Methods of Data Analysis

Introduction to information theoretic quantities

Week 6

1 Motivation

Pairwise correlation function only captures linear statistical dependence, but misses higher-order dependancies between pairs of variables. In fact, two random variables can be perfectly dependent (i.e., deterministically related), yet have a zero cross-correlation. Information theory defines a unique measure of statistical dependence, the mutual information, which is zero if and only if a pair of random variables is statistically independent. The measure also generalizes to statistical dependence between more than two variables, and related quantities are able to systematically isolate effects of intrinsically pairwise, triplet, quadruplet, etc. interactions between random variables. On the downside, estimating information-theoretic quantities from limited data is notoriously hard, because the naive estimator is biased, and schemes for removing the bias do not universally work.

Thinking about (pairwise) covariance has given rise to useful dimensionality reduction techniques and transformations of the data (most directly, to principal component analysis and whitening, which linearly rotates and scales the data such that the transformed variables have unit covariance matrix). Similarly, thinking about mutual information gave rise to more powerful ideas where the data is linearly transformed such that the resulting variables are as statistically independent as possible (in independent component analysis, or ICA).

2 Goals

- Define KL and JS divergences, mutual information, multi-information, redundancy / synergy, information decomposition.
- Mutual information for Gaussian variables and Gaussian processes.
- Estimating entropy and mutual information from data (direct estimator, nearest-neighbor, NSB).
- Introduction to independent component analysis (ICA).

3 Data

- A subset of gene expression studies from Ref [1]: a matrix of 200 genes by 173 conditions, where each value represents log fold-change in a gene expression level (microarray) in yeast, under a specific condition relative to the control condition.
• Ruderman natural image dataset from Week 1.

• A draw of $2^{16}$ 6-bit samples from an unknown distribution, saved in data.mat.

4 Quantities, definitions, details

**Entropy.** Entropy of the distribution is $S[P] = -\sum_x P(x) \log_2 P(x)$. Log base 2 gives the answer in “bits”, natural log in “nats”. For continuous distributions, $S[P] = -\int dx \, P(x) \log_2 P(x)$. In this case, one has to be careful because of the issues of measure (and for physicists, units). For instance, for a Gaussian, $G(x) = (2\pi\sigma^2)^{-1/2} \exp(-x^2/\sigma^2)$, the entropy is $S[G] = \frac{1}{2} \log_2(2\pi e\sigma^2)$, which has somewhat nonintuitive behavior as $\sigma \to 0$. Another way to view the issue with the entropy of continuous distributions is by approximating a continuous distribution with a sequence of discrete distributions defined on bins of size $\Delta$, and seeing what happens to the limit of the discrete entropy as $\Delta \to 0$ (answer: $S[p] = S[P] + \log_2 \Delta$, where $p$ is the continuous probability distribution function, and $P(x_i) = p(x_i)\Delta$, such that $\sum_i P(x_i) = 1$ is the discrete approximation to $p$). For a continuous distribution, a better suited quantity is the relative entropy or the KL-divergence, see below. Nevertheless, entropy differences are well-behaved because the log $\Delta$ terms subtract out and there are no divergences in bin size (it is interesting to note that this is similar to classical thermodynamics, where only entropy differences make sense). Mutual information and other information measures are often expressible as entropy differences, and thus they can safely be used on discrete or continuous distributions. Interpretation-wise, entropy of a distribution is a measure of the diversity of values the random variables can take (highest entropy = uniform distribution; zero entropy = delta function). Another way to view the entropy is as a measure of average surprise; $-\log p_i$ is the surprise of symbol $i$ (that is, rare symbols are more surprising when seen).

**Divergences.** The most important divergence measure between a pair of distributions is the Kullback-Leibler divergence, defined as $D_{KL}(p||q) = \int dx \, p(x) \log_2 \frac{p(x)}{q(x)}$. This is a positive quantity, measured in bits, and also known as relative entropy. It can be understood as an entropy in $p$, relative to some prior distribution $q$. KL-divergence is well-behaved for continuous distributions. It is often used as a distance measure between distributions $p$ and $q$, since $D_{KL} = 0$ if and only if $p = q$. Strictly speaking, the distance interpretation is incorrect, since the KL divergence is neither symmetric nor does it satisfy the triangle inequality. Nevertheless, $D_{KL}$ often comes up in theoretical derivations, and can actually be used as a basic quantity from which other information theoretic quantities can be derived. Interpretation-wise, $D_{KL}$ is the number of additional bits needed to code a random variable $x$, if $x$ is really drawn from $p$, but the code was designed for $x$ drawn from $q$. Jensen-Shannon divergence is a normalized (bounded between 0 and 1), symmetric “distance” measure between two distributions, defined as $D_{JS}(p,q) = 0.5(D_{KL}(p||r) + D_{KL}(q||r))$, where $r = 0.5(p+q)$.

**Mutual information.** Given a joint distribution over two random variables, the mutual information between them is $I(x; y) = \sum_{x,y} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)}$, where $P(x)$ and $P(y)$ are the marginals of the joint. Note that $I$ is a scalar, a functional of $P(x,y)$ (despite the traditional notation, it is not a function of $x$ or $y$). $I$ is positive, in bits, and is a measure of the total amount of statistical dependence between $x$ and $y$ ($I = 0$ only when they are independent, i.e., $P(x,y) = P(x)P(y)$). You can see that $I(x; y) = D_{KL}(P(x,y)||P(x)P(y))$, that is, the mutual information is the divergence (“distance”) between the joint and the factorial distributions. $I$ is symmetric, $I(x; y) = I(y; x)$. It is also reparametrization invariant, so $I(x; y) = I(f_2(x); f_3(y))$. 
where $f_x$ and $f_y$ are invertible functions of their arguments. This is very important for data applications, since many correlation measures don’t share this invariance (e.g., if you compute correlation coefficient between $x$ and $y$, or $\log(x)$ and $\log(y)$, you get a different answer). Shannon has shown that $I$ is the only (up to the unit of measure) quantity consistent with certain basic postulates that an information measure should have (e.g., such as that independence of information add). Mutual information can be rewritten as $I(x; y) = S[P(y)] - \langle S[P(y|x)] \rangle_{P(x)}$, the difference between the uncertainty in the $y$ (total entropy) and the average uncertainty that remains when $x$ is known (noise entropy). Because entropies are positive, $I(x; y) \leq S[P(y)]$ and due to symmetry, $I(x; y) \leq S[P(x)]$. Interpretation-wise, $2I(x;y)$ is the approximate number of distinguishable levels or states in $y$ accessible by changing $x$ given noise. If $x$ and $y$ are jointly Gaussian with a correlation coefficient $c$, then the mutual information $I(x; y) = -0.5 \log_2(1-c^2)$. If $x$ and $y$ are vectors whose distribution is jointly Gaussian, then the information is

$$I(x; y) = \frac{1}{2} \log_2 \left( \frac{\det C_{xx} \det C_{yy}}{\det C} \right),$$

where $C_{xx}$ and $C_{yy}$ are the covariance matrices of the marginal distributions $P(x)$ and $P(y)$, respectively; and $C$ is the covariance matrix of the joint Gaussian. Note that this formula can also be used to estimate the mutual information between two jointly Gaussian time series.

Multi-information and connected information quantities. Consider a joint distribution over several variables, $P(x_1, x_2, \ldots, x_N)$. Multi-information measures the total amount of statistical correlation between all the variables, and is defined as $I = S[P(x_1, \ldots, x_N)] - S[P(x_1)P(x_2) \ldots P(x_N)]$, i.e., the difference in entropies of the joint and the factorial distributions. There exists a unique decomposition of the total multi-information into positive contributions, $I = I^{(2)} + I^{(3)} + \ldots + I^{(N)}$, where $I^{(k)}$ is called the connected information of order $k$, and it measures the fraction of total correlations contributed by the irreducible interactions between exactly $k$ variables. The construction of this decomposition is difficult in practice and rests on constructing a series of approximating distributions to the joint (called the maximum entropy distributions consistent with correlation of up to order $k$), but this is a principled way of assessing at which order the variables interact (i.e. if $I^{(2)}$ accounts for a large fraction of total $I$, pairwise interactions between variables are sufficient to explain all the statistical structure in the joint distribution). For details on this decomposition, see Ref [3].

Redundancy and synergy. Consider variables $x$ and $y$ that both carry some information about the input, $s$. For a concrete example, let $x$ and $y$ be responses of two neurons that both respond to the common stimulus $s$. Given the quantities we already defined, we can ask how much information each of the neurons carries about the stimulus, and that is quantified by the mutual information, $I(x; s)$ and $I(y; s)$. From the two responses, we can form a new quantity, $z = [x, y]$, and ask how much information do both responses combinatorially encode about $s$, that is, what is $I(x, y; s) = I(z; s)$. An interesting quantity is redundancy, $R(x, y; s) = I(x; s) + I(y; s) - I(x, y; s)$. If $R > 0$, the neurons are redundant, that is, they encode overlapping information such that the sum of their individual informations is larger than the combinatorial response they carry. Suppose the response of the two neurons would be exactly identical. Then $I(x, y; s) = I(x; s) = I(y; s)$ and thus $R = I(x; s) > 0$. In contrast, $R < 0$ signifies synergistic (combinatorial) coding. In the extreme case, it could be true that $I(x; s) = I(y; s) = 0$, while $I(x, y; s) \neq 0$. Can you think of such an example?

Direct estimation of entropy and mutual information. The plug-in (or naive, or maximum-
likelihood) estimator of entropy is $S[P] = -\sum_i \hat{P}_i \log_2 \hat{P}_i$, where $\hat{P}_i = n_i / \sum_i n_i$ is the empirical frequency at which state $i$ occurs in a sample of $\sum_i n_i$ observations. This estimator is always negatively biased. To see that, consider true probabilities, $P_i$, and write $\hat{P}_i = P_i + \delta P_i$, where the error $\delta P_i$ is the small sample noise in the empirical estimate of probability. By expanding the entropy functional, show that $\langle S_{\text{naive}} \rangle = S - \langle D_{KL}(\hat{P}||P) \rangle$, where $\langle \cdot \rangle$ is an expectation over random sampling errors. Since the Kullback-Leibler divergence is always positive, this equation implies that the naive estimate always underestimates the true entropy. In most real-life situations, this bias (rather than the variance in the estimate because of the small sample effect) is the dominant problem that estimators have to deal with. In fact, often the number of total observations, $N = \sum_i n_i$ might be much smaller even that the total number of possible states $K$ of the system (it would seem that in such a situation, no estimator can ever do the job, but this really depends on the underlying probability distribution).

The “direct” estimate is based on the observation that the bias term scales inversely with $N$, the total sample size, so that $\langle S_{\text{naive}} \rangle = S_{\text{true}} + A/N + \cdots$. As a result, one can take many random subsets of fractions of data, e.g., $M(f_1)$ random choices of $f_1 N$ observations, where $f_1$ is the fraction between 0 and 1, $M(f_2)$ random choices of $f_2 N$ observations etc. For each random choice of observations, one estimates the empirical distribution, and the resulting naive estimate of entropy, to get $\langle S_{\text{naive}}(f) \rangle$, the average (over random selections of observations) naive estimate at a given data fraction. This can be plotted versus the inverse data size, $1/(fN)$, and linearly extrapolated to infinite data size (i.e., towards zero). This extrapolation removes the bias and extends the range of validity of the estimation beyond the point where naive estimator would already give strongly unreliable results. For a specific example relating to neural code, see Ref [5] which introduced the direct method.

Information, as a difference of two entropies, also has an inverse-$N$ bias, and can be debiased in a very similar way. A direct estimator for the information has been presented in Ref [9]; follow this reference to implement the mutual information estimator for continuous data. The key point here is that continuous data need to be discretized, but the “proper” number of levels is a priori unknown. More levels allow for capturing finer features in the joint distribution, but given the finite data sample, also leads to increasingly large bias problems. Given the data, the task is thus to select the maximal possible binning level such that the debiasing is still reliable, while information is not lost due to discretization.

**Independent component analysis (ICA).** You are given a set of $N$-dimensional vectors, $\mathbf{x}^t$ that constitute data, and the samples are enumerated by index $t = 1, 2, \ldots, T$. Eigensystem decomposition (or Principal Component Analysis) first forms the covariance matrix $\mathbf{C}$, given by $C_{ij} = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle$, where the indices $(i,j)$ range over the $N$ components of the vectors, and the brackets denote averaging over $T$ samples. Then, one spectrally decomposes the covariance, $\mathbf{C} = \mathbf{V} \mathbf{D} \mathbf{V}^T$, where the diagonal matrix $\mathbf{D}$ gives the variances in the principal directions, determined by the columns of the orthonormal matrix $\mathbf{V}$. If only a small number, $n \ll N$, of the largest variances accounts for most of the total variance (given by the trace of $\mathbf{D}$), one has a suggestion that the true dimensionality of data is $\sim n$, since in other directions the data is “flat.” This dimensionality reduction (or its SVN variant), where only the important eigendirections are retained, is the workhorse of preprocessing and dimensionality reduction algorithms. If one transforms the original data as follows, $\tilde{\mathbf{x}} = \mathbf{V} \mathbf{D}^{-1/2} \mathbf{V} \mathbf{x}$, the new data vectors (denoted by tilde) are whitened, i.e., they have unit covariance matrix (and if their generating distribution were an arbitrary Gaussian in the first place, now it is a unit Gaussian). Note that whitening doesn’t do anything to higher-order moments / correlations, which for Gaussian distribution are zero or trivially expressible in terms of the mean and the covariance. Whitening thus rotates (i.e.,
linearly transforms) the data into a system where pairwise correlations are removed.

Is there a (still linear) transformation which will remove also higher-order correlations that are left intact by whitening? This is the task of independent component analysis. Suppose we write $s = Wx$ (where $x$ are usually already whitened data which have no pairwise correlation). The task is to find such a matrix (not necessarily square, and not necessarily orthogonal!) $W$, that the transformed vectors $s$ are as statistically independent as possible, i.e., have as low mutual information as possible. The idea is that real data might be well described by underlying “causes” or signals that are independent of each other (such as the sound waveform of two people in a room talking about different things simultaneously), but the observed signal is a mix of the two (for instance, a sum arriving at a microphone, or at two stereo microphones). The question that ICA addresses is whether, given only the observed signal, it is possible to “unmix” it into independent sources (also known as a “source separation” or “blind convolution” problem).

Another way to view ICA is in “sparse coding.” Here one can view the observed signal as a linear superposition of a dictionary of possible “basic functions.” The dictionary can have a large number of basis functions (i.e., it can be overcomplete), and one way in which the signals can become statistically very independent is that they become very sparse, that is, most of the unmixed components are zero at every point in time (or for every sample), and only a few ones – different from sample to sample – are active. Sparse coding has offered a productive characterization of statistics beyond pairwise correlations (that you addressed in Week 1) in natural scenes, by extracting a dictionary of “independent components”, which look like small oriented edge segments of varying thickness [10, 11]. Interestingly, the primary visual cortex in vertebrate vision responds to exactly such features of natural scenes.

5 Study literature

- For formal definitions of information theoretic quantities, see a classic book on information theory, Ref [2].

- For Independent Component Analysis, and the Matlab implementation of the FastICA algorithm, see the web page of Hyvärinen, one of the theorists behind ICA, at http://www.cs.helsinki.fi/u/ahyvarin/whatisica.shtml.

- Direct entropy estimation, Ref [5]; Nemenman-Shafee-Bialek (NSB), one of the best performing entropy estimators for discrete distributions, Ref [6]; and the recent advance on it, Ref [7]. Entropy estimation without binning for continuous distributions, Ref [4]. An interesting bound-on-entropy estimator, but the paper contains a great overview of various estimators, Ref [8].

6 Homework

1. Implement a direct estimator for entropy for a discrete distribution (i.e., debias the estimate by extrapolating to infinite sample size). Use samples from data.mat (which you are going to use also in a subsequent problem). Randomly choose $2^3, 2^4, 2^5, \ldots, 2^{16}$ samples from the data, and estimate (i) the entropy using your direct estimator; (ii) the entropy in a naive way, by counting the frequency of patterns and plugging in into the entropy formula. When you run the direct estimator which extrapolates naive entropy to infinite sample size, show the extrapolation plot for the run with a total of $2^{10}$ samples and for the run with a total of $2^{16}$ samples (i.e., plot the estimated naive entropy as a function
of 1/(sample size), and the best fit linear extrapolation extending to 0 on the x-axis, to find the y-intercept). Do you see the expected scaling behavior for the small sample correction? Show the entropy for both methods as a function of the number of samples. Optional: download and use the NSB estimator ([6], or its improved version, [7]) and compare its performance with the direct estimation (if you work with discrete distributions, these estimators may anyway come in handy).

2. Implement the direct estimator for information between continuous variables, following [9]. First, to check that everything works properly, let’s generate synthetic data for which we know the correct answer: pairs of jointly Gaussian distributed random variables with given correlation coefficients, equal variance, and zero mean. Generate random samples for 20 such pairs with correlation coefficients $c$ going uniformly from $-0.95$ to $0.95$ in equal increments. Think carefully and explain how you generate pairs of normally distributed random variables with a desired correlation coefficient. For each pair, draw 10, 100, 1000, 10000 samples, estimate the mutual information $I$ using the direct estimator. Plot the estimated $I$ vs the correlation coefficient for all pairs and for all 4 sample sizes (different sample sizes using different colors / plot symbols), and compare on the same plot with the analytical curve $I = -0.5 \log_2 (1 - c^2)$. Do you have an intuitive explanation what it means that at finite correlation coefficient you can have a diverging mutual information?

3. Next, use the direct estimator to analyze Gasch data on gene expression. If your algorithm is fast enough, compute the mutual information for all pairs of genes, otherwise, randomly select 1000 pairs and estimate the information for them. Let your estimator output both the corrected (debiased) information value as well as the naive value, and make a scatter-plot of the corrected vs naive value. How big is the small sample bias for this dataset? What is the typical (median / mean) value of information across the analyzed pairs? What is the maximal value that you observe? Beyond factors that are intrinsic to the biological system under study, what factors (that have nothing intrinsic to do with the cell) influence the magnitude of the observed numbers? Plot the information values for all pairs on y axis, vs the correlation coefficient for those pairs, on the x axis, and compare with the predicted curve for jointly Gaussian variables. Find several pairs that deviate the most for the curve: (i) pairs that maximal information given a very small correlation coefficient, e.g., $|c| < 0.05$; (ii) pairs that have a very small amount of mutual information yet have a large correlation coefficient, e.g., $|c| > 0.6$. For several of these pairs $(g_i, g_j)$, scatter plot the data of $g_i$ vs $g_j$ (each point is a joint expression level in a given condition). Can you propose a few suggestions about the data that explain the particular deviations you see?

4. Analyze the 6-bit binary patterns in data.mat which were generated by drawing IID from an unknown distribution. Figure out the rule that generated the patterns and, if you can, write down the generating distribution. Hint: try to analyze the statistical structure order-by-order. You can start by computing the pairwise correlation coefficient between the variables, and the mutual information between all pairs. Suggest and carry out further analyses to determine the statistical structure in the distribution.

5. Optional. Compare principal components and independent components for a natural image dataset. To this end, take the Ruderman set of 45 natural images, and extract from it image patches of size $8 \times 8$ pixels (suggestion: sample the patches to be 4 pixels apart, so that they overlap by a half). Represent these patches as 64 dimensional vectors, such that your matrix of samples will be $64 \times \text{(num. of patches)}$. Extract the principal components
and plot them as $8 \times 8$ images ordered by decreasing contribution to the total variance. Run an ICA algorithm (e.g., FastICA, obtainable from the Hyvärinen webpage above) to extract independent components (perhaps you need to explore the algorithm’s parameters a bit for a good result), and show them. Comment on the comparison between the two types of components—in what important qualitative ways do they differ? Can you explain the appearance of the principal components?

References


