

$$\begin{aligned}\mathrm{ERNML}_{2D} = (T_{2D} - \mathrm{NoP}_{2D}) \log \frac{\mathrm{RMSE}^2}{T_{2D} - \mathrm{NoP}_{2D}} \\ + \mathrm{NoP}_{2D} \log \frac{\|\boldsymbol{D}_1 \boldsymbol{X} \boldsymbol{D}_2^\top\|_F^2}{T_{2D} \cdot \mathrm{NoP}_{2D}} \\ + \log\left[\mathrm{NoP}_{2D}(T_{2D} - \mathrm{NoP}_{2D})\right] + 2N \log \binom{n_1 n_2}{s}.\end{aligned} \tag{22}$$

Adapting dictionary size via ITC in [6] involves routinely evaluating the model dimension-performance trade-off for a number n_{cand} of candidate models, more precisely for different sized dictionaries. Following Occam's razor, the strategy is biased towards examining smaller dictionaries, which also has the benefit of keeping the complexity of computing the criterion in check. It does, however, account for cases where the optimal size is larger. The idea is, briefly, to test whether the current dictionary is not over-sized with useless atoms. Accordingly, every $iter_{adapt}$ iterations, the (current) dictionary atoms are ordered based on their representation power. The power of an atom j is defined as [1]

$$P_j = \|\boldsymbol{x}_j^\top\|_2^2. \tag{23}$$

The candidate models are formed by excluding the least powerful atoms, the smallest candidate having $n_{current} - n_{cand} + 1$ atoms (where, clearly, $n_{current}$ is the size of the dictionary at the present iteration). A representation is computed for each model and the corresponding ERNML criteria are evaluated. The best dictionary is that for which ERNML is smallest. The case where the current dictionary is the one yielding the lowest value is interpreted as an indication that the model is not complex enough and hence atom addition is performed.

When working with separable dictionaries, however, we refer to the representation power of an atom combination $\boldsymbol{D}_{1,i}$ and $\boldsymbol{D}_{2,j}$. More precisely, the measure now takes the form

$$P_{2D} = \sum_{k=1}^{N} X_{i,j,k}^2. \tag{24}$$

We therefore sort the atom combinations based on power and produce the corresponding ranked atom lists for D_1 and D_2 respectively. There are now n_{cand}^2 candidates, since at most n_{cand} atoms can be excluded from each dictionary. The criterion is then computed for every pair combination of dictionary dimensions. Size adjustment is done separately on the two dictionaries, by performing the necessary atom additions/removals that are indicated by the lowest ERNML value. The resulting best $\boldsymbol{D}_d$s may have different sizes, which lays the ground for a well adapted model, since the signal patterns in one dimension (say the rows of $\boldsymbol{y}$) may differ in size and complexity from the ones in the other (columns of $\boldsymbol{y}$), thus requiring a different number of atoms for their representation.

We also propose a solution that minimizes interference of dimension modification with learning, which involves relying on ERNML_{2D} only as an indicator of the change. The actual optimal size is computed by smoothing the optimal size values provided by the ITC criterion over the last ws iterations, in order to avoid abrupt size changes that may lead to under-trained atoms. The solution uses a moving average filter of window ws. The complete size adaptation scheme is presented in Algorithm 1.

Results show good performance in determining the true dictionary sizes for several dimension configurations, different sparsity levels and noise values. Most encouragingly, dictionary size is hardly underestimated, especially when the true size is small.

4 Anomaly Detection

The anomaly detection (AD) problem can be seen as binary classification, where one class represents the normal signals, while the second the anomalies. Our contributions use dictionary learning to solve the AD problem for three applications: malware identification, financial fraud detection and the more general problem of graph anomaly detection.

Malware detection is essentially a large scale problem, given the multitude of software applications. The computational costs involved in learning such a large set of signals can be alleviated by resorting to online algorithms that can be successively trained on parts of the original dataset, without significantly compromising on accuracy. Another aspect concerns the dynamics of the field, where new types of malware are constantly produced, to evade the advances in anti-virus solutions. Models must accommodate this adaptability, which translates into the ability of identifying malware for which no previous example is available. Unsupervised methods are, thus, preferable.

The majority of the above observations also hold for the problem of detecting financial frauds, perhaps even more so. Financial fraud poses an additional problem for machine learning. A transaction is best described as a link between two nodes - the financial entities. Therefore, a suited data representation is that of a graph, which has the advantage of the representing inter-dependencies of transactions. Usually, both nodes and edges are attributed with information such as concerning identity, transaction amount and currency. Therefore, data consists in both numeric and relational information. This underlying graph is especially relevant with money laundering, where individual transactions in the scheme may in themselves seem legitimate; it is only in the context of the graph which links them that fraud becomes apparent.

Motivated by the problem of identifying money-laundering schemes and other financial frauds, our work [8] reviews anomaly detection methods that are aimed at graph data.

The general DL framework can be extended for the classification task, especially in the supervised setting. The goal is to learn a dictionary such that different sets of atoms describe samples from each class. Moreover, it is also

Algorithm 1: ITC-ADL-2D

Data:
signals, $\boldsymbol{Y} \in \mathbb{R}^{m \times N}$
sparsity, s
number of DL iterations, K
iterations step for size adaptation, K_{adapt}
number of candidate sizes, n_{cand}
minimum number of dictionary atoms n_{min}
moving average window, ws

Result: optimum-sized dictionaries, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$

Initialize dictionaries $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ of sizes n_d
for $k = 1 : K_{adapt} : K$ **do**
    Perform K_{adapt} iterations of 2D DL and obtain updated dictionary, $\boldsymbol{D}_1$, $\boldsymbol{D}_2$ and representation, $\boldsymbol{X}$ with s target sparsity
    Order atoms separately in $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ according to power
    Compute ERNML_{2D} criterion for all n_{cand}^2 candidates using (22) and return sizes $n_{d,ITC}$ that yield smallest ERNML_{2D} value
    Compute $n_{d,opt}$ as moving average of $n_{d,ITC}$ over a ws window
    Apply $\boldsymbol{D}_d =$ `AdjustSize(`$\boldsymbol{D}_d$, $n_{d,opt}$, n_d`)`, where $d \in \{1, 2\}$ for both $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$
Perform K_{adapt} more DL iterations with $n_{d,opt}$ sized dictionaries
Function $\boldsymbol{D}_d =$ `AdjustSize(`$\boldsymbol{D}_d$, $n_{d,opt}$, n_d`)`
    if $n_{d,opt} - n_d > 0$ **then**
        Set current adjusted size $n_d = n_{d,opt}$
        Add $n_{d,opt} - n_d$ atoms to dictionary $\boldsymbol{D}_d$
    else if $n_{d,opt} - n_d < 0$ **then**
        Set current adjusted size $n_d = \min(n_{d,opt}, n_{min})$
        Trim $\boldsymbol{D}_d$ to its most used n_d atoms
    else
        Set current adjusted size $n_d = n_{d,opt} + 1$
        Add 1 atom to dictionary $\boldsymbol{D}_d$

beneficial that the representations of the signals in each class differ significantly from the ones in other classes. These two constitute additional constraints imposed on the learning problem and can be achieved by forcing $\boldsymbol{X}$ to represent class labels and respect class atom allocation, along with the regular signal approximation objective.

Adding the three objectives together: signal reconstruction, discrimination (between classes) and label consistency (atom-class allocation), results in the following optimization problem

$$\min_{\boldsymbol{D},\boldsymbol{W},\boldsymbol{A},\boldsymbol{X}} \|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2+\alpha\|\boldsymbol{H}-\boldsymbol{W}\boldsymbol{X}\|_F^2+\beta\|\boldsymbol{Q}-\boldsymbol{A}\boldsymbol{X}\|_F^2. \tag{25}$$

where $\boldsymbol{H}\in\mathbb{R}^{c\times N}$ denotes the label matrix, $\boldsymbol{W}\in\mathbb{R}^{c\times n}$ the classifier matrix, $\boldsymbol{Q}\in\mathbb{R}^{n\times N}$ the atom allocation matrix, $\boldsymbol{A}\in\mathbb{R}^{n\times n}$ is a dictionary that imposes label consistency on the representation, and c stands for the number of classes.

The solution is called Label Consistent K-SVD (LC-KSVD) [9] since it can be recast as a regular DL problem and solved by standard algorithms such as K-SVD. Specifically, (25) is equivalent to

$$\min_{\boldsymbol{D},\boldsymbol{W},\boldsymbol{A},\boldsymbol{X}} \left\| \begin{bmatrix} \boldsymbol{Y} \\ \sqrt{\alpha}\boldsymbol{H} \\ \sqrt{\beta}\boldsymbol{Q} \end{bmatrix} - \begin{bmatrix} \boldsymbol{D} \\ \sqrt{\alpha}\boldsymbol{W} \\ \sqrt{\beta}\boldsymbol{A} \end{bmatrix} \boldsymbol{X} \right\|_F^2. \tag{26}$$

Online DL approaches to anomaly detection use several heuristics to control the confidence of classification. While this has indeed a positive impact on accuracy, it is not suited for the two applications we described. Both financial frauds and malware entail a high degree of novelty: it is to be expected that new fraudulent schemes are developed as old ones are exposed by anti-money laundering efforts and new types of malware are created as antivirus programs catch existing ones.

We propose a semi-supervised approach (called Tolerant Online Discriminative DL with Regularization - TODDLeR), which provides a solution to this compromise. The setting involves an offline pre-training stage, where a dictionary is learned using a small labeled sample set. Regular classification methods for DL can be performed for this task, such as LC-KSVD described above. The dictionary is used to initialize the unsupervised online stage, where $\boldsymbol{D}$ is updated by feeding the bulk of the dataset one sample at a time.

The following solution and results constitute our contribution published in [10]. The method is based on an existing method [11], but in order to make the solution better suited for malware detection, where unseen malware types need to inform the model for future use, we aim at using all signals to train the model. Improper labeling can misguide learning, therefore we wish to prevent signals from drastically modifying the current model, since there is no guarantee the change is for the best. The solution is to add regularization terms for the classifier and label consistency matrix that control their rate of change. Modified in this way, the label consistent classification problem (26) becomes

$$\begin{aligned} \min_{\boldsymbol{D},\boldsymbol{W},\boldsymbol{A}} &\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{x}\|_2^2+\alpha\|\boldsymbol{h}-\boldsymbol{W}\boldsymbol{x}\|_2^2+\beta\|\boldsymbol{q}-\boldsymbol{A}\boldsymbol{x}\|_2^2 \\ &+\lambda_1\|\boldsymbol{W}-\boldsymbol{W}_0\|_F^2+\lambda_2\|\boldsymbol{A}-\boldsymbol{A}_0\|_F^2. \end{aligned} \tag{27}$$

In the formulation above we have also adapted the online approach: instead of dealing with the entire signal matrix $\boldsymbol{Y}$, the objective is minimized with every incoming signal $\boldsymbol{y}$, that has a corresponding label vector $\boldsymbol{h}$ and an associated atom allocation vector $\boldsymbol{q}$. By $\boldsymbol{W}_0$ and $\boldsymbol{A}_0$ we denote the current values of the dictionaries, learned based on the previous signals.

The updated values of $\boldsymbol{W}$ and $\boldsymbol{A}$ can be obtained by fixing everything else in (25) and solving the corresponding minimization objectives

$$f(\boldsymbol{W}) = \|\boldsymbol{h} - \boldsymbol{W}\boldsymbol{x}\|_2^2 + \lambda_1 \|\boldsymbol{W} - \boldsymbol{W}_0\|_F^2 , \tag{28}$$

$$g(\boldsymbol{A}) = \|\boldsymbol{q} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda_2 \|\boldsymbol{A} - \boldsymbol{A}_0\|_F^2 . \tag{29}$$

Setting the gradients of the above functions to 0 leads to the following least squares solutions

$$\boldsymbol{W} = (\boldsymbol{h}\boldsymbol{x}^T + \lambda_1 \boldsymbol{W}_0)(\boldsymbol{x}\boldsymbol{x}^T + \lambda_1 \boldsymbol{I})^{-1}, \tag{30}$$

$$\boldsymbol{A} = (\boldsymbol{q}\boldsymbol{x}^T + \lambda_2 \boldsymbol{A}_0)(\boldsymbol{x}\boldsymbol{x}^T + \lambda_2 \boldsymbol{I})^{-1}. \tag{31}$$

We test the solution on two malware datasets and a financial fraud dataset. Results show comparable or improved classification accuracy over other online DL methods for classification.

Our second contribution to AD is an unsupervised method that progressively filters out the normal signals [12]. It is based on the assumption that normal samples, which generally outnumber the anomalies, are better represented by the model, and we thus used an error criteria to differentiate between the two. The solution also takes into consideration the fact that, as filtering leaves less regular signals to train on, this imbalance attenuates. We propose a composite dictionary structure, in which the new models from each iteration are combined with existing ones, since overtraining the dictionary on anomalies would corrupt the error criterion.

We also experiment with using atom properties to inform the class labels. An atom can be characterized in terms of the number of signals which are represented using the atom, a property that is called usefulness, $U_j = \|\boldsymbol{x}_j^\top\|_0$. Another approach to progressively filtering signals involves the restriction of the set $\mathcal{A}$ of potential anomalies to signals represented by atoms $\boldsymbol{d}_j$ with usefulness $U_j < N_a$, namely

$$\mathcal{A} = \{\, \boldsymbol{y}_k \mid x_{j,k} \neq 0 \wedge U_j < N_a, \forall j, k \,\} . \tag{32}$$

In other words, we expect atoms being used by less than N_a signals to represent the features of anomalies.

Both methods yield good results in lowering the false positive count, while keeping the false negatives in check and can thus be used for obtaining a smaller dataset with better class balance, since only normal signals are eliminated by filtering, if proper stopping conditions are set.

We also employ atom usefulness, together with the atom power measure in a multiscale dictionary setup to investigate the separability of normal and abnormal signals. Intermediate results show that using the standard DL formulation, without any added classification constraints, can still be informative on class labels if we use atom properties in order to characterize the signals. The experiments show that the distributions of atom usefulness and atom power

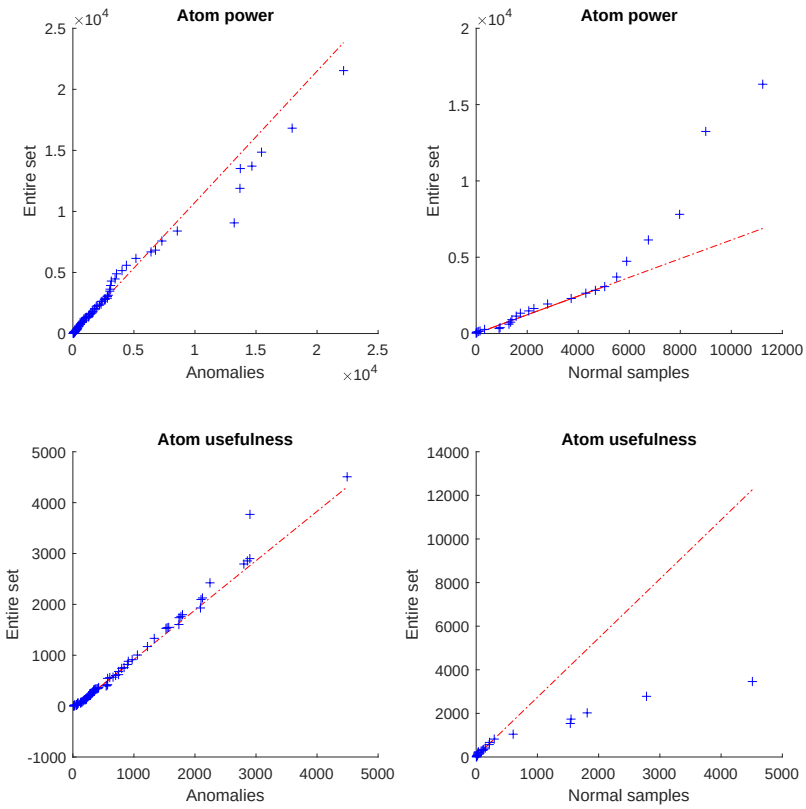

Figure 2: Quantile-Quantile plot of atom power and usefulness distributions in the normal and anomalies classes

differ significantly in anomalies from the normal samples. Figure 2 shows the quantile-quantile plot illustrating this comparison.

Our last contribution, called Separable Laplacian Classification [13] involves exploiting the structural information of graphs in order to identify anomalous patterns. We work directly with the structure of the graphs and adapt the separable dictionary learning problem, which takes into account vicinity patterns in 2D data. Our strategy is to exploit the two dimensional structure of a graph Laplacian in order to learn connectivity patterns that are specific to each class of graphs. The solution is tested on a synthetic dataset consisting of graph signals, in which the anomalies are structurally different from the normal samples. The investigated patterns are rings and cliques, since they are common in fraud-detection tasks. The training signals from each class are used separately to train one pair of dictionaries. Classifying a new, test signal, accounts to evaluating which pair of dictionaries is better at representing the signal. We also perform a test that investigates the effect of dictionary size on the ability of the Separable Laplacian Classification method in identifying the circular graph patterns.

References

[1] B. Dumitrescu, P. Irofti, Dictionary Learning Algorithms and Applications, Springer International Publishing, 2018 (2018).

[2] M. Tipping, Sparse Bayesian Learning and the Relevance Vector Machine, Journal of Machine Learning Research 1 (2001) 211–244 (2001).

[3] D. P. Wipf, B. D. Rao, Sparse Bayesian learning for basis selection, IEEE Transactions on Signal Processing 52 (8) (2004) 2153–2164 (2004).

[4] A. Băltoiu, B. Dumitrescu, Sparse bayesian learning algorithm for separable dictionaries, Digital Signal Processing 111 (2021) 102990 (2021).

[5] O. Stegle, C. Lippert, J. Mooij, N. Lawrence, K. Borgwardt, Efficient inference in matrix-variate Gaussian models with iid observation noise, in: Proceedings of the 24th Neural Information Processing Systems Conference, 2011, pp. 630 – 638 (2011).

[6] B. Dumitrescu, C. D. Giurcăneanu, Adaptive-Size Dictionary Learning Using Information Theoretic Criteria, unpublished document (2019).

[7] A. Baltoiu, B. Dumitrescu, Size adaptation of separable dictionary learning with information-theoretic criteria, in: 2019 22nd International Conference on Control Systems and Computer Science (CSCS), 2019, pp. 7–11 (2019).

[8] P. Irofti, A. Băltoiu, A. Pătrașcu, Fraud detection in networks, in: Enabling AI applications in Data Science, Springer, 2020, pp. 517–536 (2020).

[9] Z. Jiang, Z. Lin, L. Davis, Learning A Discriminative Dictionary for Sparse Coding via Label Consistent K-SVD, in: IEEE Conf. Computer Vision and Pattern Recognition, 2011, pp. 1697–1704 (2011).

[10] P. Irofti, A. Băltoiu, Malware identification with dictionary learning, in: 27th European Signal Processing Conference, 2019, pp. 1–5 (2019).

[11] S. Matiz, K. Barner, Label consistent recursive least squares dictionary learning for image classification, in: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 1888–1892 (2016).

[12] P. Irofti, A. Băltoiu, Unsupervised dictionary learning for anomaly detection, in: International Traveling Workshop on Interactions Between Sparse Models and Technology, 2020, pp. 1–3 (2020). arXiv:2003.00293.

[13] A. Băltoiu, A. Pătrașcu, P. Irofti, Graph anomaly detection using dictionary learning, in: The 21st World Congress of the International Federation of Automatic Control, 2020, pp. 3551–3558 (2020).