K-means Recovers ICA Filters when Independent Components are Sparse

Alon Vinnikov ALON.VINNIKOV@MAIL.HUJI.AC.IL
Shai Shalev-Shwartz SHAIS@CS.HUJI.AC.IL
School of Computer Science and Engineering, The Hebrew University of Jerusalem, ISRAEL

Abstract

Unsupervised feature learning is the task of using unlabeled examples for building a representation of objects as vectors. This task has been extensively studied in recent years, mainly in the context of unsupervised pre-training of neural networks. Recently, Coates et al. (2011) conducted extensive experiments, comparing the accuracy of a linear classifier that has been trained using features learnt by several unsupervised feature learning methods. Surprisingly, the best performing method was the simplest feature learning approach that was based on applying the K-means clustering algorithm after a whitening of the data. The goal of this work is to shed light on the success of K-means with whitening for the task of unsupervised feature learning. Our main result is a close connection between K-means and ICA (Independent Component Analysis). Specifically, we show that K-means and similar clustering algorithms can be used to recover the ICA mixing matrix or its inverse, the ICA filters. It is well known that the independent components found by ICA form useful features for classification (Le et al., 2012; 2011; 2010), hence the connection between K-mean and ICA explains the empirical success of K-means as a feature learner. Moreover, our analysis underscores the significance of the whitening operation, as was also observed in the experiments reported in Coates et al. (2011). Finally, our analysis leads to a better initialization of K-means for the task of feature learning.

1. Introduction

Many deep learning algorithms attempt to learn multiple layers of representations in an unsupervised manner. These representations are commonly used as features for classification tasks. Particularly, it was demonstrated that sparse features, i.e. features which are rarely activated, perform well in object recognition tasks.

Several algorithms have been proposed for learning sparse features. Some examples are: sparse auto-encoders, sparse coding, Restricted Boltzmann Machines and Independent Component Analysis (ICA). In particular, several variants of ICA have been shown to achieve highly competitive or state-of-the-art results for object classification (Le et al., 2011; 2010).

In computer vision applications, learning features is often interpreted as learning dictionaries of “visual words”, that are later being used for construction of higher level image features. While some works learn visual words by one of the aforementioned feature learning methods, the most widely used approach in the computer vision literature is to employ the vanilla K-means clustering as a method for obtaining such dictionaries (Wang et al., 2010; Csurka et al., 2004; Lazebnik et al., 2006; Winn et al., 2005; Fei-Fei & Perona., 2005).

Recently, Coates et al. (2011) considered various feature learning algorithms as part of a single-layer unsupervised feature learning framework. They applied K-means, sparse auto-encoders, and restricted Boltzmann machines. Surprisingly, the simple K-means prevailed over the more complicated algorithms, achieving state-of-the-art results. Two particular observations are of interest. One is that whitening plays a crucial role in classification performance when using K-means. Another is that when whitening is applied to the input, K-means learns centroids that resemble the oriented edge patterns which are typically recovered by ICA (Bell & Sejnowski, 1997).

Coates & Ng (2012) have observed empirically that K-means tends to discover sparse projections and have raised the question of whether this is accidental or there is a deeper relation to sparse decomposition methods such as ICA. In this work, we draw a connection between ICA and K-means, showing that when K-means is applied after whitening then, under certain conditions, it is able to

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32. Copyright 2014 by the author(s).

recover both the filters and the mixing matrix of the more expressive ICA model. This is despite the fact that the original goal of K-means is to attach a single centroid to each example. In addition, our analysis suggests a family of clustering algorithms with the same ability, and a simple way to empirically test whether an algorithm belongs to this family. Finally, our analysis reveals the importance of whitening, and leads to a new way to apply K-means for feature learning.

Based on these insights, we give an interpretation of the features learned by the framework in Coates et al. (2011). In general, the discovered properties of K-means suggest that, when applying K-means to computer vision tasks such as classification or denoising, it may be beneficial to incorporate whitening, as was done in the experiments presented in Coates et al. (2011).

2. Background and Basic Definitions

In this paper we draw a connection between ICA and K-means. We first define these two learning methods.

2.1. K-means

The K-means objective is defined as follows. We are given a sample $S = \{x^{(i)}\}_{i=1}^{m} \subseteq \mathbb{R}^d$, and a number of clusters $k \in \mathbb{N}$, and our goal is to find centroids $c = \{c^{(1)}, \dots, c^{(k)}\} \subseteq \mathbb{R}^d$, which are a global minimum of the objective function:

$$\hat{J}_S(c) = \frac{1}{m}\sum_{i=1}^{m} \min_{j\in[k]} \left\| x^{(i)} - c^{(j)} \right\|_2^2 . \tag{1}$$

Let A_k denote the algorithm which given S and k outputs a (global) minimizer of Equation (1). While it is intractable to implement A_k in the general case, and one usually employs some heuristic search (such as Lloyd's algorithm), here we will focus on the ideal K-means algorithm which finds a global minimum of Equation (1).

In addition, we will refer to the density based version of K-means defined as a minimizer of

$$J_x(c) = \mathop{\mathbb{E}}_{x} \min_{j\in[k]} \left\| x - c^{(j)} \right\|_2^2 , \tag{2}$$

where x is some random vector with a distribution over $\mathbb{R}^d$. We denote by $\mu = \{\mu^{(1)}, \dots, \mu^{(k)}\} \subseteq \mathbb{R}^d$ a minimizer of $J_x(c)$.

2.2. Independent Component Analysis (ICA)

The linear noiseless ICA model is a generative model (Hyvarinen & Oja, 2000), defining a distribution over a random vector $x = (x_1, \dots, x_d)^\top$. To generate an instance of x we should first generate a hidden random vector $s = (s_1, \dots, s_d)^\top$, where each s_k is a statistically independent component, distributed according to some prior distribution over $\mathbb{R}$. Then, we set

$$x = As \tag{3}$$

where $A \in \mathbb{R}^{d,d}$ is some deterministic matrix, often called *the mixing* matrix. We assume that A is invertible. A specific ICA model is parameterized by the mixing matrix A and by the prior distribution over s_k. Throughout this paper we mostly focus on the prior distribution being a Laplace distribution, that is, the density function is $p(s_k) \propto \exp(-\sqrt{2}|s_k|)$. We denote by s_{lap} the random vector over $\mathbb{R}^d$ whose components are i.i.d zero-mean unit-variance Laplace random variables. That is, $p(s) \propto \exp(-\sqrt{2}\|s\|_1)$, with $\|s\|_1$ being the ℓ_1 norm.

Given a sample $x^{(1)}, ..., x^{(N)}$ of N i.i.d. instantiations of the random vector x, the task of ICA is to estimate both the mixing matrix A and the sources (i.e., the hidden vectors) $s^{(1)}, ..., s^{(N)}$. Since the mixing matrix is invertible, once we know A we can easily compute the sources by $s = A^{-1}x$. From now on, we will refer to this model and task simply as ICA. We denote $W = A^{-1}$. The rows of W are commonly referred to as *filters*.

Since we can always scale the columns of A, we can assume w.l.o.g. that s_k have unit variance $\mathbb{E}[s_k^2] = 1$. In addition, we can assume, w.l.o.g., that s_k has zero mean, $\mathbb{E}[s_k] = 0$, since otherwise, we can subtract the mean of x by a simple preprocessing operation. Such preprocessing is often called *centering*.

Another preprocessing, which is often performed before ICA, is called *whitening*. This preprocessing linearly transforms the random variable x into $y = Tx$ such that y has identity covariance, namely, $E\{yy^\top\} = I$. Concretely, if $UDU^\top = E\{xx^\top\}$ is the spectral decomposition of the covariance matrix of x, then one way to obtain whitened data is by $y = D^{-1/2}U^\top x$. Another way, called ZCA whitening (Bell & Sejnowski, 1997), is $y = UD^{-1/2}U^\top x$.

The utility of whitening resides in the fact that the new mixing matrix is orthogonal, which reduces the number of parameters to be estimated. Instead of having to estimate n^2 parameters for the original matrix A, we only need to estimate the new orthogonal matrix that contains $n(n-1)/2$ degrees of freedom. A review of approaches for estimating the ICA model can be found in (Hyvarinen & Oja, 2000).

2.3. Additional Notation

Let $\|.\|_p$ denote the p-norm, and let the set of numbers $\{1, \dots, .k\}$ be denoted by $[k]$. e_i will represent the i-th unit vector in $\mathbb{R}^d$. Given $c = \{c_1, \dots, c_k\} \subseteq \mathbb{R}^d$ and $H \in \mathbb{R}^{d,d}$, for purposes of brevity we define $H * c \triangleq \{Hc_1, \dots, Hc_k\}$.

If $x^{(1)}, x^{(2)}, \ldots$ is an infinite sequence of i.i.d copies of a random vector x, $S_m = \left\{x^{(i)}\right\}_{i=1}^m$ is a sequence of random sets sharing a common sample space. We will simply say $S_m = \left\{x^{(i)}\right\}_{i=1}^m$ is an i.i.d sample of random variable x to refer to this sequence. We will extend the notion of almost sure convergence to sets of random vectors. A sequence of sets of random vectors $C_n = \{c_1^n, \ldots, c_k^n\}$ is said to converge almost surely to a set of fixed vectors $B = \{b_1, \ldots, b_k\}$ if there exists a labeling $c_{n1}^n, \ldots, c_{nk}^n$ of the points in C_n such that $c_{ni} \to b_i$ almost surely. We will denote this relation by $C_n \to B$ a.s.

3. K-means and ICA relationship

Before stating the main results, we first rewrite both the K-means and ICA objectives in terms of a matrix A and sources $s^{(1)}, \ldots, s^{(m)}$, and discuss the differences in the objectives.

In ICA we wish to estimate the unknown mixing matrix A, or equivalently its inverse W. Given m i.i.d samples $\left\{x^{(i)}\right\}_{i=1}^m \subseteq \mathbb{R}^d$ of ICA random vector x (see Equation (3)), a popular approach for estimating W is maximum likelihood estimation (Hyvarinen et al., 2001). The log-likelihood maximization takes the form:

$$\operatorname*{argmax}_W \sum_{i=1}^m \sum_{j=1}^d \log(p(w_j^\top x^{(i)})) + m \log|\det W|$$

where p is the prior density of the independent components s_j. In the context of natural images, sparsity is dominant (Hyvarinen et al., 2009), therefore p is often chosen to be the Laplace prior, which yields the L_1 penalty, $-\log(p(s_j)) = |s_j|$. Another popular prior distribution is the Cauchy prior which yields the penalty $-\log(p(s_j)) = \log(1 + s_j^2)$.

Equivalently, we can write the ICA problem as

$$\operatorname*{argmax}_{A, s^{(1)}, \ldots, s^{(m)}} \sum_{i=1}^m \sum_{j=1}^d \log(p(s_j^{(i)})) - m \log|\det A|$$
$$\text{s.t. } \forall i,\ x^{(i)} = As^{(i)}$$

For comparison, consider a simple reformulation of the K-means objective. Given $S = \left\{x^{(i)}\right\}_{i=1}^m \subseteq \mathbb{R}^d$, $k \in \mathbb{N}$, we can rewrite the objective given in (1) as

$$\operatorname*{argmin}_{A, s^{(1)}, \ldots, s^{(m)}} \frac{1}{m} \sum_{i=1}^m \left\| x^{(i)} - As^{(i)} \right\|_2^2$$
$$\text{s.t. } \forall i,\ s^{(i)} \in \{e_1, \ldots, e_k\}$$

Thus, both K-means and ICA can be viewed as a dictionary learning problem, seeking a matrix A and sources $s^{(1)}, \ldots, s^{(m)}$, that best explain the input set. K-means attempts to choose A so as to explain every sample with a *single* column from it. ICA attempts to find a perfect explanation of the input set, but allows each sample to depend on a *combination* of the columns of A. Therefore, in general, these objectives seem to be quite different, and there is no guarantee that the two optimization problems will recover the same A.

Nevertheless, in the next sections we will see that if the independent components come from a sparse distribution (e.g. Laplace or Cauchy), K-means and ICA recovers the very same mixing matrix A.

4. Main results

In this section we state our main results, showing conditions under which both K-means and ICA recovers the same mixing matrix A. Throughout this section, we consider the ICA task restricted to the case in which the prior distribution over the independent components is the Laplace distribution, possibly the most common prior in the context of sparsity. Extending the results to other sparse distributions remains to future work. In the experiments section we mention a variety of distributions that behave similarly to Laplace in practice.

We begin with a result regarding general clustering algorithms, beyond K-means. We define a family of clustering algorithms that satisfy two particular properties: Rotation Invariant and Sparse Sensitivity (RISS). We call this family the family of "RISS" clustering algorithms. We prove that any "RISS" clustering algorithm can be used to solve the ICA task. We then claim that the ideal K-means algorithm is "RISS". This work makes the first steps towards a complete proof. For ICA in two dimensions we prove that a close variant of K-means is indeed "RISS", and we provide experiments that support the claim for larger dimensions.

Furthermore, our analysis relies on the following two ideal assumptions: we can obtain the exact whitening matrix T for the ICA random variable x (Equation (3)), and we have access to the ideal algorithm for K-means or its variants. In practice, T can be estimated using a procedure similar to PCA and the K-means objective can be minimized to a local minimum using the standard Lloyd's algorithm. In the experiments section we show that even in the non-ideal setting our results tend to hold. We also discuss implementation details of our algorithm and derive from our theory an initialization technique for K-means that gives better results in practice.

4.1. "RISS" clustering algorithms

A clustering algorithm for the purpose of our discussion is any algorithm that receives a set $S = \left\{x^{(i)}\right\}_{i=1}^m \subseteq \mathbb{R}^d$ as

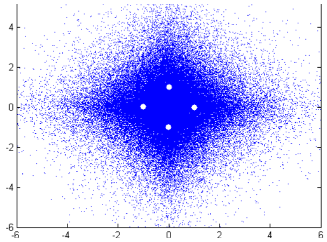


Figure 1. Illustration of property 2 of a "RISS" algorithm. (Blue) i.i.d samples of s_{lap} in two dimensions (White) output of standard K-means with k=4 converges to points that lie on the axes and hence to the unit vectors when normalized

input and outputs a set whose size is twice the dimension, $C = \left\{c^{(i)}\right\}_{i=1}^{2d} \subseteq \mathbb{R}^d$. We will use the term "RISS" clustering algorithm, for a clustering algorithm that satisfies the following two properties: First, a rotation of the input set should cause the same rotation of the output centroids. This is a reasonable assumption for any distance based clustering algorithm (such as K-means), since a rotation of all the examples does not change the distances between examples. The second property we require deals with the centroids the clustering algorithm finds when its input is a large enough sample of s_{lap}. We require that the output would be a set of centroids that lie near the axes, which is where the bulk of the Laplace distribution is. See an illustration in Figure 1. For the purpose of our paper, we are only interested in the direction of the centroids and are not interested in their magnitude. Therefore, we will always assume that the output of the clustering algorithm is normalized to have a unit Euclidean norm. Formally:

Definition 1. *A clustering algorithm B is "RISS" if it satisfies two properties.*

1. *For every input $S = \left\{x^{(i)}\right\}_{i=1}^{m} \subseteq \mathbb{R}^d$ and every orthonormal matrix $U \in \mathbb{R}^{d,d}$, $B(U * S) = U * B(S)$.*
2. *Let $S_m = \left\{s^{(i)}\right\}_{i=1}^{m} \subseteq \mathbb{R}^d$ be an i.i.d sample of s_{lap}, then $b_m = B(S_m)$ is a sequence of random sets such that $b_m \rightarrow \{e_1, ..., e_d, -e_1, ..., -e_d\}$ a.s.*

We now present the `Cluster-ICA` algorithm that employs a "RISS" clustering algorithm and prove it solves the ICA task, that is, it recovers the mixing matrix and filters of ICA.

Algorithm 1 Cluster-ICA

```
1: Input: i.i.d sample of ICA random variable $x$ (Equation (3)) $S = \left\{x^{(i)}\right\}_{i=1}^{m} \subseteq \mathbb{R}^d$
2: Obtain whitening matrix $T$ for $x$
3: Apply whitening to input $y^{(i)} = Tx^{(i)}$
4: Set $\{c^{(1)}, \ldots, c^{(2d)}\} = \text{ClusterAlgorithm}(\{y^{(i)}\}_{i=1}^{m})$
5: Output: $\{T^\top c^{(1)}, \ldots, T^\top c^{(2d)}\}$
```

Before proving correctness of the proposed algorithm, we need the following lemma adapted from (Hyvarinen & Oja, 2000).

Lemma 1. *Suppose x is an ICA random variable with independent components s, T is a whitening matrix for x and let $y = Tx$, then $y = Us$ where U is orthonormal.*

Proof. Note that $y = Tx = TAs$. Denoting $TA = U$ we have $I = \mathbb{E}\{yy^\top\} = U\mathbb{E}\{ss^\top\}U^\top = UU^\top$ where the first equality follows from definition of T and the last equality is true since we assume w.l.o.g $\mathbb{E}\{s_k^2\} = 1$. □

Theorem 1. *Let $S_m = \left\{x^{(i)}\right\}_{i=1}^{m}$ be an i.i.d sample of ICA random variable x with $s = s_{\text{lap}}$. Then given a "RISS" clustering algorithm and S_m, the output of `Cluster-ICA` converges a.s. to $\{w_1, \ldots, w_d, -w_1, \ldots, -w_d\}$, where $w_i^\top$ are the rows of $W = A^{-1}$.*

Proof. From lemma 1, after whitening we have $y^{(i)} = Us^{(i)}$, for some orthonormal matrix U. Therefore $y^{(i)}$ is an i.i.d sample of ICA random variable $y = Us$ for some orthonormal matrix U. Let B be the "RISS" clustering algorithm used by `Cluster-ICA`. We show below that the properties of B can be used to recover U.

First, from property 2,

$$B(\{s^{(i)}\}_{i=1}^{m}) \rightarrow \{e_1, ..., e_d, -e_1, ..., -e_d\}\ a.s.$$

In addition, from property 1,

$$B(\{y^{(i)}\}_{i=1}^{m}) = U * B(\{s^{(i)}\}_{i=1}^{m})\ .$$

Therefore

$$B(\{y^{(i)}\}_{i=1}^{m}) \rightarrow \{U_1, \ldots, U_d, -U_1, \ldots, -U_d\}\ a.s.$$

That is, after step 4 we have a set $\{c^{(1)}, \ldots, c^{(2d)}\}$ that converges to the columns of U and their negatives. Finally since we have U, it is easy to recover A^{-1} since $U = TA$, and therefore $A^{-1} = U^\top T$. It follows that $W^\top = (A^{-1})^\top = T^\top U$ and so

$$\{T^\top c^{(1)}, \ldots, T^\top c^{(2d)}\} \rightarrow \{w_1, \ldots, w_d, -w_1, \ldots, -w_d\}\ a.s.$$

□

Similarly, it can be shown that if we change the output of `Cluster-ICA` to $\{T^{-1}c^{(1)}, \ldots, T^{-1}c^{(2d)}\}$, then it converges to the columns of the mixing matrix A and their negatives.

Two conclusions of practical interest arise from the above analysis. Firstly, there are many ways to perform whitening in step 2 of the algorithm. In computer vision tasks, a common method is ZCA whitening which tends to preserve the appearance of image patches as much as possible. According to Theorem 1, All methods are equally valid in our context. Secondly, when we wish to empirically test whether some clustering algorithm is "RISS", if we can prove property 1 in definition 1, then property 2 is easy to validate by performing the same experiment as in section 6.1.

4.2. K-means is "RISS"

We conjecture that K-means with $k = 2d$ is "RISS"[1]. In section 6 we present experiments supporting this conjecture. In this section we make small modifications to the standard K-means objective (Equation (1)) that make the analysis easier. Specifically, we introduce a set of constraints to the objective that are, as evidenced in the experiments, already satisfied by K-means when applied to ICA data. We then prove this variant is "RISS" in the two-dimensional case and hope the proof sheds some light as to why the same might also be true for standard K-means and for any dimension.

First, we change the distance metric in Equation (1) from ℓ_2 norm to cosine distance, which is equivalent to constraining the centroids to the unit ℓ_2 sphere. Next, we leave d centroids free while constraining the other d to be their negatives. We then get

$$\underset{\substack{\forall i,\ \|c^{(i)}\|_2=1 \\ \forall i>d,\ c^{(i)}=-c^{(i-d)}}}{\operatorname{argmin}} \quad \frac{1}{m}\sum_{i=1}^{m} \min_{j\in[2d]} \left\|x^{(i)} - c^{(j)}\right\|_2^2$$

Or equivalently, denoting

$$\hat{J}_S^{cos}(c) = \frac{1}{m}\sum_{i=1}^{m} \max_{j\in[2d]} \langle x^{(i)}, c^{(j)}\rangle$$

we have

$$\underset{\substack{\forall i,\ \|c^{(i)}\|_2=1 \\ \forall i>d,\ c^{(i)}=-c^{(i-d)}}}{\operatorname{argmax}} \quad \hat{J}_S^{cos}(c) \tag{4}$$

In the equality we used the facts that $\|c^{(i)}\|_2 = 1$ and $\min(f(x)) = -\max(-f(x))$. Let A_{2d}^{cos} denote an ideal algorithm which given a sample $S = \left\{x^{(i)}\right\}_{i=1}^{m} \subseteq \mathbb{R}^d$, returns a set of centroids $c = \{c^{(1)}, \ldots, c^{(2d)}\} \subseteq \mathbb{R}^d$, which solve Equation (4).

Theorem 2. *A_{2d}^{cos} satisfies property 1 of a "RISS" clustering algorithm*

Proof. The proof follows directly from the invariance of the inner product to orthonormal projection, that is $\langle x^{(i)}, c^{(j)}\rangle = \langle Ux^{(i)}, Uc^{(j)}\rangle$ for any orthonormal U. □

The following theorem establishes the second property.

Theorem 3. *For $d = 2$, A_{2d}^{cos} satisfies property 2 of a "RISS" clustering algorithm.*

5. Proof sketch of Theorem 3

We now describe the main lemmas we rely upon. Their proofs are deferred to the appendix. Recall that in section 2 we defined the density based version of K-means. We adapt Equation (2) to our variant as well and define

$$J_x^{cos}(c) = \mathop{\mathbb{E}}_{x} \max_{j\in[2d]} \langle x, c^{(j)}\rangle$$

where x is some random variable with distribution $p(x)$ over $\mathbb{R}^d$. In this section we denote by $\mu = \{\mu^{(1)}, \ldots, \mu^{(2d)}\} \subseteq \mathbb{R}^d$ an optimal solution to the density-based counterpart of Equation (4)

$$\underset{\substack{\forall i,\ \|c^{(i)}\|_2=1 \\ \forall i>d,\ c^{(i)}=-c^{(i-d)}}}{\operatorname{argmax}} \quad J_x^{cos}(c)\ . \tag{5}$$

We first characterize μ:

Lemma 2. *For $d = 2$ and $x = s_{\text{lap}}$, $\mu = \{e_1, e_2, -e_1, -e_2\}$ is the unique maximizer of Equation (5)*

An important step in the proof of the above lemma is simplifying the objective $J_{s_{\text{lap}}}^{cos}(c)$ by the following.

Lemma 3. *Let u_{lap} be a random variable with uniform density over the ℓ_1 sphere, that is, all points satisfying $\|x\|_1 = 1$ have equal measure and the rest zero. Then, for any feasible set C we have*

$$\operatorname*{argmax}_{c\in C} J_{s_{\text{lap}}}^{cos}(c) = \operatorname*{argmax}_{c\in C} J_{u_{\text{lap}}}^{cos}(c).$$

Thus, Lemma 3 allows us to rewrite Equation (5) w.r.t u_{lap} instead of s_{lap}. The statement in Lemma 2 now becomes more obvious and easy to illustrate:

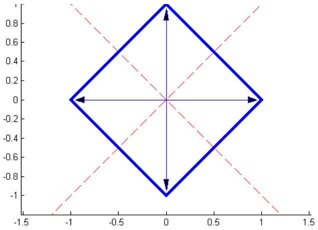


Figure 2. $\mu = \{e_1, e_2, -e_1, -e_2\}$. Blue arrows mark $\mu^{(i)}$, solid blue lines are the support of u_{lap}, and dotted red lines are the boundaries between the clusters $V_i^{\mu} = \{x \in \mathbb{R}^2 : \operatorname*{argmax}_{j\in[4]}\langle x, \mu^{(j)}\rangle = i\}$. Intuitively, on the ℓ_1 sphere, points near the axes have maximal length. Therefore, to maximize the expected inner product it is most beneficial to choose $\mu^{(i)}$ on the axes as well.

[1] Given that we normalize the resulting centroids to unit Euclidean norm.

Before continuing we mention that Lemma 2 is the only point in our analysis that is restricted to $d = 2$. The extension to higher dimensional spaces, $d > 2$, relies on proving the following conjecture, which is derived by setting $x = u_{\text{lap}}$ in Equation (5) and using the fact $\max\{x, -x\} = |x|$.

Conjecture 1.

$$\underset{\substack{c^{(1)},\dots,c^{(d)} \\ \|c^{(i)}\|_2=1}}{\operatorname{argmax}} \underset{u_{\text{lap}}}{\mathbb{E}} \{\max_{j\in[d]} |\langle u_{\text{lap}}, c^{(j)}\rangle|\} = \{e_1, \dots, e_d\}$$

Now that we have characterized the optimizer for the density based objective, let us prove convergence for a finite sample.

Lemma 4. *Let $S_m = \left\{s^{(i)}\right\}_{i=1}^{m} \subseteq \mathbb{R}^d$ be an i.i.d. sample of s_{lap}. If μ is a unique maximizer of Equation (5) where $x = s_{\text{lap}}$, then $A_{2d}^{cos}(S_m) \to \mu$ a.s.*

The proof of Theorem 3 now follows directly from Lemma 2 and Lemma 4.

6. Experiments

In this section we present experiments to support our results in the non-ideal setting. In appendix B we describe the non-ideal versions of A_k and A_k^{cos}, that is, the standard K-means algorithm and the algorithm for the variant presented in Equation (4) (referred to as the cosine-K-means). The K-means algorithm, also known as Llyod's algorithm, can be viewed as attempting to solve Equation (1) by alternating between optimizing for assignments of data points while keeping centroids fixed, and vice versa. The cosine-K-means algorithm is derived in the same manner for Equation (4).

6.1. K-means is "RISS"

We now perform an experiment showing that both K-means and cosine-K-means tend to satisfy the definition of a "RISS" clustering algorithm when the number of requested clusters k is twice the dimension.

Property 1 of definition 1 can easily be verified for both algorithms similarly to lemma 2. We therefore focus on property 2.

The experiment is as follows. We sample a number of d-dimensional vectors with each entry randomized i.i.d according to some distribution D. Then we run K-means and cosine-K-means with our sample and $2d$ randomly initialized centroids as input, resulting in centroids $c = \{c^{(1)}, \dots, c^{(2d)}\}$. For the output of standard K-means we normalize $c^{(i)}$ to have unit norm. Finally, we measure the distance of c to the set $e = \{e_1, ..., e_d, -e_1, ..., -e_d\}$ by matching pairs of vectors from c and e, taking their differences ϵ_j and reporting the largest $\|\epsilon_j\|_\infty$. More precisely, $\text{dist}(c, e) = \max_j \|e_j - c_{n_j}\|_\infty$ where $n_1, \dots, n_{2d}$ is the matching permutation. Note that if $\text{dist}(c, e) = 0$, we have $c = e$. Table 1 shows the resulting distances for various sample sizes and dimensions with D set to Laplace distribution, meaning the input is a sample of s_{lap}. Indeed, distances tend to zero as sample size increases, and so c converges to e per coordinate.

	$\lvert S\rvert = 10^4$	$\lvert S\rvert = 10^5$	$\lvert S\rvert = 5 \times 10^6$
d=2	0.0306, 0.0131	0.0063, 0.0033	0.0023, 0.00058
d=10	0.0908, 0.0495	0.0190, 0.0148	0.0032, 0.0024
d=20	0.3849, 0.0749	0.0367, 0.0238	0.0044, 0.0033
d=50	0.6124, 0.3748	0.2466, 0.1722	0.0079, 0.0046

Table 1. $\text{dist}(c, e)$ for different dimensions and sample sizes. Left values - K-means, Right values - cosine-K-means

Regarding the question of which distributions this work applies to, the same experiment has been repeated for various distributions D. The distributions that exhibited similar results are: Hyperbolic Secant, Logistic, Cauchy, and Student's t. A common property of most of these is that they are unimodal symmetric and have positive excess kurtosis. A simple adaption of Theorem 1 will therefore tell us that the corresponding `Cluster-ICA` algorithm can be used to solve the ICA task w.r.t. all of the aforementioned distributions.

6.2. Recovery of a predetermined mixing matrix

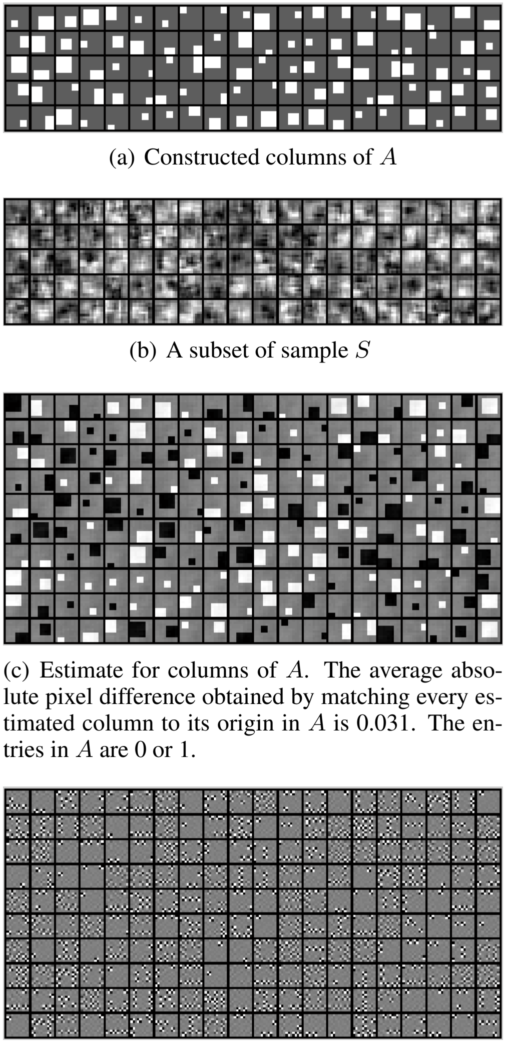
In this experiment we show that `Cluster-ICA` combined with standard K-means solves the ICA task. To estimate the whitening matrix, we use the regularized approach described in (Ng., 2013). The experiment is as follows:

1. Construct 100, 10-by-10 images of rectangles at random locations and sizes and set the columns of mixing matrix A to be their vectorization. (Figure 3.a)

2. Take a sample S of size 5×10^5 from the ICA random variable $x = As_{\text{lap}}$. (Figure 3.b)

3. Run `Cluster-ICA` with standard K-means and S as input to obtain estimate for clumns of A and rows of its inverse

Figure 3.c-d shows the recovered columns and filters. As can be seen in Figure 3.c, we indeed recover the correct matrix A. We also exhibited similar results when replacing s_{lap} with any of the distributions discussed in the above section and repeating the same experiment.



(a) Constructed columns of A

(b) A subset of sample S

(c) Estimate for columns of A. The average absolute pixel difference obtained by matching every estimated column to its origin in A is 0.031. The entries in A are 0 or 1.

(d) Estimate for rows of A^{-1}

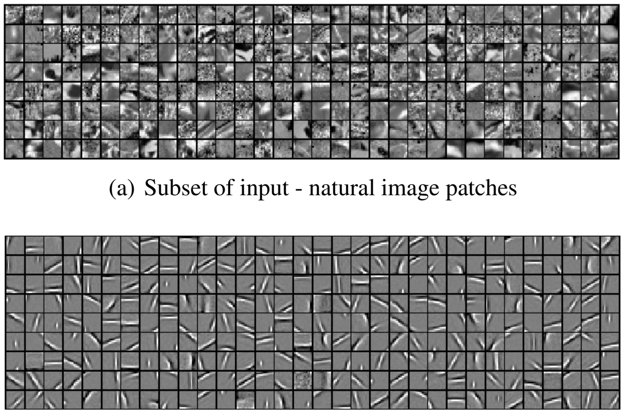
Figure 3. Input and output of `Cluster-ICA`

6.3. The filters obtained for natural images

For natural scenes, ICA has been shown to recover oriented edge-like filters with sparsly distributed outputs (Bell & Sejnowski, 1997). If natural image patches can be captured by an ICA model with sparse independent components, the `Cluster-ICA` algorithm should be able to recover similar filters.

We repeat the same experiment as in the previous section, only this time instead of the sample S we take random 10-by-10 patches from the natural scenes provided by (Olshausen, 1996).

Figure 4 shows that the filters learned by `Cluster-ICA` are indeed oriented edges.



(a) Subset of input - natural image patches

(b) Estimated rows of A^{-1} given by `Cluster-ICA`

Figure 4. `Cluster-ICA` on natural image patches

6.4. K-means initialization technique and practical remarks

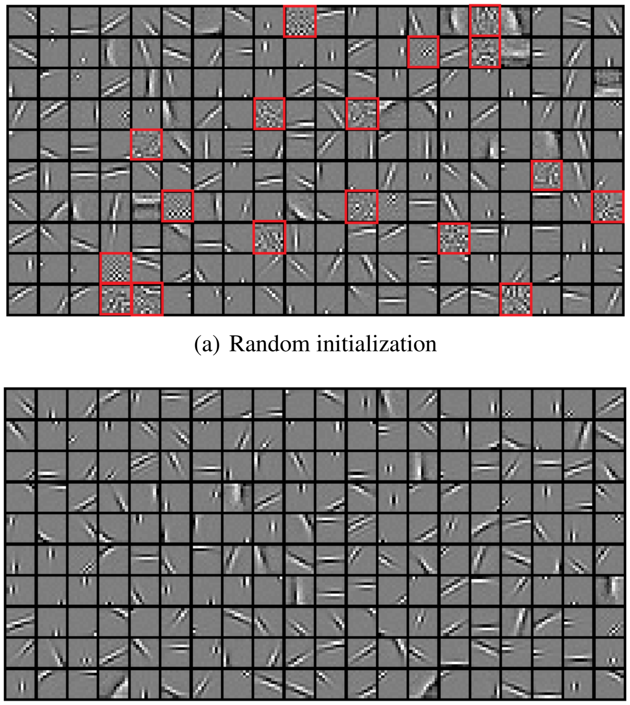
When applying K-means, an important question is how to initialize the centroids. A “RISS” clustering algorithm should return centroids that lie on the axes and `Cluster-ICA` is expected to return a rotated version of that, meaning each centroid’s neighborhood is some quadrant of $\mathbb{R}^d$. We therefore propose the following K-means initialization technique for our context: randomize an orthonormal matrix with columns u_i, and set the initial centroids to $\{u_1, \ldots, u_d, -u_{1,} \ldots, -u_d\}$ with the hope that as the K-means iterations progress, the centroids will rotate themselves symmetrically into the optimal solution.

Figure 5 shows the same experiment as above, with and without the proposed initialization technique. It can be seen that with this technique applied, many of the noisy filters are replaced with clear ones.

A few notes on applying the `Cluster-ICA` algorithm:

1. For numerical stability, it is recommended to apply a regularized estimation of the whitening matrix, as described in (Ng., 2013).

2. Throughout this work we have been setting $k = 2d$. It may be beneficial to learn a larger number of centroids. In practice, K-means may get stuck in a local minimum or the ICA model with square mixing matrix A may not represent reality (e.g. the real A could be overcomplete - more columns than rows). In these cases learning more centroids could recover more of the meaningful filters, which translates into better performance as reported in (Coates et al., 2011). A natural extension of the proposed initialization technique for $k > 2d$ is to scatter centroids on the ℓ_2 sphere evenly, or uniformly at random.



(a) Random initialization

(b) Proposed initialization

Figure 5. Good initialization eliminates noisy filters

7. Interpreting the results of (Coates et al., 2011)

It is interesting to attempt to understand why K-means is the winning approach in (Coates et al., 2011). The training phase in the classification framework in (Coates et al., 2011) essentially implements the `Cluster-ICA` algorithm and returns centroids $\{c^{(i)}\}$. For simplicity of the argument, let us assume that the statistics of the data this framework is applied to can be captured by the ICA model with sparse independent components $x = As$. The feature encoding is:

$$f(x)_i = \max\{0, \mu(z) - z_i\}$$

where $z_i = \|x - c^{(i)}\|_2$ and $\mu(z)$ is the mean of the elements of z. From Theorem 1, if $k = 2d$ then we have $c^{(i)} \approx w_i$, and so

$$\begin{aligned} z_i &\approx \sqrt{\|c^{(i)}\|_2^2 + \|x\|_2^2 - 2w_i^\top x} \\ &\approx \sqrt{\text{constant} - 2s_i} \end{aligned}$$

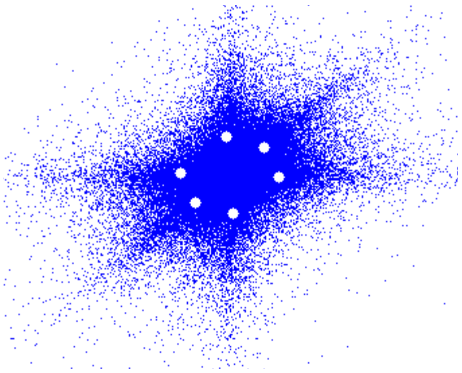
The approximate constant is due to normalization of input x to unit norm, and since $c^{(i)}$ have, at least visually, roughly the same norm. Thus, the features are the sources of a patch with non-linearity on top. When the ICA model does not represent reality (e.g. when A is over-complete) and $k > 2d$ the features could be similar in spirit to sparse coding, as the next section may suggest.

8. Discussion and Open Problems

We have presented the family of "RISS" clustering algorithms whose properties enable us to solve the ICA task with a simple algorithm incorporating the whitening operation. K-means and a variant of it, cosine-K-means, appear to belong to this family. It is interesting to better understand Conjecture 1 and to extend Theorem 3 to higher dimensional spaces, both for K-means and its variant. It is also interesting to analyze convergence rates and compare them to standard methods for solving ICA.

In our analysis "RISS" clustering algorithms have been defined w.r.t the Laplace distribution but as the experiments suggest, K-means behaves similarly for a larger class of distributions. It is interesting to characterize this class. This class of distributions can be regarded as a weaker prior, compared to maximum-likelihood approaches for solving ICA that assume a specific distribution of the independent components.

Perhaps most intriguing is to understand the behavior in over-complete cases. Learning $k > 2d$ centroids over natural patches appears to recover more filters as well as improve classification results, suggesting that K-means may be able to recover an over-complete mixing matrix. Consider, for example, the following over-complete mixture of independent components that have more extreme sparsity than Laplace. K-means appears to recover the columns of the mixing matrix when $k = 6$:



To summarize the practical implications, whenever K-means is being used for dictionary learning, under suitable settings, it may be beneficial to unlock its ICA-like properties by combining the whitening operation, and to treat the resulting centroids according to the interpretation presented. Together with its ability to learn an overcomplete representation, K-means could become a powerful tool.

Acknowledgments

This research is supported by the Intel Collaborative Research Institute for Computational Intelligence (ICRI-CI).

References

Bell, A. and Sejnowski, T. The independent components of natural scenes are edge filters. In *Vision Research*, 1997.

Coates, Adam and Ng, Andrew Y. Learning feature representations with k-means. In *Neural Networks: Tricks of the Trade (2nd ed.)*, pp. 561–580. 2012.

Coates, Adam, Ng, Andrew Y., and Lee, Honglak. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, pp. 215–223, 2011.

Csurka, G., Dance, C., Fan, L., Willamowski, J., , and Bray, C. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

Fei-Fei, L. and Perona., P. A bayesian hierarchical model for learning natural scene categories. In *Conference on Computer Vision and Pattern Recognition*, 2005.

Gupta, A. K. and Song, D. Lp-norm spherical distribution. *Journal of Statistical Planning and Inference*, 60 (2):241–260, May 1997.

Hyvarinen, A. and Oja, E. Independent component analysis: Algorithms and application. In *Neural Networks*, 2000.

Hyvarinen, A., Karhunen, J., and Oja., E. Independent component analysis. In *Wiley Interscience*, 2001.

Hyvarinen, A., Hurri, J., and Hoyer., P. O. Natural image statistics. In *Springer*, 2009.

Lazebnik, S., Schmid, C., and Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition*, 2006.

Le, Q. V., Ngiam, J., Chen, Z., Chia, D., Koh, P. W., and Ng., A. Y. Tiled convolutional neural networks. In *NIPS*, 2010.

Le, Q. V., Karpenko, A., Ngiam, J., and Y., Ng A. Ica with reconstruction cost for efficient overcomplete feature learning. In *NIPS*, 2011.

Le, Quoc V., Ranzato, Marc'Aurelio, Monga, Rajat, Devin, Matthieu, Corrado, Greg, Chen, Kai, Dean, Jeffrey, and Ng, Andrew Y. Building high-level features using large scale unsupervised learning. In *ICML*, 2012.

Ng., A. Y. Unsupervised feature learning and deep learning tutorial. In *http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial*, 2013.

Olshausen, Bruno. Sparse coding simulation software. In *http://redwood.berkeley.edu/bruno/sparsenet/*, 1996.

Pollard, D. Strong consistency of k-means clustering. In *Annals of Statistics 9, 135-140*, 1981.

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., and Gong, Y. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition*, 2010.

Winn, J., Criminisi, A., and Minka., T. Object categorization by learned universal visual dictionary. In *In International Conference on Computer Vision, volume 2*, 2005.