# Infomax

The second topic we will consider is the Infomax algorithm. This is a approach to unmixing mixed data proposed in Bell and Sejnowski (1995) and gives an in-principle explanation for how the brain might perform auditory source separation. In other words, it shows how source separation might be performed and indicates some of the ingredients that may be present in source separation in the brain.

The problem is as follows: imagine you are in a crowded room, in the classic telling, at a cocktail party. Lots of people are talking, but if you concentrate on one voice you can separate it from the overall hub bub. We can experience the opposite effect when meditating or sitting in thoughtful silence: we suddenly notice how loud distance noises are, the noise of voices on the street, other people coughing and clearing their throats, the sound of wind through nearby trees. All these are sounds we automatically filter out when listening. This process of picking out individual sources of sound from a mixture of sounds is called auditory source separation, or, the question of how to do this is called the cocktail party problem. In fact, the problem is much more general than auditory source separation; the same techniques should apply to any problem where we can observe a mixed signal and would like the unmixed signal. In this more general context the problem of source separation is called independent component analysis.

As noted above, we are interested in the particularly neuronal approach described in Bell and Sejnowski (1995); however this isn't the only approach to the problem and a different algorithm, fastICA (Hyvarinen, 1999), is the most popular algorithm for source separation.

The problem can be phrased like this, given the sources $\mathbf{s}(t)$ where $\mathbf{s}$ is a vector over multiple sources. Now we do source separation with only two recordings, one for each ear; here we are just going to consider the simpler problem of source separation when there are as many recordings are as there are sources, we also assume the mixing is linear and instantaneous, real mixing of auditory signals in a room will only have these properties approximately. However, given these assumption we have

$$\mathbf{r}(t) = M\mathbf{s}(t) \tag{1}$$

with $M$ a square mixing matrix. The goal is to find the unknown source signals $\mathbf{s}(t)$ from the known recordings $\mathbf{r}(t)$. Although it makes little difference, lets restrict ourselves to two sources, so **s** and **r** are two-dimensional vectors and $M$ is a two-by-two matrix.

We assume the two sources are independent, $p_{S_1,S_2} = p_{S_1}p_{S_2}$; the point of source separation is to unmix independent sources. Now, we want to find an unmixing matrix $W$ so that knowing

$$\mathbf{x}(t) = W\mathbf{r}(t) \tag{2}$$

is as good as knowing the sources; precisely, since $\mathbf{x} = WM\mathbf{s}$ we want $WM$ to be a diagonal matrix multiplied by a permutation matrix, we do not mind if unmixing changes the overall amplitude of the source, or if it reorders the sources. In this two-to-two example, that means

$$MW = \text{diag}\,(d_1, d_2) \tag{3}$$

or

$$MW = \left(\begin{array}{cc} 0 & d_1 \\ d_2 & 0 \end{array}\right) \tag{4}$$

where $d_1$ and $d_2$ are real numbers. Hence:

$$\mathbf{s} \overset{\text{mixing}}{\longrightarrow} \mathbf{r} = M\mathbf{s} \overset{\text{unmixing}}{\longrightarrow} \mathbf{x} = W\mathbf{r} \tag{5}$$

One difficulty with looking at this problem is that it involves continuous random variables, whereas the unit so far has concentrated on the discrete case: $s_1(t)$ and $s_2(t)$ are continuous variable. For this reason some of the results will just be quoted and any exam questions will reflect the difficulty of this topic.

The main point with continuous random variables is that the probability distribution is now a density, so

$$Pr(a < x < b) = \int_a^b p_X(x)dx \tag{6}$$

with

$$\int_{-\infty}^{\infty} p_X(x) = 1dx \tag{7}$$

An important difference is demonstrated by the following formula: if $Y = g(X)$ are two random variables related by an invertible function $g$

$$p_Y(\lambda x) = \frac{p_X(x = g^{-1}(y))}{|g'(x)|} \tag{8}$$

To see roughly where this comes from, consider a simple scaling $Y = \lambda X$, now, to preserve the probabilities

$$\int_{\lambda a}^{\lambda b} p_Y(y)dy = \int_a^b p_X(x)dx \tag{9}$$

but if you do a change of variables $y = \lambda x$ on the left hand side this give

$$\int_a^b p_Y(\lambda x)\lambda dx = \int_a^b p_X(x)dx \tag{10}$$

which means

$$p_Y(y) = \frac{p_x(x)}{\lambda} \tag{11}$$

The information is defined in the same way as before

$$H(X) = -\int p(x)\log p(x)dx \tag{12}$$

but it is no longer always positive. Furthermore, by substituting from the formula above, you can see that

$$H(\lambda X) = H(X) + \log|\lambda| \tag{13}$$

and so the entropy has no maximum or minimum: a similar idea is the maximization of entropy over distributions with the same variance and this would rule out the scaling we see in this formula.

Now, the idea is to solve the problem by using the fact that $S_1$ and $S_2$ are independent: we just need to find $W$ so that $X_1$ and $X_2$ are also independent. One approach might be to decorrellate the random variables:

$$C(X_1, X_2) = \langle (X_1 - \langle X_1\rangle)(X_2 - \langle X_2\rangle)\rangle_{(X_1,X_2)} \tag{14}$$

where the expectation value for continuous random variables has the obvious definition

$$\langle g(X)\rangle = \int p_X(x)g(x)dx \tag{15}$$

It is easy to check that the correlation vanishes if $X_1$ and $X_2$ are independent, however, the flaw in this approach is that the converse is not true, it is possible to have zero correlation while still having statistical dependence. To see this, imagine, for convenience and without loss of generality, that $EX_1$

and $EX_2$ are zero and that we have chosen $W$ so that the correlation matrix is the identity:

$$C_{ab} = C(X_a, X_b) = \mathbf{1} \tag{16}$$

then it is easy to see that rotations

$$\left(\begin{array}{c} X_1' \\ X_2' \end{array}\right) = \left(\begin{array}{cc} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{array}\right)\left(\begin{array}{c} X_1 \\ X_2 \end{array}\right) \tag{17}$$

do not change the correlation matrix. Thus, the decorrelation prescription has a rotational ambiguity and something more is needed. That something, of course, is to require $I(X_1, X_2) = 0$ since this happens if and only if $X_1$ and $X_2$ are independent.

The problem is that $I(X_1, X_2)$ is difficult to calculate: the idea behind infomax is to look at $H(X_1, X_2)$:

$$I(X_1, X_2) = H(X_1) + H(X_2) - H(X_1, X_2) \tag{18}$$

The idea is that maximizing the joint entropy, $H(X_1, X_2)$ will give a minimum of the mutual information, in other words, the variations in the individual entropies $H(X_1)$ and $H(X_2)$ can be ignored. However, as stated, this will not work because the entropy can be increased by a trivial scaling, $X_a \rightarrow \lambda X_a$, changes the joint entropy, $H(X_1, X_2) \rightarrow H(X_1, X_2) + \log|\lambda|$ so $H(X_1, X_2)$ can be made arbitrarily large by scaling, something that tells us nothing about mixing and unmixing. Inspired by the behaviour of neurons, in Bell and Sejnowski (1995) this is solved by adding a saturation non-linearity:

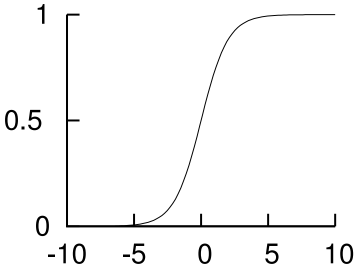
$$\begin{array}{rcl} y_1 & = & g(x_1 + w_1) \\ y_2 & = & g(x_2 + w_2) \end{array} \tag{19}$$

where $w_1$ and $w_2$ are parameters and, for example,

$$g(u) = \frac{1}{1 + e^{-u}} \tag{20}$$

so $g : (-\infty, \infty) \to (0, 1)$.

$$\tag{21}$$



Now we have

$$\mathbf{s} \overset{\text{mixing}}{\longrightarrow} \mathbf{r} = M\mathbf{s} \overset{\text{unmixing}}{\longrightarrow} \mathbf{x} = W\mathbf{r} \overset{\text{non-linearity}}{\longrightarrow} \mathbf{y} : y_a = g(x_a + w_a) \tag{22}$$

For later notational convenience, let write

$$y_a = g(x_a + w_a) = f(r_1, r_2; W, w_a) \tag{23}$$

where $f$ is the function, parameterized by $W$ and $w_a$, mapping from the recording to $y$. So reason the saturating non-linearity will help is that with a non-linearity like this a large enough scaling will reduce the entropy: consider $H(g(\lambda X))$ where $\lambda$ is a constant. If $\lambda$ is very large it will spread $X$ out so that it will take lots of very big positive or negative values; if $\lambda x$ is a big positive number then $g(\lambda x)$ will be near to one, conversely, if it is a large negative number, $g(\lambda x)$ will be near to zero: a distribution that is often zero and often one is not very entropic.

Before considering the source separation problem, lets look at the effect of the non-linearity on its own: we consider the one-to-one case

$$r \overset{\text{multiply}}{\longrightarrow} x = Wr \overset{\text{non-linearity}}{\longrightarrow} y = g(x + w) = f(r; w, W) \tag{24}$$

where $W$ and $w$ are now both scalars and $r$, $x$ and $y$ are outcomes for random variables $R$, $X$ and $Y$. We consider maximizing the entropy $H(Y)$. What does this do; well, it maximizes the information in $Y$ about $R$:

$$I(R; Y) = H(Y) - H(Y|R) \tag{25}$$

but $H(Y|R)$ is constant since $R$ determines $Y$. In the discrete case we are familiar with this would be easy to discuss, $H(Y|R)$ would be zero, in the continuous case it is not that simple, it is actually minus infinity, but, the consequence is the same, it does not depend on $W$ and $w$. To maximize $H(Y)$ we need to calculate its derivative with respect to the parameters $W$ and $w$; this is a feature of the algorithm, ultimately we need to calculate derivatives and these are calculable and well defined, even if the quantity being differentiated is not. In this case

$$H(Y) = -\int p(y)\log p(y)dy \tag{26}$$

and this is estimated by

$$\tilde{H}(y) = -\log p(y) \tag{27}$$

In other words, if $n$ values $y_i$ is drawn from $Y$ then

$$\frac{1}{n}\sum_i \tilde{H}(y_i) \to H(Y) \tag{28}$$

as $n$ gets large.

Of course we do not have $p_Y(y)$ and it would be difficult to estimate, but, it turns out we do not need it to get the derivative of $H\tilde{(}y)$. We have seen already that since $y = f(r; W, w)$

$$p_Y(y) = \frac{p_R[r = f^{-1}(y)]}{|f'(f^{-1}(y)|} \tag{29}$$

so

$$\tilde{H}(y) = -\log p_R(r) + \log|f'| \tag{30}$$

and $p_R(r)$ is independent of the parameters. Now, for our choice of saturating non-linearity

$$\begin{aligned} g(u) &= \frac{1}{1+\exp(-u)} \\ \frac{dg}{du} &= g(1-g) \end{aligned} \tag{31}$$

and hence

$$\log|f'| = \log W + \log f + \log(1-f) \tag{32}$$

Now we know $f$:

$$f = g(Wr + w) \tag{33}$$

so

$$\frac{df}{dW} = rf(1-f) \tag{34}$$

and hence,

$$\frac{d\tilde{H}(y)}{dW} = \frac{1}{W} + \frac{1}{f}rf(1-f) - \frac{1}{1-f}rf(1-f) = \frac{1}{W} + r(1-2y) \tag{35}$$

Similarly

$$\frac{d\tilde{H}(y)}{dw} = 1 - 2y \tag{36}$$

These quantities: $s$, $y$ and, of course, $W$, are numbers we have access to, $W$ is a parameter, $s$ is the signal, we can sample $s(t)$ at a set of times to get a set of $s$'s and $y$ is a function of $s$. This means we can estimate these derivatives, giving the gradient of $H$ at a point $(W, w)$ in the parameter space. This is a common situation in numerical optimization, we don't know the function and, here, it isn't even so easy to define, but we do know its gradient. This means that locally we know which direction brings us in the direction of greater $H$ and so numerical hill-climbing routines can be used, these are well described in, for example, Press et al. (2007): steepest ascent, conjugate gradient or metric gradient methods work well here.[1]

The idea then is to chose a starting $W$ and $w$; estimate the gradient and then change $W$ and $w$ a small amount, repeating until the optimum values are found. What would the optimum value look like, well we know

$$p_Y(y) = \frac{p_R(r)}{|f'(r)|} \tag{37}$$

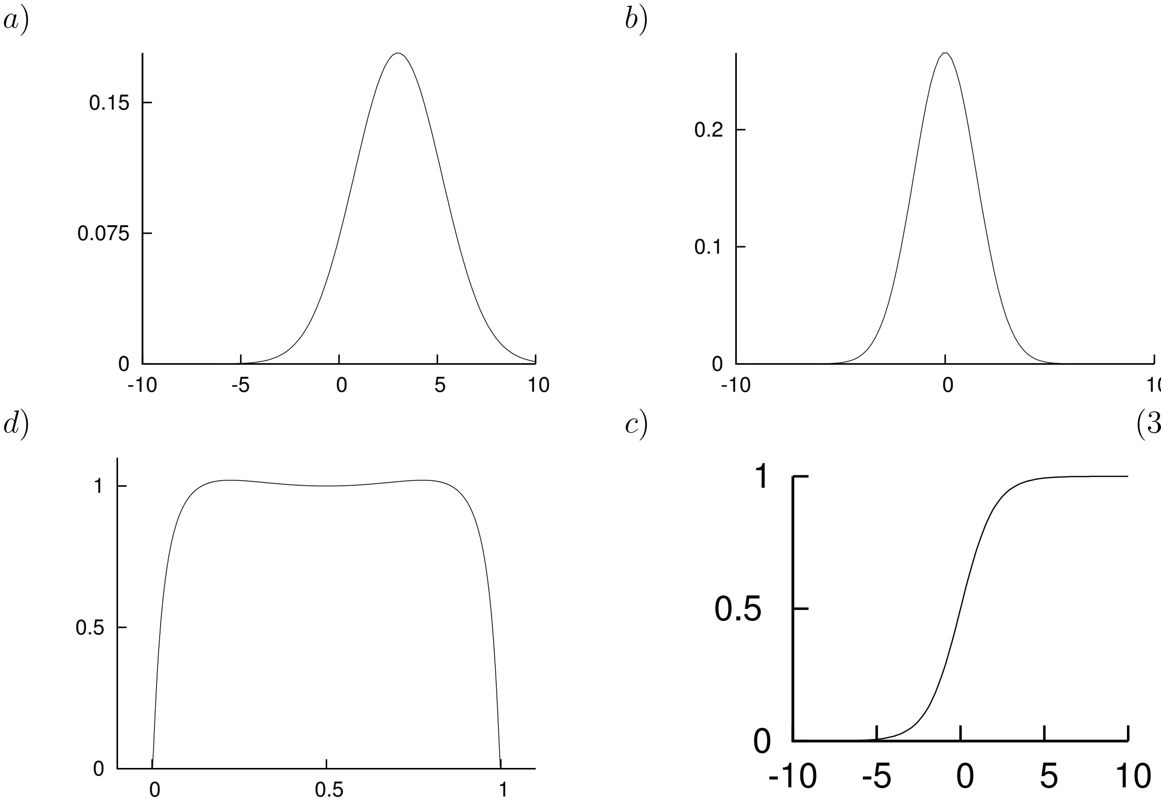
with $r = f^{-1}(y)$. Hence, if $Y$ is evenly distributed on its interval $(0, 1)$ and $f'(r)$ is always positive then

$$f(r) = \int_{-\infty}^{r} p_R(u)du \tag{38}$$

Now this is not what we do, we do not know $p_R(r)$ and we chose $f(r)$ at the start, here it a member of a two-parameter family of functions parameterized by $W$ and $w$. However, ideally, if the derivative of the saturating non-linearity

[1] The metric method uses a slightly surprising, but effective, choice of metric and is defined in Amari (1998).

is somewhat close to the distribution of $R$ then infomax will find the $W$ and $w$ that line everything up so that $Y$ will have something close to an even distribution. Here is an example:



(39)

Figure a) shows an initial distribution

$$p_R(r) = \frac{1}{\sqrt{10\pi}} e^{(r-3)^2/10} \tag{40}$$

b) is a new distribution arrived at by shifting a rescaling $r$ using $W$ and $w$:

$$p_U(u) = \frac{1}{\sqrt{4.5\pi}} e^{u^2/4.5} \tag{41}$$

where $u = Wr + w$. Now, this distribution is all lined up with the rectifying non-linearity c) so that $p_Y(y)$ where $y = g(u)$ is d).

Now, back to the two-to-two case:

$$\mathbf{s} \xrightarrow{\text{mixing}} \mathbf{r} = M\mathbf{s} \xrightarrow{\text{unmixing}} \mathbf{x} = W\mathbf{r} \xrightarrow{\text{non-linearity}} \mathbf{y} : \; y_a = g(x_a + w_a) \tag{42}$$

and here we want to maximize $H(Y_1, Y_2)$; the idea being that this should find a matrix $W$ whose eigen-directions give statistically independent $Y_a$, this is the bit we want since it will also make the $X_a$ independent, and whose eigenvalues, along with the values of $w_a$ make $H(Y_1)$ big by lining the saturating non-linearity up with the underlying distributions: the orientation of $W$ deals with the unmixing, the scale of $W$ and the vector of $w_a$s deals with the one-to-one part. Anyway, doing the calculation gives

$$\begin{aligned}\frac{d\tilde{H}(y)}{dW_{ab}} &= (W^T)^{-1}_{ab} + r_a(1-2y_b)\\ \frac{d\tilde{H}(y)}{dw_a} &= 1-2y_a \qquad (43)\end{aligned}$$

allowing the maximum of $H(Y_1, Y_2)$ to be, hopefully, found and this, again, hopefully, will unmix the signal. Note that this algorithm is not as blind as we might of hoped, the non-linearity needs to be chosen judiciously: however, the algorithm is reasonably robust; reasonably successful and certainly interesting mathematically.

# References

Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276.

Bell, A. J. and Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159.

Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks*, 10(3):626–634.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press.