Machine Intelligence 2 **Exercise Sheet 5**
SoSe 2022, Obermayer/Kashef/Strömsdörfer due: 2022-06-09 at 23:55

Independent Component Analysis: Infomax

Exercise T5.1: The Infomax method (tutorial)

(a) What is the difference between uncorrelatedness and statistical independence of random variables?

(b) Are you familiar with the following measures from information theory?

- Entropy
- Conditional Entropy
- Relative Entropy or Kullback-Leibler (KL) divergence
- Mutual Information

(c) What is the Infomax principle?

(d) How do we formulate the optimization objective for the Infomax method?

(e) How do we apply empirical risk minimization to the Infomax method?

(f) How do we train a model that uses the Infomax method for solving the ICA problem?

This exercise is about implementing the *Infomax Principle* for Independent Component Analysis (ICA) using gradient based learning. The files `sound1.dat` and `sound2.dat` in `sounds.zip` contain recordings of two acoustic sources. Your implementation should contain the following steps:

Exercise H5.1: Initialization (homework, 2 points)

(a) Load the sound files. Each of the $N = 2$ sources is sampled at 8192 Hz and contains $p = 18000$ samples.

(b) Create a random and invertible[1] $N \times N$ mixing matrix $\underline{\mathbf{A}}$ and mix the sources:

$$\underline{\mathbf{x}}^{(\alpha)} = \underline{\mathbf{A}}\,\underline{\mathbf{s}}^{(\alpha)} \qquad \alpha = 1, \ldots, p$$

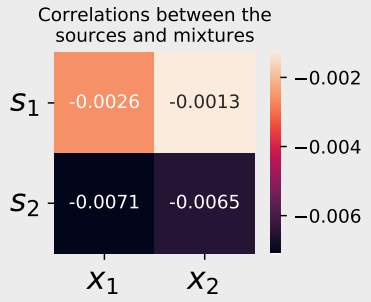
(c) Remove the temporal structure by permuting randomly the columns of the $N \times p$ data matrix $\underline{\mathbf{X}} = (\underline{\mathbf{x}}^{(1)}, \ldots, \underline{\mathbf{x}}^{(p)})$. Use these shuffled mixtures data in all subsequent steps.

(d) Calculate the correlations between the sources and the mixtures: $\rho_{s_i,x_j} = \dfrac{\mathrm{cov}(s_i, x_j)}{\sigma_{s_i}\sigma_{x_j}}$, with covariance in the numerator and standard deviations in the denominator.

[1] You can simply check that the matrix is invertible and re-create it, if it's not.

Solution

(d)



The absence of any correlation between the sources and the mixtures is a result of removing all temporal structure from the mixtures in the previous step (c).

(e) Center the data s.t. that each observed variable x_i has zero mean.

(f) Initialize the unmixing matrix $\underline{\mathbf{W}}$ with random values.

Exercise H5.2: Optimization (homework, 4 points)

Implement a *matrix version* of the ICA *online* learning algorithm that iterates as often as required over the training data. For $\hat{f}(\cdot)$ use the logistic function (see lecture slides). This should reduce your code for this part to a few lines. Implement two variants of this learning algorithm:

(a) Compute the update matrix $\Delta\underline{\mathbf{W}}$ using the standard gradient.

(b) Compute the update matrix $\Delta\underline{\mathbf{W}}$ using the *natural gradient* as described in the lecture.

(c) Find a suitable learning rate ε that decays exponentially (but sufficiently slowly: e.g. $\varepsilon_0 = 0.01$, $\varepsilon_{t+1} = 0.9999\varepsilon_t$), and apply both gradient algorithms to the data (after it has been shuffled and centered) for unmixing the sources.

Hint: :

You can start by implementing the component-wise update of the weights before figuring out the matrix version. The only advanatage of the matrix version is to speed up your implementation and to practice how to "vectorize" your implementation. You can use the component-wise implementation as a means to verify your implementation.

Solution

(a) $$\Delta\underline{\mathbf{W}} = \varepsilon \left(\underline{\mathbf{W}}^{-1}\right)^{\top} + \left(\frac{\hat{f}''\left(\underline{\mathbf{W}}\,\underline{\mathbf{x}}^{(\alpha)}\right)}{\hat{f}'\left(\underline{\mathbf{W}}\,\underline{\mathbf{x}}^{(\alpha)}\right)} \right)^{\top} \underline{\mathbf{x}}^{(\alpha)}$$

(b) Multiplying the above $\Delta\underline{\mathbf{W}}$ with the matrix product $\underline{\mathbf{W}}^{\top}\underline{\mathbf{W}}$ gives us an expression for the natrual gradient. For a derivation, see Haykin Ch. 10.14 Eq. 10.138.

$$\Delta\underline{\mathbf{W}}_{\text{natural}} = \varepsilon \left(\underline{\mathbf{I}} + \frac{\hat{f}''\left(\underline{\mathbf{W}}\,\underline{\mathbf{x}}^{(\alpha)}\right)}{\hat{f}'\left(\underline{\mathbf{W}}\,\underline{\mathbf{x}}^{(\alpha)}\right)} \left(\underline{\mathbf{W}}\,\underline{\mathbf{x}}^{(\alpha)}\right)^{\top} \right) \underline{\mathbf{W}}$$

The component-wise implementation for the natural gradient update rule:

```
sigmoid = lambda x: 1. / (1.+np.exp(-x))

def ica_natgrad_componentwise(x_alpha, N, W):
    dW = np.zeros_like(W)
    for i in range(N):
        for j in range(N):
            wk = np.dot(W[i,:], x_alpha)
            for ll in range(N):
                dil = int(i==ll)
                fpp_fp = 1.-2.*sigmoid(wk) # f"(y)/f'(y)
                dW[i,j] += (dil + fpp_fp*np.dot(W[ll,:], x_alpha)) * W[ll,j]
    return dW
```

Exercise H5.3: Results (homework, 4 points)

(a) Plot & Play[2]

(i) the original sounds,

(ii) the mixed sources (before and after the data permutation),

(iii) the recovered signals (estimated sources) $\hat{\underline{\mathbf{s}}} = \underline{\mathbf{W}}\,\underline{\mathbf{x}}$ using the *unpermuted* data.

(b) Calculate the correlations (as above) between the true sources and the estimations.

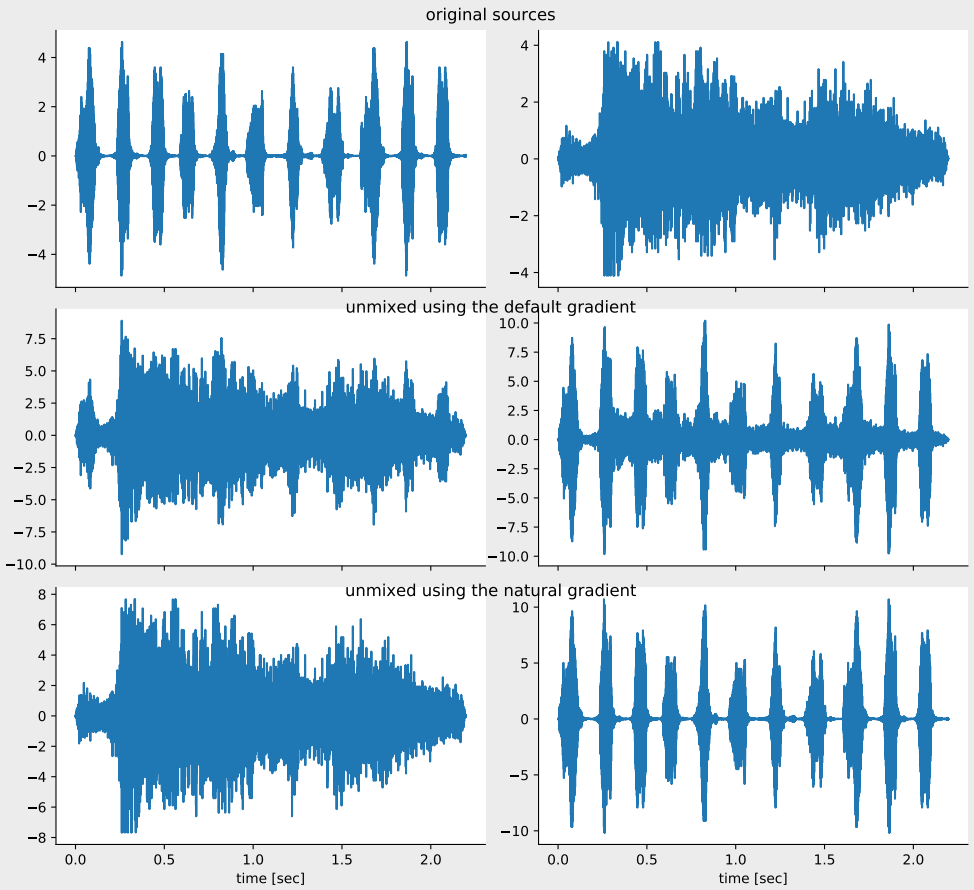
(c) For every 1000$^{\text{th}}$ update, plot the square of the Frobenius norm $||\Delta\underline{\mathbf{W}}||_F^2 := \sum_{i=1,j=1}^{N}(\Delta w_{ij})^2$ to compare the convergence speed for the two gradient methods. Whiten your data before applying ICA and compare the learning speeds again. Describe the differences between the two variants of the learning algorithm.

(d) Plot the density of the mixed, unmixed, and true signals & interpret your results.

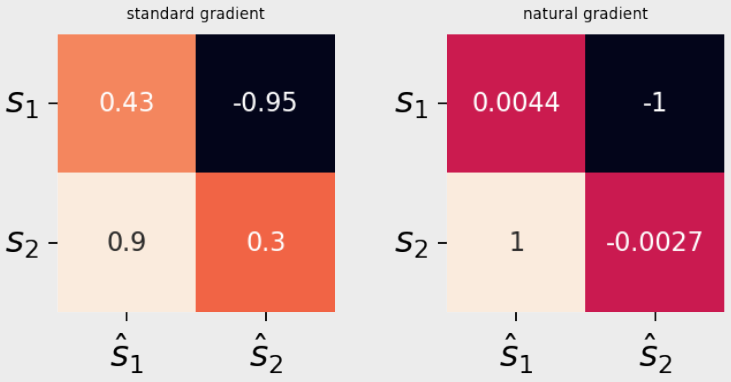
Solution

[2]Python users can use `scipy.io.wavfile` to save a signal to a playable audio file.

(a) Unmixed signals



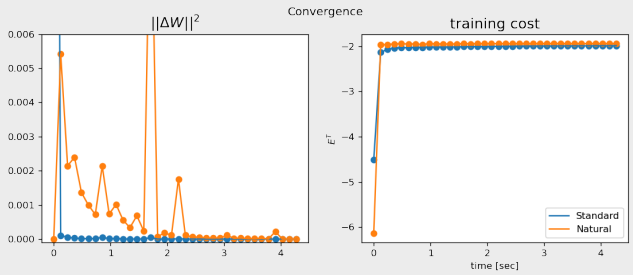
(b) The Correlations between the original signals and reconstructions



We look at the above correlations to verify the quality of our reconstructions.
Ideally, the reconstruction $\hat{s}_i$ will correlate very strongly with exactly one of the original sources, i.e. ± 1. The sign of the correlation is irrelevant and expected because ICA cannot resolve the sign of the signal. Repeating the process will also reveal the permutation ambiguity of ICA in that the value of the correlation between $\hat{s}_i$ with both sources can switch between independent runs.

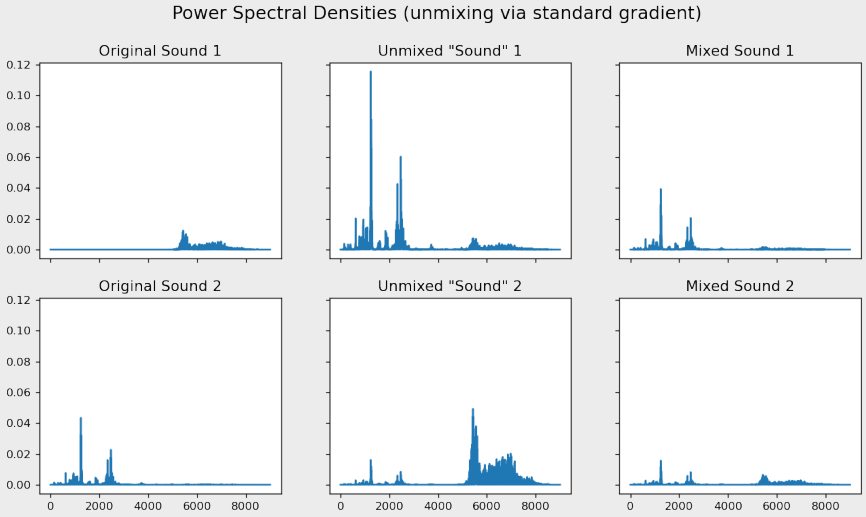
The above correlations show that higher correlations between the reoncstructions via the natural gradient vs. those obtained via the standard gradient, implying a better reconstruction of the sources via natural gradient update rule.

(c) Convergence speed:



The modification introduced by the natural gradient leads to "instability" of the norm of the weight matrix. However, we see from the training cost traces, that it is faster at optimizing the cost function than the standard gradient.

(d) Comparison of (power spectral) densities



The purpose of examining the densities in this way is to see how ICA finds filters to extract each source. You can treat the units of the x-axis above as *frequencies*. The two sound files we mixed in this exercise occupy different frequency bands. A classicial signal processing approach to separating the sources from the mixtures would entail constructing the appropriate low-pass and high-pass filter for each source. ICA accomplishes the same effect inspite of removing the temporal structure before feeding it into the algorithm.

total: 10 points