Sifting Common Information from Many Variables

Greg Ver Steeg, Shuyang Gao, Kyle Reing, Aram Galstyan
University of Southern California
Information Sciences Institute
gregv@isi.edu, gaos@usc.edu, reing@usc.edu, galstyan@isi.edu

Abstract

Measuring the relationship between any *pair* of variables is a rich and active area of research that is central to scientific practice. In contrast, characterizing the common information among any *group* of variables is typically a theoretical exercise with few practical methods for high-dimensional data. A promising solution would be a multivariate generalization of the famous Wyner common information, but this approach relies on solving an apparently intractable optimization problem. We leverage the recently introduced information sieve decomposition to formulate an incremental version of the common information problem that admits a simple fixed point solution, fast convergence, and complexity that is linear in the number of variables. This scalable approach allows us to demonstrate the usefulness of common information in high-dimensional learning problems. The sieve outperforms standard methods on dimensionality reduction tasks, solves a blind source separation problem that cannot be solved with ICA, and accurately recovers structure in brain imaging data.

1 Introduction

One of the most fundamental measures of the relationship between two random variables, X_1, X_2, is given by the mutual information, $I(X_1; X_2)$. While mutual information measures the strength of a relationship, the "common information" provides a concrete representation, Y, of the information that is shared between two variables. According to [Wyner, 1975], if Y contains the common information between X_1, X_2, then we should have $I(X_1; X_2|Y) = 0$, i.e., Y makes the variables conditionally independent. We can extend this idea to many variables using the multivariate generalization of mutual information called total correlation [Watanabe, 1960], so that conditional independence is equivalent to the condition $TC(X_1, \ldots, X_n|Y) = 0$ [Xu *et al.*, 2013]. The most succinct Y that has this property represents the multivariate common information in X but finding such a Y in general is a challenging, unsolved problem.

The main contribution of this paper is to show that the concept of common information, long studied in information theory for applications like distributed source coding and cryptography [Kumar *et al.*, 2014], is also a useful concept for machine learning. Machine learning applications have been overlooked due to the intractability of recovering common information for high-dimensional problems. We propose a concrete and tractable algorithmic approach to extracting common information by exploiting a connection with the recently introduced "information sieve" decomposition [Ver Steeg and Galstyan, 2016]. The sieve decomposition works by searching for a single latent factor that reduces the conditional dependence in the data as much as possible. Then the data is transformed to remove this dependence and the "remainder information" trickles down to the next layer. The process is repeated until all the dependence has been extracted and the remainder contains nothing but independent noise. Thm. 3.3 connects the latent factors extracted by the sieve to a measure of common information.

Our second contribution is to show that under the assumptions of linearity and Gaussianity this optimization has a simple fixed-point solution (Eq. 6) with fast convergence and computational complexity linear in the number of variables. Although our final algorithm is limited to the linear case, extracting common information is an unsolved problem and our approach represents a logical first step in exploring the value of common information for machine learning. We offer suggestions for generalizing the method.

Our final contribution is to validate the usefulness of our approach on some canonical machine learning problems. While PCA finds components that explain the most variation, the sieve discovers components that explain the most dependence, making it a useful complement for exploratory data analysis. Common information can be used to solve a natural class of blind source separation problems that are impossible to solve using independent component analysis (ICA) due to the presence of Gaussian sources. Finally, we show that common information outperforms standard approaches for dimensionality reduction and recovering structure in fMRI data.

2 Preliminaries

Using standard notation [Cover and Thomas, 2006], capital X_i denotes a continuous random variable whose instances are denoted in lowercase, x_i. We abbreviate multivariate random variables, $X \equiv X_{1:n} \equiv X_1, \ldots, X_n$, with an associated probability density function, $p_X(X_1 = x_1, \ldots, X_n = x_n)$, which is typically abbreviated to $p(\mathbf{x})$, with vectors in bold. We will index different groups of multivariate random variables with superscripts, X^k, as defined in Fig. 1. We let X^0 denote the original observed variables and we omit the superscript for readability when no confusion results.

Entropy is defined as $H(X) \equiv \langle \log 1/p(\mathbf{x}) \rangle$, where we use brackets for expectation values. Conditional multivariate mutual information, or conditional total correlation, is defined as the Kullback-Leibler divergence between the joint distribution, and the one that is conditionally independent.

$$TC(X|Y) \equiv D_{KL}\left(p(\mathbf{x}|y) \middle\| \prod_{i=1}^{n} p(x_i|y) \right) \tag{1}$$

This quantity is non-negative and zero if and only if all the X_i's are independent conditioned on Y. $TC(X)$ can be obtained by dropping the conditioning on Y in the expression above. In other words, $TC(X) = 0$ if and only if the variables are (unconditionally) independent. If Y were the hidden source of all dependence in X, then $TC(X|Y) = 0$. Therefore, we consider the problem of searching for a factor Y that minimizes $TC(X|Y)$. In the statement of the theorems we make use of shorthand notation, $TC(X;Y) \equiv TC(X) - TC(X|Y)$, which is the reduction of TC after conditioning on Y. This notation mirrors the definition of mutual information between two groups of random variables, X and Y, as the reduction of uncertainty in one variable, given information about the other, $I(X;Y) = H(X) - H(X|Y)$.

3 Extracting Common Information

For Y to contain the common information in X, we need $TC(X|Y) = 0$. Instead of enforcing the condition that $TC(X|Y) = 0$ and looking for the most succinct Y that satisfies this condition, as Wyner does [Wyner, 1975], we consider the dual formulation where we minimize $TC(X|Y_1, \ldots, Y_r)$ subject to constraints on r, the size of the state space [Op't Veld and Gastpar, 2016a]. This optimization can be written equivalently as follows.

$$\min_{\mathbf{y}=\mathbf{f}(\mathbf{x})} TC(X_1, \ldots, X_n | Y_1, \ldots, Y_r) \tag{2}$$

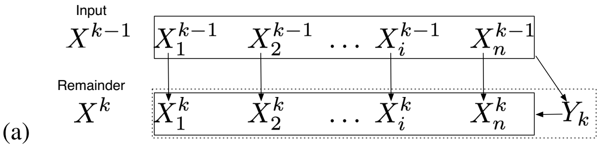
We will show in Thm. 3.3 that an upper bound for this objective is obtained by solving a sequence of optimization problems of the following form, indexed by k.

$$\min_{y_k = f(\mathbf{x}^{k-1})} TC(X_1^{k-1}, \ldots, X_{n_k}^{k-1} | Y_k) \tag{3}$$

The definition of X^k is discussed next, but the high level idea is that we have reduced the difficult optimization over many latent factors in Eq. 2 to a sequence of optimizations with a single latent factor in Eq. 3. Each optimization gives us a tighter upper bound on our original objective, Eq. 2.

Incremental Decomposition We begin with some input data, X, and then construct Y_1 to minimize $TC(X|Y_1)$. After doing so, we would like to transform the original data into the remainder information, X^1, so that we can use the same optimization to learn a factor, Y_2, that extracts more common information that was not already captured by Y_1. We diagram this construction at layer k in Fig. 1 and show in Thm 3.1 the requirements for constructing the remainder information. The result of this procedure is encapsulated in Cor. 3.2 which says that we can iterate this procedure and $TC(X|Y_1, \ldots, Y_k)$ will be reduced at each layer until it reaches zero and Y captures all the common information.

(a)



(b)

$$\begin{aligned}
X^0 &: \mathbf{X_1} \ldots \mathbf{X_n} \\
X^1 &: X_1^1 \ldots X_n^1 \; Y_1 \\
X^2 &: X_1^2 \ldots X_n^2 \; Y_1^2 \; Y_2 \\
&\ldots \\
X^k &: X_1^k \ldots X_n^k \; Y_1^k \; Y_2^k \; Y_k
\end{aligned}$$

Figure 1: (a) This diagram describes one layer of the sieve. Y_k is some function of the X_i^{k-1}'s that is optimized to capture dependence. The remainder, X_i^k contains information that is not explained by Y_k. (b) We summarize the naming convention for multiple layers.

Theorem 3.1. Incremental decomposition of common information *For Y_k a function of X^{k-1}, the following decomposition holds,*

$$TC(X^{k-1}) = TC(X^k) + TC(X^{k-1}; Y_k), \tag{4}$$

if the remainder information X^k satisfies two properties.

1. *Invertibility: there exist functions g, h so that $x_i^{k-1} = g(x_i^k, y_k)$ and $x_i^k = h(x_i^{k-1}, y_k)$*
2. *Remainder contains no information about Y_k: $\forall i, I(X_i^k; Y_k) = 0$*

Proof. We refer to Fig. 1(a) for the structure of the graphical model. We set $\bar{X} \equiv \bar{X}_1, \ldots, \bar{X}_n, Y$ and we will write $\bar{X}_{1:n}$ to pick out all terms except Y. Expanding the definition of $TC(X;Y)$, the equality in Eq. 4 becomes

$$TC(\bar{X}) - TC(X|Y) = \left\langle \log \frac{p(\bar{x}, y) \prod_i p(x_i|y)}{p(y)p(x|y) \prod_i p(\bar{x}_i)} \right\rangle = 0$$

We have to show that this quantity equals zero under the assumptions specified. First, we multiply the fraction by one by putting $\prod_i p(\bar{x}_i|y)$ terms in the numerator and denominator. After applying condition (2) that $I(\bar{X}_i; Y) = 0$, we can remove two terms leaving the following.

$$\left\langle \log \frac{p(\bar{x}|y) \prod_i p(x_i|y)}{p(x|y) \prod_i p(\bar{x}_i|y)} \right\rangle$$

If condition (1) of the theorem is satisfied, then, conditioned on y, $\bar{x}_i$ and x_i are related by a deterministic formula. We can see from applying the change of variables formula for probability distributions that the terms in this expression cancel, leaving us with $\langle \log 1 \rangle = 0$, as we intended to prove. □

The decomposition above was originally introduced for discrete variables as the "information sieve" [Ver Steeg and Galstyan, 2016]; the continuous formulation we introduce here replaces the first condition used in the original statement with an analogous one that is appropriate for continuous variables. Note that because we can always find non-negative solutions for $TC(X^{k-1}; Y_k)$, it must be that $TC(X^k) \leq TC(X^{k-1})$. In other words, the remainder information is more independent than the input data. This is consistent with the intuition that the sieve is sifting out the common information at each layer.

Corollary 3.2. Iterative decomposition of TC *With a hierarchical representation where each Y_k is a function of X^{k-1} and X^k is the remainder information as defined in Thm 3.1, $TC(X) = TC(X^r) + \sum_{k=1}^{r} TC(X^{k-1}; Y_k)$.*

This follows from repeated application of Eq. 4. $TC(X)$ is a constant that depends on the data. For high-dimensional data, it is impossible to measure $TC(X)$, but by learning latent factors extracting progressively more dependence, we get a sequence of better bounds.

Theorem 3.3. Decomposition of common information *For the sieve decomposition, the following bound holds.*

$$TC(X|Y_{1:r}) \leq TC(X^r) = TC(X) - \sum_{k=1}^{r} TC(X^{k-1}; Y_k)$$

Proof. The equality comes from Cor. 3.2.

$$\begin{aligned}
&TC(X_{1:n}|Y_{1:r})\\
&= \left\langle \log \frac{p(x_{1:n}|y_{1:r})}{\prod_{i=1}^{n} p(x_i|y_{1:r})} \right\rangle = \left\langle \log \frac{p(x^r_{1:n}|y^r_{1:r})}{\prod_{i=1}^{n} p(x^r_i|y^r_{1:r})} \right\rangle\\
&= \left\langle \log \frac{p(x^r_{1:n}, y^r_{1:r})}{\prod_{i=1}^{n} p(x^r_i) \prod_{k=1}^{r} p(y^r_k)} \frac{\prod_{i=1}^{n} p(x^r_i) \prod_{k=1}^{r} p(y^r_k)}{p(y^r_{1:r}) \prod_{i=1}^{n} p(x^r_i|y^r_{1:r})} \right\rangle\\
&= TC(X^r) + \left\langle \log \frac{\prod_{i=1}^{n} p(x^r_i) \prod_{k=1}^{r} p(y^r_k)}{p(y^r_{1:r}) \prod_{i=1}^{n} p(x^r_i|y^r_{1:r})} \right\rangle\\
&= TC(X^r) - TC(Y^r_{1:r}) - \sum_{i=1}^{n} I(X^r_i; Y^r_{1:r})\\
&\leq TC(X^r)
\end{aligned}$$

The first line follows from the the change of variables formula for the transformation connecting layer r to the input layer. On the second line we multiply by 1 and re-arrange, collecting terms in the next two lines. The last inequality follows from non-negativity of TC and mutual information. □

Recalling that $TC(X;Y) = TC(X) - TC(X|Y)$, Thm. 3.3 shows how the sum of terms optimized in Eq. 3 provide a successively tighter upper bound on the objective of Eq. 2. In other words, as we keep adding and optimizing latent factors they reduce the conditional TC until all the common information has been extracted.

Optimization It remains to solve the optimization in Eq. 3. For now we drop the k index and focus on minimizing $TC(X|Y)$ for a single factor Y. To get a simple and tractable solution to this non-convex problem, we consider a further simplification where X is Gaussian with covariance matrix Σ and inverse covariance $\Lambda = \Sigma^{-1}$. If X is Gaussian and Y's dependence on X is linear and Gaussian, the joint distribution over X, Y will also be Gaussian. We write out the optimization in Eq. 3 under this condition.

$$\min_{Y|X \sim \mathcal{N}(\mathbf{w}\cdot\mathbf{x}, \eta^2)} \sum_{i=1}^{n} H(X_i|Y) - H(X|Y) \tag{5}$$

Two immediate simplifications are apparent. First, this objective is invariant to scaling of Y. Any solution with $\eta, \mathbf{w}$ would be equivalent to a scaled solution $s\eta, s\mathbf{w}$. Therefore, without loss of generality we set $\eta = 1$. Second, we invoke Bayes rule to see $H(X|Y) = H(Y|X) + H(X) - H(Y)$ where the first two terms on the right hand side are constants with respect to the optimization. We re-write the optimization accordingly.

$$\min_{Y|X \sim \mathcal{N}(\mathbf{w}\cdot\mathbf{x}, 1)} \sum_{i=1}^{n} H(X_i|Y) + H(Y)$$

The objective is invariant to translation of the marginals, so w.l.o.g. we also set $\langle X_i \rangle = \langle Y \rangle = 0$. Define a nonlinear change of variables in terms of the correlation coefficient, $\rho_i = \langle X_i Y \rangle / \sqrt{\langle X_i^2 \rangle \langle Y^2 \rangle}$. To translate between $\mathbf{w}$ and $\boldsymbol{\rho}$, we also note, $(\Sigma \mathbf{w})_i = \langle X_i Y \rangle, \mathbf{w} = \Lambda \boldsymbol{\rho} \sqrt{\langle X_i^2 \rangle \langle Y^2 \rangle}$ and $\langle Y^2 \rangle = 1/(1 - \boldsymbol{\rho}^\top \Lambda \boldsymbol{\rho}) = \mathbf{w}^\top \Sigma \mathbf{w} + 1$. This leads to the following optimization, neglecting some constants.

$$\min_{Y|X \sim \mathcal{N}(\mathbf{w}\cdot\mathbf{x}, 1)} \sum_{i=1}^{n} 1/2 \log(1 - \rho_i^2) - 1/2 \log(1 - \rho^\top \Lambda \rho)$$

Next, we set derivatives with respect to each ρ_i to zero.

$$\partial_{\rho_i} \mathcal{T}C(X|Y) = -\rho_i/(1-\rho_i)^2 + \Lambda \boldsymbol{\rho}/(1 - \boldsymbol{\rho}^\top \Lambda \boldsymbol{\rho}) = 0.$$

Now we use the identities to translate back to a fixed-point equation in terms of $\mathbf{w}$ and rearrange.

$$w_i = \frac{\langle X_i Y \rangle}{\langle X_i^2 \rangle \langle Y^2 \rangle - \langle X_i Y \rangle^2} \tag{6}$$

Interestingly, we arrive at a novel nonlinear twist on the classic Hebbian learning rule [Baldi and Sadowski, 2015]. If X_i and Y "fire together they wire together" (i.e. correlations lead to stronger weights), but this objective strongly prefers correlations that are nearly maximal, in which case the denominator becomes small and the weight becomes large. This optimization of $TC(X|Y)$ for continuous random variables X and Y is, to the best of our knowledge, the first tractable approach except for a special case discussed by [Op't Veld and Gastpar, 2016a]. Also note that although we used Σ, Λ in the derivation, the solution does not require us to calculate these computationally intensive quantities.

A final consideration is the construction of remainder information (i.e., how to get X^k from X^{k-1} and Y in Fig. 1) consistent with the requirements in Thm. 3.1. In the discrete formulation of the sieve, constructing remainder information is a major problem that ultimately imposes a bottleneck on its usefulness because the state space of remainder information can grow quickly. In the linear case, however, the construction of remainder information is a simple linear transformation reminiscent of incremental PCA. We define the remainder information with a linear transformation,

$X_i^k = X_i^{k-1} - \langle X_i^{k-1} Y_k\rangle / \langle Y_k^2\rangle Y_k$. This transformation is clearly invertible (condition (i)), and it can be checked that $\langle X_i^k Y_k\rangle = 0$ which implies $I(X_i^k; Y_k) = 0$ (condition (2)).

Generalizing to the Non-Gaussian, Nonlinear Case The solution for the linear, Gaussian case is more flexible than it looks. We do not actually have to require that the data, X, is drawn from a *jointly* normal distribution to get meaningful results. It turns out that if each of the individual marginals is Gaussian, then the expression for mutual information for Gaussians provides a lower bound for mutual information [Foster and Grassberger, 2011]. Also, the objective (Eq. 2) is invariant under invertible transformations of the marginals [Cover and Thomas, 2006]. Therefore, to ensure that the optimization that we solved (Eq. 5) is a lower bound for the optimization of interest, we should transform the marginals to be individually Gaussian distributed. Several nonlinear, parametric methods to Gaussianize one-dimensional data exist, including a recent method that works well for long-tailed data [Goerg, 2014]. Alternatively, a nonparametric approach is to Gaussianize data based on the rank statistics [Van der Waerden, 1952]. Finally, [Singh and Pøczos, 2017] study information measures for a large family of distributions that can be nonparametrically transformed into normal distributions.

4 Implementation Details

A Single Layer A concrete implementation of one layer of the sieve transformation is straightforward and the algorithm is summarized in Alg. 1. Our implementation is available online [Ver Steeg, 2016]. The minimal preprocessing of the data is to subtract the mean of each variable. Optionally, further Gaussianizing preprocessing can be applied. Our fixed point optimization requires us to start with some weights, $\mathbf{w}^0$ and we iteratively update $\mathbf{w}^t$ using Eq. 6 until we reach a fixed point. This only guarantees that we find a local optima so we typically run the optimization 10 times and take the solution with the highest value of the objective. We initialize $\mathbf{w}_i^0$ to be drawn from a normal with zero mean and scale $1/\sqrt{n\sigma^2_{x_i}}$. We scale each w_i^0 by the standard deviation of each marginal so that one variable does not strongly dominate the random initialization, $y = \mathbf{w}^0 \cdot \mathbf{x}$.

```
Data: Data matrix, $N$ iid samples of vectors, $\mathbf{x} \in \mathbb{R}^n$
Result: Weights, $\mathbf{w}$, so that $y = \mathbf{w} \cdot \mathbf{x}$ optimizes $TC(X;Y)$
        and remainder information, $\bar{\mathbf{x}}$.
Subtract mean from each column of data;
Initialize $w_i \sim \mathcal{N}(0, 1/(\sqrt{n}\sigma_{x_i}))$;
while not converged do
    Calculate $y = \mathbf{w} \cdot \mathbf{x}$ for each sample ;
    Calculate moments from data, $\langle X_i Y\rangle, \langle Y^2\rangle, \langle X_i^2\rangle$;
    $\forall i, w_i \leftarrow \langle X_i Y\rangle/(\langle Y^2\rangle\langle X_i^2\rangle - \langle X_i Y\rangle^2)$;
end
For each column of data, $i$, return $\bar{x}_i = x_i - \frac{\langle X_i Y\rangle}{\langle Y^2\rangle} y$ ;
```

Algorithm 1: Algorithm to learn one layer of the sieve.

The iteration proceeds by estimating marginals and then applying Eq. 6. Estimating the covariance at each step is the main computational burden, but the steps are all linear. If we have N samples and n variables, then we calculate labels for each data point, $y = \mathbf{w} \cdot \mathbf{x}$, which amounts to N dot products of vectors with length n. Then we calculate the covariance, $\langle X_i Y\rangle$, which amounts to n dot products of vectors of length N. These are the most intensive steps and could be easily sped up using GPUs or mini-batches if N is large. Convergence is determined by checking when changes in the objective of Eq. 5 fall below a certain threshold, 10^{-8} in our experiments.

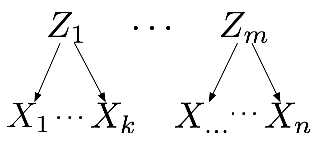


Figure 2: This is the generative model used for synthetic experiments. Each observed variable, $X_i = Z_{pa(i)} + \varepsilon_i$ combines its parent, $Z_{pa(i)}$, with Gaussian noise.

Multiple Layers After training one layer of the sieve, it is trivial to take the remainder information and feed it again through Alg. 1. While our optimization in Eq. 5 formally involved a probabilistic function, we take the final learned function to be deterministic, $y = \mathbf{w} \cdot \mathbf{x}$, as required by Thm. 3.1. Each layer contributes $TC(X^{k-1}; Y_k)$ in our decomposition of $TC(X)$, so we can stop when these contributions become negligible. This occurs when the variables in X^k become independent. In that case, $TC(X|Y_{1:k}) = TC(X^k) = 0$ and since $TC(X^k) \geq TC(X^k; Y_{k+1})$, we get no more positive contributions from optimizing $TC(X^k; Y_{k+1})$.

5 Results

We begin with some benchmark results on a synthetic model. We use this model to show that the sieve can uniquely recover the hidden sources, while other methods fail to do so.

Data Generating Model For the synthetic examples, we consider data generated according to a model defined in Fig. 2. We have m sources, each with unit variance, $Z_j \sim \mathcal{N}(0, 1)$. Each source has k children and the children are not overlapping. Each channel is an additive white Gaussian noise (AWGN) channel defined as $X_i = Z_{pa(i)} + \varepsilon_i$. The noise has some variance that may be different for each observed variable, $\varepsilon_i \sim \mathcal{N}(0, \epsilon_i^2)$. Each channel can be characterized as having a capacity, $C_i = 1/2 \log(1 + 1/\epsilon_i^2)$ [Cover and Thomas, 2006], and we define the total capacity, $C = \sum_{i=1}^k C_i$. For experiments, we set C to be some constant, and we set the noise so that the fraction, C_i/C, allocated to each variable, X_i, is drawn from the uniform distribution over the simplex.

Empirical Convergence Rates We examine how quickly the objective converges by plotting the error at the t-th iteration. The error is defined as the difference between TC at each iteration and the final TC. We take the final value of TC to be the value obtained when the magnitude of changes falls below 10^{-14}. We set $C = 1$ for these experiments. In Fig. 3, we look at convergence for a few different settings of the generative model and see linear rates of convergence (where error is plotted on a log scale, as is conventional for convergence plots),

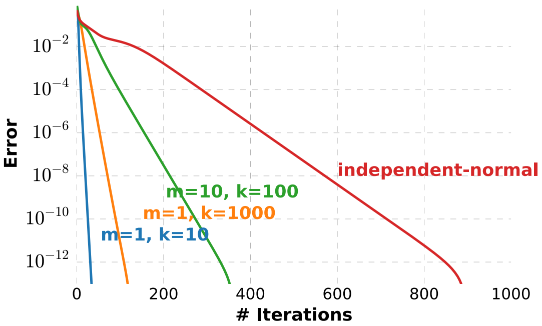


Figure 3: Empirical error plots on synthetic data show linear rates of convergence. We obtained similar results on real-world data.

with a coefficient that seems to depend on problem details. The slowest rate of convergence comes from data where each X_i is generated from an independent normal distribution (i.e., there is no common information).

Recover a Single Source from Common Information As a first test of performance, we consider a simple version of the model in Fig. 2 in which we have just a single source and we have k observed variables that are noisy copies of the source. For this experiment, we set total capacity to $C = 4$. By varying k, we are spreading this capacity across a larger number of noisier variables. We use the sieve to recover a single latent factor, Y, that captures as much of the dependence as possible (Eq. 2), and then we test how close this factor is to the true source, Z, using Pearson correlation. We also compare to various other standard methods: PCA [Halko *et al.*, 2011], ICA [Hyvärinen and Oja, 2000], Non-Negative Matrix Factorization (NMF) [Lin, 2007], Factor Analysis (FA) [Cattell, 1952], Local Linear Embedding (LLE) [Roweis and Saul, 2000], Isomap [Tenenbaum *et al.*, 2000], Restricted Boltzmann Machines (RBMs) [Hinton and Salakhutdinov, 2006], and k-Means [Sculley, 2010]. All methods were run using implementations in the scikit library [Pedregosa *et al.*, 2011].

Looking at the results in Fig. 4(a), we see that for a small number of variables almost any technique suffices to recover the source. As the number of variables rises, however, intuitively reasonable methods fail and only the sieve maintains high performance. The first component of PCA, for instance, is the projection with the largest variance but it can be shown that by changing the scale of the noise in different directions, this component can be made to point in any direction. Unlike PCA, the sieve is invariant under scale transformations of each variable. Error bars are produced by looking at the standard deviation of results over 10 randomly generated datasets. Some error bars are smaller than the plot markers. Besides being the most accurate method, the sieve also has the smallest variance.

5.1 Source Separation with Common Information

In the generative model in Fig. 2, we have m independent sources that are each Gaussian distributed. We could imagine applying an orthonormal rotation, R, to the vector of sources and call these $\tilde{Z}_j = \sum_k R_{jk} Z_k$. Because of the Gaussianity of the original sources, $\tilde{Z}$ also represent m independent Gaussian sources. We can write down an equivalent generative model for the X_i's, but each X_i now depends on all the $\tilde{Z}$ (i.e., $X_i = \sum_j R^{-1}_{i,j} \tilde{Z}_j + \varepsilon_i$). From a generative model perspective, our original model is unidentifiable and therefore independent component analysis cannot recover it [Hyvärinen and Oja, 2000]. On the other hand, the original generating model is special because the common information about the X_i's are localized in invidivual sources, while in the rotated model, you need to combine information from all the sources to predict any individual X_i. The sieve is able to uniquely recover the true sources because they represent the optimal way to sift out common information.

To measure our ability to recover the independent sources in our model, we consider a model with $m = 10$ sources and varying numbers of noisy observations. The results are shown in Fig. 4(b). We learn 10 layers of the sieve and check how well $Y_1, \ldots, Y_{10}$ recover the true sources. We also specify 10 components for the other methods shown for comparison. As predicted, ICA does not recover the independent sources. While the generative model is in the class described by Factor Analysis (FA), there are many FA models that are equally good generative models of the data. In other words, FA suffers from an identifiability problem that makes it impossible to uniquely pick out the correct model [Shalizi, 2013]. In contrast, common information provides a simple and effective principle for uniquely identifying the true sources.

Exploratory Data Analysis The first component of PCA explains the most variance in the data, and the weights of the first component are often used in exploratory analysis to understand the semantics of discovered factors. Analogously, the first component of the sieve extracts the largest source of common information. In Fig. 5 we compare the top components learned by the sieve on the Olivetti faces dataset to those learned by PCA. The sieve may be more practical for extracting components if data is high dimensional since its complexity is linear in the number of variables while PCA is quadratic. Like PCA, we can also use the sieve for reconstructing data from a small number of learned factors. Note that the sieve transform is invertible so that $X_i = X_i^1 + \langle X_i^0 Y_1 \rangle / \langle Y_1^2 \rangle Y_1$. If we have a sieve transformation with r layers, then we can continue this expansion as follows.

$$X_i = X_i^r + \sum_{k=1}^{r} \langle X_i^{k+1} Y_k \rangle / \langle Y_k^2 \rangle Y_k$$

If we knew the remainder information, X_i^r, this reconstruction would be perfect. However, we can simply set the $X_i^r = 0$ and we will get a prediction for X_i based only on the learned factors, Y, as in Fig. 5.

Source Separation in fMRI Data To demonstrate that our approach is practical for blind source separation in a more realistic scenario, we applied the sieve to recover spatial brain components from fMRI data. This data is generated according to a synthetic but biologically motivated model that incorporates realistic spatial modes and heterogeneous temporal

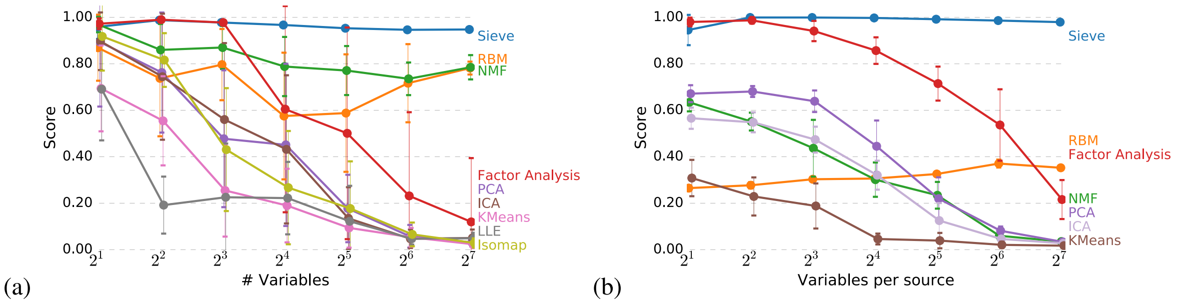


Figure 4: Each source is compared to the best match of the components returned by each method. The score is the average of the absolute Pearson correlations. Each point is a mean score over ten randomly generated datasets, with error bars representing standard deviation. (a) We attempt to recover a single hidden source variable from varying numbers of observed variables. We set $C = 4$ and use 500 samples. (b) We attempt blind source separation for ten independent, hidden source variables given varying numbers of observed variables per source. We set $C = 12$ and use 10000 samples.

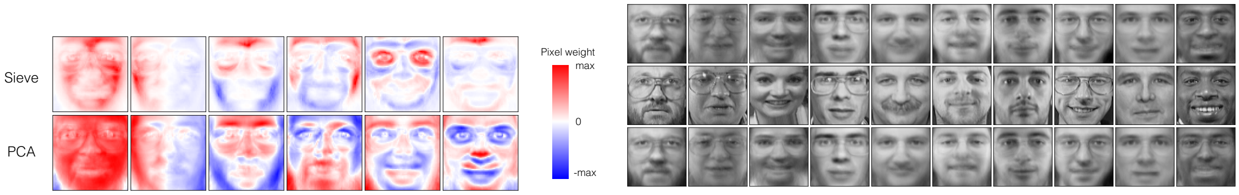


Figure 5: (Left) The top 6 components for the Olivetti faces dataset using the information sieve (top) and PCA (bottom). Red and blue correspond to negative and positive weights respectively. (Right) We take Olivetti faces (middle row) and then try to reconstruct them using the top 20 components from the sieve (top row) or PCA (bottom row).

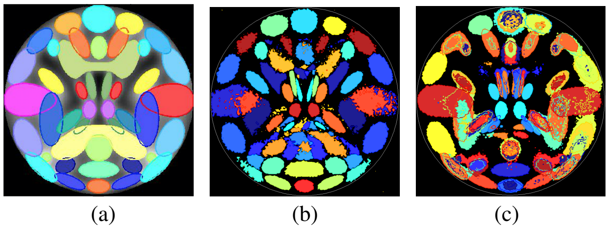


Figure 6: Colors represent different spatial components. (a) The spatial map of 27 components used to generate fMRI data. (b) 27 spatial components recovered by the information sieve. (c) 27 spatial components recovered by ICA where components visualize the recovered mixing matrix.

signals [Erhardt *et al.*, 2012]. We show in Fig. 6(b) that we recover components that match well with the true spatial components. For comparison, we show ICA's performance in Fig. 6(c) which looks qualitatively worse. ICA's poor performance for recovering spatial MRI components is known and various extensions have been proposed to remedy this [Allen *et al.*, 2012]. This preliminary result suggests that the concept of "common information" may be a more useful starting point than "independent components" as an underlying principle for brain imaging analysis.

5.2 Dimensionality Reduction

The sieve can be viewed as a dimensionality reduction (DR) technique. Therefore, we apply various DR methods to two standard datasets and use a Support Vector Machine with a Gaussian kernel to compare the classification accuracy after dimensionality reduction. The two datasets we studied were GISETTE and MADELON and consist of 5000 and 500 dimensions respectively. For each method and dataset, we learn a low-dimensional representation on training data and then transform held-out test data and report the classification accuracy on that. The results are summarized in Fig. 7.

For the GISETTE dataset, we see factor analysis, the sieve, and PCA performing the best, producing low dimensional representations with similar quality using a relatively small number of dimensions. For the MADELON dataset, the sieve representation gives the best accuracy with factor analysis and PCA resulting in accuracy drops of about five and ten percent respectively. Interestingly, all three techniques peak at five dimensions, which was intended to be the correct number of latent factors embedded in this dataset [Guyon *et al.*, 2004].

6 Related Work

Although the sieve is linear, the information objective that is optimized is nonlinear so the sieve substantially differs from methods like PCA. Superficially, the sieve might seem related to methods like Canonical Correlation Analysis (CCA) that seek to find a Y that makes X and Z independent, but that method requires some set of labels, Z. One possibility would be to make Z a copy of X, so that Y is reducing dependence between X and a copy of itself [Wang *et al.*, 2010]. However, this objective differs from common information as can be seen by considering the case where X consists of independent

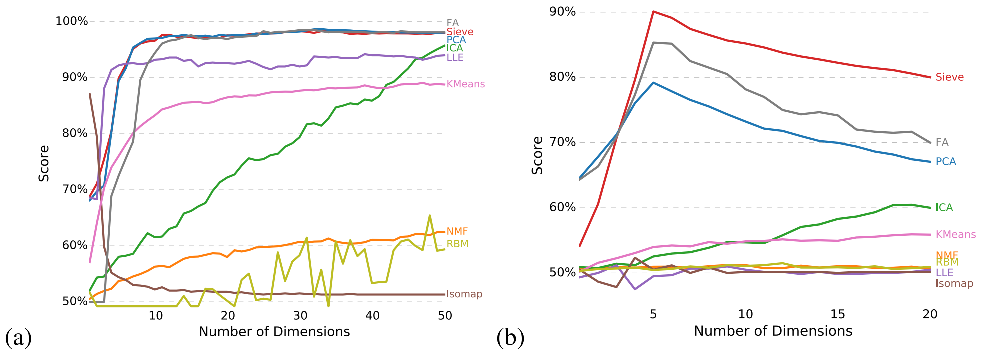


Figure 7: (a) Validation accuracy for GISETTE dataset (b) Validation accuracy for MADELON dataset. All the scores are averaged by running 20 trials.

variables. In that case the common information within X is zero, but X and its copy still have dependence. The concept of "common information" has largely remained restricted to information-theoretic contexts [Xu *et al.*, 2013; Wyner, 1975; Kumar *et al.*, 2014; Op't Veld and Gastpar, 2016a; Op't Veld and Gastpar, 2016b]. The common information in X that is *about* some variable, Z, is called intersection information and is also an active area of research [Griffith *et al.*, 2014].

Insofar as the sieve reduces the dependence in the data, it can be seen as an alternate approach to independent component analysis [Comon, 1994] that is more directly comparable to "least dependent component analysis" [Stögbauer *et al.*, 2004]. As an information theoretic learning framework, the sieve could be compared to the information bottleneck [Tishby *et al.*, 2000], which also has an interesting Gaussian counterpart [Chechik *et al.*, 2005]. The bottleneck requires labeled data to define its objective. In contrast, the sieve relies on an unsupervised objective that fits more closely into a recent program for decomposing information in high-dimensional data [Ver Steeg and Galstyan, 2014; Ver Steeg and Galstyan, 2015; Ver Steeg and Galstyan, 2016], except that work focused on discrete latent factors.

The sieve could be viewed as a new objective for projection pursuit [Friedman, 1987] based on common information. The sieve stands out from standard pursuit algorithms in two ways. First, an information based "orthogonality" criteria for subsequent projections naturally emerges and, second, new factors may depend on factors learned at previous layers (note that in Fig. 1 each learned latent factor is included in the remainder information that is optimized over in the next step). More broadly, the sieve can be viewed as a new approach to unsupervised deep representation learning [Bengio *et al.*, 2013; Hinton and Salakhutdinov, 2006]. In particular, our setup can be directly viewed as an auto-encoder with a novel objective [Bengio *et al.*, 2007]. From that point of view, it is clear that the sieve can also be directly leveraged for unsupervised density estimation [Dinh *et al.*, 2014].

7 Conclusion

We introduced a new scheme for incrementally extracting common information from high-dimensional data. The foundation of our approach is an efficient information theoretic optimization that finds latent factors that capture as much information about multivariate dependence in the data as possible. With a practical method for extracting common information from high-dimensional data, we were able to explore new applications of common information in machine learning. Besides promising applications for exploratory data analysis and dimensionality reduction, common information seems to provide a compelling approach to blind source separation.

While the results here relied on assumptions of linearity and Gaussianity, the invariance of the objective under nonlinear marginal transforms, a common ingredient in deep learning schemes, suggests a straightforward path to generalization that we leave to future work. The greedy nature of the sieve construction may be a limitation so another potential direction would be to jointly optimize several latent factors at once. Sifting out common information in high-dimensional data provides a practical and distinctive new principle for unsupervised learning.

Acknowledgments

GV thanks Sanjoy Dasgupta, Lawrence Saul, and Yoav Freund for encouraging exploration of the linear, Gaussian case for decomposing multivariate information. This work was supported in part by DARPA grant W911NF-12-1-0034 and IARPA grant FA8750-15-C-0071.

References

[Allen *et al.*, 2012] Elena A Allen, Erik B Erhardt, Yonghua Wei, Tom Eichele, and Vince D Calhoun. Capturing inter-subject variability with group independent component analysis of fmri data: a simulation study. *Neuroimage*, 59(4), 2012.

[Baldi and Sadowski, 2015] Pierre Baldi and Peter Sadowski. The ebb and flow of deep learning: a theory of local learning. *arXiv preprint arXiv:1506.06472*, 2015.

[Bengio *et al.*, 2007] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19:153, 2007.

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[Cattell, 1952] Raymond B Cattell. Factor analysis: an introduction and manual for the psychologist and social scientist. 1952.

[Chechik *et al.*, 2005] Gal Chechik, Amir Globerson, Naftali Tishby, and Yair Weiss. Information bottleneck for gaussian variables. In *Journal of Machine Learning Research*, pages 165–188, 2005.

[Comon, 1994] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[Cover and Thomas, 2006] Thomas M Cover and Joy A Thomas. *Elements of information theory*. Wiley-Interscience, 2006.

[Dinh *et al.*, 2014] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[Erhardt *et al.*, 2012] Erik B Erhardt, Elena A Allen, Yonghua Wei, Tom Eichele, and Vince D Calhoun. Simtb, a simulation toolbox for fmri data under a model of spatiotemporal separability. *Neuroimage*, 59(4):4160–4167, 2012.

[Foster and Grassberger, 2011] David V Foster and Peter Grassberger. Lower bounds on mutual information. *Phys. Rev. E*, 83(1):010101, 2011.

[Friedman, 1987] Jerome H Friedman. Exploratory projection pursuit. *Journal of the American statistical association*, 82(397), 1987.

[Goerg, 2014] Georg M. Goerg. The lambert way to gaussianize heavy-tailed data with the inverse of tukey's h transformation as a special case. *The Scientific World Journal: Special Issue on Probability and Statistics with Applications in Finance and Economics*, 2014.

[Griffith *et al.*, 2014] Virgil Griffith, Edwin KP Chong, Ryan G James, Christopher J Ellison, and James P Crutchfield. Intersection information based on common randomness. *Entropy*, 16(4):1985–2000, 2014.

[Guyon *et al.*, 2004] Isabelle Guyon, Steve Gunn, Asa Ben-Hur, and Gideon Dror. Result analysis of the nips 2003 feature selection challenge. In *Advances in neural information processing systems*, pages 545–552, 2004.

[Halko *et al.*, 2011] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[Hyvärinen and Oja, 2000] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4):411–430, 2000.

[Kumar *et al.*, 2014] G Ramesh Kumar, Cheuk Ting Li, and Abbas El Gamal. Exact common information. In *Information Theory (ISIT), 2014 IEEE International Symposium on*. IEEE, 2014.

[Lin, 2007] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[Op't Veld and Gastpar, 2016a] Giel Op't Veld and Michael C Gastpar. Caching gaussians: Minimizing total correlation on the gray–wyner network. In *50th Annual Conference on Information Systems and Sciences (CISS)*, 2016.

[Op't Veld and Gastpar, 2016b] Giel J Op't Veld and Michael C Gastpar. Total correlation of gaussian vector sources on the gray-wyner network. In *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*, pages 385–392. IEEE, 2016.

[Pedregosa *et al.*, 2011] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Roweis and Saul, 2000] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[Sculley, 2010] David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pages 1177–1178. ACM, 2010.

[Shalizi, 2013] Cosma Shalizi. Advanced data analysis from an elementary point of view, 2013.

[Singh and Pøczos, 2017] Shashank Singh and Barnabás Pøczos. Nonparanormal information estimation. *arXiv preprint arXiv:1702.07803*, 2017.

[Stögbauer *et al.*, 2004] Harald Stögbauer, Alexander Kraskov, Sergey A Astakhov, and Peter Grassberger. Least-dependent-component analysis based on mutual information. *Physical Review E*, 70(6):066123, 2004.

[Tenenbaum *et al.*, 2000] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[Tishby *et al.*, 2000] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv:physics/0004057*, 2000.

[Van der Waerden, 1952] BL Van der Waerden. Order tests for the two-sample problem and their power. In *Indagationes Mathematicae (Proceedings)*, volume 55, pages 453–458. Elsevier, 1952.

[Ver Steeg and Galstyan, 2014] Greg Ver Steeg and Aram Galstyan. Discovering structure in high-dimensional data through correlation explanation. *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[Ver Steeg and Galstyan, 2015] Greg Ver Steeg and Aram Galstyan. Maximally informative hierarchical representations of high-dimensional data. In *Artificial Intelligence and Statistics (AISTATS)*, 2015.

[Ver Steeg and Galstyan, 2016] Greg Ver Steeg and Aram Galstyan. The information sieve. In *International Conference on Machine Learning (ICML)*, 2016.

[Ver Steeg, 2016] Greg Ver Steeg. Linear information sieve code. `http://github.com/gregversteeg/LinearSieve`, 2016.

[Wang *et al.*, 2010] Meihong Wang, Fei Sha, and Michael I Jordan. Unsupervised kernel dimension reduction. In *Advances in Neural Information Processing Systems*, pages 2379–2387, 2010.

[Watanabe, 1960] Satosi Watanabe. Information theoretical analysis of multivariate correlation. *IBM Journal of research and development*, 4(1):66–82, 1960.

[Wyner, 1975] Aaron D Wyner. The common information of two dependent random variables. *Information Theory, IEEE Transactions on*, 21(2):163–179, 1975.

[Xu *et al.*, 2013] Ge Xu, Wei Liu, and Biao Chen. Wyner's common information: Generalizations and a new lossy source coding interpretation. *arXiv preprint arXiv:1301.2237*, 2013.