Goals

Review the basics of probability distributions

Images and scenes

Graphical models for inference

Introduction to tasks

Probability Overview

Random variables, discrete probabilities, probability densities, cumulative distributions

Discrete: random variable X can take on a finite set of discrete values

\[ X = \{x(1), \ldots, x(N)\} \]

\[ \sum_{i=1}^{N} \eta_i = \sum_{j=1}^{N} p(X = x(j)) = 1 \]

Densities: X takes on continuous values, x, in some range.

Density: \( p(x) \)

\[ \text{prob}(x < X < x + dx) = \int_x^{x+dx} p(x) \, dx \]

\[ \text{prob}(x < X < x + dx) \approx p(x) \, dx \]

\[ \int_{-\infty}^{\infty} p(x) \, dx = 1 \]

Cumulative distribution:

\[ \text{prob}(X < x) = \int_{-\infty}^{x} p(X) \, dX \]

Densities of discrete random variables

The Dirac Delta function, \( \delta[\cdot] \), allows us to use the mathematics of continuous distributions for discrete ones, by defining the density as:

\[ p[x] = \sum_{i=1}^{N} p[i] \delta[x - x[i]] \]

where \( \delta[x - x[i]] = \infty \) for \( x = x[i] \)

Think of the delta function, \( \delta[\cdot] \), as \( \epsilon \) wide and \( 1/\epsilon \) tall, and then let \( \epsilon \rightarrow 0 \), so that:

\[ \int_{-\infty}^{\infty} \delta(y) \, dy = 1 \]

The density, \( p[x] \), is a series of spikes. It is infinitely high only at those points for which \( x = x[i] \), and zero elsewhere. But "infinity" is scaled so that the local mass or area around each point \( x[i] \), is \( p[i] \).

Joint probabilities

\[ \text{Prob}(X \text{ AND } Y) = p(X, Y) \]

Joint density: \( p(x, y) \)

Three basic rules of probability

Rule 1: Conditional probabilities from joints: The product rule

Probability about an event changes when new information is gained.

\[ \text{Prob}(X \text{ given } Y) = p(X|Y) \]

\[ p(X) \]

\[ p(X \text{ given } Y) \]

\[ p(X) \]

\[ p(X, Y) = p(X|Y) \]

\[ p(Y) \]

The form of the product rule is the same for densities as for probabilities.
Rule 2: Lower dimensional probabilities from joints: The sum rule (marginalization)

\[ p(X) = \sum_{Y} p(X, Y) \]
\[ p(x) = \int_{-\infty}^{\infty} p(x, y) \, dy \]

Rule 3: Bayes' rule

From the product rule, and since \( p(X,Y) = p(Y,X) \), we have:

\[ p(Y|X) = \frac{p(X|Y) p(Y)}{p(X)} \]

Bayes Terminology in visual perception

\[ p(S | I) = \frac{p(I | S) p(S)}{p(I)} \]

Usually, we will be thinking of the \( Y \) term as a random variable over the hypothesis space, and \( X \) as data. So for visual inference, \( Y = S \) (the scene), and \( X = I \) (the image data), and \( I = f(S) \).

We'd like to have:

- \( p(S|I) \) is the posterior probability of the scene given the image
- i.e. what you get when you condition the joint by the image data. The posterior is often what we'd like to base our decisions on, because as we discuss below, picking the hypothesis \( S \) which maximizes the posterior (i.e. maximum a posteriori or MAP estimation) minimizes the average probability of error.

\( p(S) \) is the prior probability of the scene.

\( p(I|S) \) is the likelihood of the scene. Note this is a probability of \( I \), but not of \( S \).

Independence

Knowledge of one event doesn't change the probability of another event.

\[ p(X) = p(X|Y) \]
\[ p(X,Y) = p(X)p(Y) \]

Density mapping theorem

Discrete random variables

No change to probability function.

Continuous random variables

Form of probability density function does change because we require \( p(x)dx = p(y)dy \)

Suppose, \( y = f(x) \)

\[ p_Y(y) \delta y = p_X(x) \delta x \]

\[ p_Y(y) = \int \delta(y - f(x)) f^{-1}(x) p_X(x) \, dx \]

over each monotonic part of \( f \).
Statistics

- **Expectation & variance**

  **Definition of expectation or average:**
  \[ \text{Average}[X] = X - E[X] = \sum x[i] \cdot p[x[i]] \]
  \[ \mu = E[X] = \int x \cdot p(x) \cdot dx \]

  **Some rules:**
  \[ E[X+Y]=E[X]+E[Y] \]
  \[ E[aX]=aE[X] \]
  \[ E[X+a]=a+E[X] \]

  **Definition of variance:**
  \[ \sigma^2 = \text{Var}[X] = E[(X-\mu)^2] = \sum_{i=1}^{N} (p(x(i))(x(i) - \mu))^2 = \sum_{i=1}^{N} p(x(i) - \mu)^2 \]
  \[ \text{Var}[X] = \int (x-\mu)^2 \cdot p(x) \cdot dx \]

  **Standard deviation:**
  \[ \sigma = \sqrt{\text{Var}[X]} \]

  **Some rules:**
  \[ \text{Var}[X] = E[X^2] - E[X]^2 \]
  \[ \text{Var}[aX] = a^2 \cdot \text{Var}[X] \]

- **Covariance & Correlation**

  **Covariance:**
  \[ \text{Cov}[X,Y] = E[(X-\mu_X)(Y-\mu_Y)] \]

  **Correlation:**
  \[ \rho_{x,y} = \frac{\text{Cov}[X,Y]}{\sigma_x \sigma_y} \]

- **Independent random variables**

  \[ E[X \cdot Y] = E[X] \cdot E[Y] \]
  \[ \text{Cov}[X,Y] = \rho_{x,y} = 0 \]
  \[ \text{Var}[X+Y] = \text{Var}[X] + \text{Var}[Y] \]

---

**Degree of belief vs., relative frequency**

What is the probability that the Vikings will win the Superbowl in 2001? Assigning a number between 0 and 1 is assigning a degree of belief. These probabilities are also called subjective probabilities. "Odds", determine subjective probabilities, where the "odds of x to y" means probability = \( \frac{x}{x+y} \).

What is the probability that a coin will come up heads? In this case, we can do an experiment. Flip the coin \( n \) times, and count the number of heads, say \( h[n] \), and the set the probability, \( p = \frac{h[n]}{n} \) -- the relative frequency. Of course, if we did it again, we may not get the same estimate of \( p \). One solution often given is:

\[ p = \lim_{n \to \infty} \frac{h[n]}{n} \]

A problem with this, is that there is no guarantee that a well-defined limit exists.

In visual inference, we can at least in principle, measure statistics, and model probabilities of both image and scene variables. So the relative frequency interpretation seems reasonable. In practice, the dimensions of vision problems are so high, that it is impractical to do this. Once we use the statistical framework to model perception, say of a particular shape from shading information, then probabilities are more like "subjective unconscious beliefs".

**Principle of insufficient reason**

- **Principle of symmetry**

  Suppose we have \( N \) events, \( x[1],x[2],x[3],...,x[N] \) that are all physically identical except for the label. The we assume that \( \text{prob}(x[1]) = \text{prob}(x[2]) = \text{prob}(x[3]) = \text{prob}(x[N]) = \frac{1}{N} \).

  In other words, if we have no additional information about the events, we should assume that they are uniformly distributed.

- **Information theory and Maximum entropy**

  Information theory provides a powerful extension to the principle of symmetry. Information of event \( X \) is:

  \[ \text{Information}[X] = - \log_2(p(X)) \]

  Using the definition of expectation above, we can specify the expectation of information, which is called entropy. Entropy of a random variable \( X \) with probability distribution \( p(X) \) is:

  \[ H(X) = \text{Average}[\text{Information}[X]] = - \sum_{x} p(X) \log_2(p(X)) \]

  It can be shown that out of all possible probability distributions, \( p(X) = \frac{1}{N} \), is biggest for the uniform distribution. \( p(X) = \frac{1}{N} \). It turns out that a more powerful formulation of the principle of symmetry is maximum entropy. For example, out of all possible probability distributions of a random variable with infinity range, but with a specific mean and standard deviation, the Gaussian is unique in having the largest entropy. If the range goes from zero to infinity, and we know the mean, the maximum entropy distribution is an exponential (Cover and Thomas).

  An interesting application of the maximum entropy principle is to learning image textures joint probabilities: \( p(I[1],...,I[N]) \).
where \( N \) is very big, but where one has only a relatively small number of measured statistics relative to the number of possible images (which is really huge). The measurements underdetermine the dimensionality of the probability space—i.e. there are many different probability distributions which give the same statistics. So the principle of symmetry, or insufficient reason, says to choose the one with the maximum entropy.

Graphical Models of dependence

Graphs: causal structure and conditional independence

In general, natural image pattern formation is specified by a high-dimensional joint probability, requiring an elaboration of the causal structure that is more complex than the simple SDT model. The idea is to represent the probabilistic structure of the joint distribution \( P(S,L,I) \) by a Bayes net (e.g. Ripley, 1996), which is a graphical model that expresses how variables influence each other. There are just three basic building blocks: converging, diverging, and intermediate nodes. For example, multiple (e.g. scene) variables causing a given image measurement, a single variable producing multiple image measurements, or a cause indirectly influencing an image measurement through an intermediate variable. These types of influence provide a first step towards modeling the joint distribution and the means to compute probabilities of the unknown variables given known values.

Components of the generative structure for image patterns involve converging, diverging, and intermediate nodes. For example, these could correspond to multiple (scene) causes (shape \( S_1 \), illumination \( S_2 \) giving rise to the same image measurement \( I \); one cause, \( S \) influencing more than one image measurement, a single variable producing multiple image measurements, or a cause indirectly influencing an image measurement through an intermediate variable \( L \) (3D shape).

The arrows tell us how to factor the joint probability into conditionals. So for the three examples above, we have:

\[
\begin{align*}
p(S_1,S_2,I) &= p(I|S_1,S_2)p(S_1)p(S_2)
p(S,I_1,I_2) &= p(I_1|S)p(I_2|S)p(S)
p(S,L,I) &= p(I|L)p(L|S)p(S)
\end{align*}
\]

Primary, secondary variables.

The following figure draws a parallel between the causal structure, as determined by the generative model, for signal detection theory (as in the light detection problem), and the general problem of visual inference.

We can interpret the causal structure in terms of conditional probability. The top panel shows one possible generative graph structure for an ideal observer problem in classical signal detection theory (SDT). The data are determined by the signal hypotheses plus (additive gaussian) noise. Knowledge is represented by the joint probability \( p(x,H,n) = p(x|H,n)p(H)p(n) \). The lower panel shows a simplified example of the generative structure for perceptual inference from a pattern inference theory perspective. The image measurements \( x \) are determined by a typically non-linear function \( \phi \) of primary signal variables \( S_e \) and confounding secondary variables \( S_g \). Knowledge is represented by the joint probability \( p(x,S_e,S_g) \).

Influences between variables are represented by conditioning, and a graphical model expresses the conditional independencies between variables. Two random variables may only become independent, however, once the value of some third variable is known. This is called conditional independence. Recall from above that two random variables are independent if and only if their joint probability is equal to the product of their individual probabilities. Thus, if \( p(A,B) = p(A)p(B) \), then \( A \) and \( B \) are independent. If \( p(A|B,C) = p(A|C)p(B|C) \), then \( A \) and \( B \) are conditionally independent. When corn prices drop in the summer, hay fever incidence goes up. However, if the joint on corn price and hay fever is conditioned on "ideal weather for corn and ragweed", the correlation between corn prices and hay fever drops. This is because Corn price and hay fever symptoms are conditionally independent.

There is a correlation between eating ice cream and drowning. Why? What event should you condition on to make the dependence go away?

What is noise? Primary and secondary variables in SDT and in pattern inference theory

Noise is whatever you don't care to estimate, but contributes to the data.
Inference: Fruit example (due to James Coughlan; see Yuille, Coughlan, Kersten & Schrater).

The graph specifies how to decompose the joint probability:

The prior model on hypotheses, \( F \) & \( C \)

More apples (\( F=1 \)) than tomatoes (\( F=2 \)), and:

\[
\begin{align*}
\text{pp}_D[F] &= 1, 9 / 16, 7 / 16; \\
\text{TableForm}[\text{Table}[\text{pp}_D[F], \{F, 1, 2\}], \text{TableHeads} \rightarrow \{"F=a", "F=t"\}] \\
\end{align*}
\]

The conditional probability \( cpCF[F] \):

\[
\begin{align*}
\text{cpCF}[F, C] &= \text{Which}[F = 1 \&\& C = 1, 5 / 9, \\
& F = 1 \&\& C = 2, 4 / 9, F = 2 \&\& C = 1, 6 / 7, F = 2 \&\& C = 2, 1 / 7]; \\
\text{TableForm}[\text{Table}[\text{cpCF}[F, C], \{F, 1, 2\}, \{C, 1, 2\}], \\
\text{TableHeads} \rightarrow \{"F=a", "F=t"\}, \{"C=r", "C=g"\})]
\end{align*}
\]

So the joint is:

\[
\begin{align*}
\text{jpFC}[F, C] &= \text{cpCF}[F, C] \text{pp}_D[F]; \\
\text{TableForm}[\text{Table}[\text{jpFC}[F, C], \{F, 1, 2\}, \{C, 1, 2\}], \\
\text{TableHeads} \rightarrow \{"F=a", "F=t"\}, \{"C=r", "C=g"\})]
\end{align*}
\]

We can marginalize to get the prior probability on color alone:

\[
\text{pp}_C[C] := \sum_{F} \text{jpFC}[F, C]
\]

Question: Is fruit identity independent of material color—i.e., is \( F \) independent of \( C \)?

■ Answer

No.
The total joint probability

We now have enough information to put probabilities on the 2x2x2 "universe" of possibilities, i.e. all possible combinations of fruit, color, and image measurements. Looking at the graphical model makes it easy to use the product rule to construct the total joint, which is:

\[
p(F, C, Is, Ic) = p(Ic | C) p(C | F) p(Is | F) p(F)
\]

The generative model: Imaging probabilities

Analogous to collecting histograms for the two switch positions in the SDT experiment, suppose that we have gathered some "image statistics" which provides us knowledge of how the image measurements for shape Is, and for color Ic depend on the type of fruit F, and material color, C. For simplicity, our measurements are discrete and binary (a more realistic case, they would have continuous values), say Is = {am, tm}, and Ic = {rm, gm}.

P(Is=am,tm | F=am) = \{11/16, 5/16\}
P(Is=am,tm | F=tm) = \{9/16, 7/16\}
P(Ic=rm,gm | C=rm) = \{1/2, 1/2\}
P(Ic=rm,gm | C=gm) = \{7/16, 9/16\}

We use the notation am, rm, gm because the measurements are already suggestive of the likely cause. So there is a correlation between apple and apple-like shapes, am; and between red material, and "red" measurements. In general, there may not be an obvious correlation like this.

We define a function for the probability of Ic given C, \(cp(Ic | C)\):

\[
\text{cp}(Ic | C) := \text{Which}[Ic = \{\text{am}, \text{gm}\}, F = \{am, \text{tm}\} | C = \{\text{rm}, \text{gm}\}]
\]

The probability of Is conditional on F is \(cp(Is | F)\):

\[
\text{cp}(Is | F) := \text{Which}[Is = \{\text{am}, \text{tm}\}, F = \{am, \text{tm}\}]
\]

Three MAP tasks

Suppose that we measure Is=am, and Ic = rm. The measurements suggest "red apple", but to find the most probable, we need to take into account the priors too.

- Define argmax[] function:

\[
\text{argmax}[\cdot] := \text{Position}[\cdot, \text{Max}[\cdot]]
\]

- Pick most probable fruit AND color--Answer "red tomato"

Using the total joint, \(p(F,C | Is, Ic) = \sum_{\text{Is}, \text{Ic}} p(F,C,Is, Ic)\)
"Red tomato" is the most probable once we take into account the difference in priors.

Calculating \( p(F, C \mid I_s, I_c) \). We didn't actually need \( p(F, C \mid I_s, I_c) \), but we can calculate it by conditioning the total joint on the probability of the measurements:

\[
p_{F \mid C, I_s, I_c} = \frac{p_{F, C, I_s, I_c}}{p_{C, I_s, I_c}}
\]

\( p_{F, C, I_s, I_c} \) is the most probable once we take into account the difference in priors.

In this case, we want maximize the posterior:

\[
p(C \mid I_s, I_c) = \sum_{F} p(F, C \mid I_s, I_c)
\]

The answer is "apple"
Moral of the story: Optimal inference depends on the precise definition of the task

### Inference Tasks

Three types of knowledge in a graphical model: known, unknown to be estimated, unknown to be marginalized

We have three basic states for nodes in a graphical model: known, unknown to be estimated, unknown to be integrated out (marginalization). Suppose we have causal factors $S$, intermediate variables $M$ (e.g., that tell us how the components of $S$ get mixed), and image data $I$. For concreteness, consider a simple linear generative model:

$I = M \cdot S$, where $I$ and $S$ are vectors, and $M$ is a matrix. $M$ isn’t necessarily square. Now think of $I, M, S$ as all being random variables that can take on values over some domain.

**Synthesis** (forward, generative model): We want to model $I$. $p(I|S)$ or $p(I|S,M)$.

**Inference**, we want to model samples for $S$: $p(S|I)$ or $p(S|I,M)$.

**Learning**, we want to model $M$: $p(M|S)$, or $p(M)$

More details on Inference: Three types

- Detection
Classification

MAP rule: $\arg\max_i p(S_i \mid x)$.

Estimation (continuous)

$\arg\max_S p(S \mid x)$
Learning

Regression: estimating parameters that provide a good fit to data. E.g. slope and intercept for a straight line through points \( \{x_i, y_i\} \).

Density estimation: Regression on a probability density, with the added condition that the area under the fitted curve must sum to one.

**MAP minimizes probability of error: Proof for detection**

Here is why MAP minimizes average error. Suppose that \( x \) is fixed at a value for which \( P(S =s_b | x) > P(S = s_d | x) \). This is exactly like the problem of guessing “heads” or “tails” for a biased coin, with a probability of heads \( P(S = s_b | x) \).

Imagine the light discrimination experiment repeated many times and you have to decide whether the switch was set to bright or not — but only on those trials for which you measured exactly \( x \). The optimal strategy is to always say “bright”.

\[ p(\text{error}(t)) = \text{error} \text{ as a function of } t, \text{ defines a straight line with slope } p(\text{actually dim } | x) - p(\text{actually bright } | x). \]

(Just take the partial derivative with respect to \( t \).) We've assumed \( P(S = s_b | x) > P(S = s_d | x) \), so \( p(\text{error}(t)) \) has a negative slope, with the smallest non-negative value of \( t \) being one. So, error is minimized when \( \text{implies } "\text{bright}" | x = 1 \). I.e. Always say “bright”.

Always saying "bright" results in a probability of error \( P(\text{error } | x) = P(S = s_d | x) \). That's the best that can be done on average. On the other hand, if the observation is in a region for which \( P(S = s_d | x) > P(S = s_b | x) \), the minimum error strategy is to always pick “dim” with a resulting \( P(\text{error } | x) = P(S = s_b | x) \). Of course, \( x \) isn't fixed from trial to trial, so we calculate the total probability of error which is determined by the specific values where signal states and decisions don't agree:

\[ p(\text{error}) = \sum_{i \neq j} p(\hat{s}_i, s_j) \]
\[ = \sum_{i \neq j} \int p(\hat{s}_i, s_j | x) p(x) \, dx = \sum_{i \neq j} \int p(\hat{s}_i | x) p(s_j | x) p(x) \, dx \]

Because the MAP rule ensures that \( p(\hat{s}_i, s_j | x) \) is the minimum for each \( x \), the integral over all \( x \) minimizes the total probability of error.
Next time

Ideal pattern detector for a signal which is exactly known

Psychophysical tasks & techniques

- The ROC
- The 2AFC (two-alternative forced-choice) method
- Rating scale
- Adaptive procedures for finding thresholds using 2AFC or yes/no.

Appendices

Marginalization and conditioning: A small dimensional example using list manipulation in Mathematica

- A discrete joint probability

All of our knowledge regarding the signal discrimination problem can be described in terms of the joint probability of the hypotheses, \( H \) and the possible data measurements, \( x \). The probability function assigns a number to all possible combinations:

\[
p[H, x]
\]

That is, we are assuming that both the hypotheses and the data are discrete random variables.

Let's assume that \( x \) can only take on one of three values, 1, 2, or 3. And suppose the joint probability is:

\[
p = \left\{ \begin{array}{ccc}
\frac{1}{12} & \frac{1}{12} & \frac{1}{6} \\
\frac{1}{3} & \frac{1}{6} & \frac{1}{6} \\
\frac{1}{12} & \frac{1}{12} & \frac{1}{6}
\end{array} \right\}
\]

The total probability should sum up to one. Let's test to make sure. We first turn the list of lists into a single list of scalars using Flatten[]. And then we can sum either with Apply[Plus, Flatten[p]].

\[
\text{Flatten}[p]
\]

We can pull out the first row of \( p \) like this:

\[
p[[1]]
\]

Is this the probability of \( x \)? No. For a start, the numbers don't sum to one. But we can get it through the two processes of marginalization and conditioning.

- Marginalizing

What are the probabilities of the data, \( p(x) \)? To find out, we use the sum rule to sum over the columns:
Apply Plus, p
\[ \begin{pmatrix} 5 & 1 & 1 \\ 12 & 4 & 3 \end{pmatrix} \]

"Summing over" is also called marginalization or "integrating out". Note that marginalization turns a probability function with higher degrees of freedom into one of lower degrees of freedom.

What are the prior probabilities? \( p(H) \)? To find out, we sum over the rows:

\[ p_H = \text{Apply[Plus, Transpose[p]]} \]
\[ \begin{pmatrix} 1 & 2 \\ 3 & 3 \end{pmatrix} \]

### Conditioning

Now that we have the marginals, we can get use the product rule to obtain the conditional probability through conditioning of the joint:

\[ p(x|H) = \frac{p(H,x)}{p(H)} \]

In the Exercises, you can see how to use Mathematica to do the division for conditioning. The syntax is simple:

\[ pxH = p / pH \]
\[ \begin{pmatrix} 1 & 1 & 1 \\ 4 & 4 & 4 \end{pmatrix} \]

Note that the probability of \( x \) conditional on \( H \) sums up to 1 over \( x \), i.e. each row adds up to 1. But, the columns do not.

\( p(x|H) \) is a probability function of \( x \), but a likelihood function of \( H \). The posterior probability is obtained by conditioning on \( x \):

\[ p(H|x) = \frac{p(H,x)}{p(x)} \]

Syntax here is a bit more complicated, because the number of columns of \( px \) don't match the number of rows of \( p \). We use Transpose[] to exchange the columns and rows of \( p \) before dividing, and then use Transpose again to get back the 2x3 form:

\[ p_Hx = \text{Transpose[Transpose[p] / px]} \]
\[ \begin{pmatrix} 5 & 1 & 1 \\ 12 & 4 & 3 \end{pmatrix} \]

**Plotting the joint**

The following BarChart[] graphics function requires in add-in package \("<< Graphics'Graphics"\), which is specified at the top of the notebook. You could also use ListDensityPlot[].

**Using Mathematica lists to manipulate discrete priors, likelihoods, and posteriors**

- **A note on list arithmetic**

  We haven't done standard matrix/vector operations above to do conditioning. We've take advantage of how Mathematica divides a 2x3 array by a 2-element vector:

\[ M = \text{Array[m,\{2,3\}]} \]
\[ X = \text{Array[x,\{2\}]} \]

\[ \begin{pmatrix} m(1, 1) & m(1, 2) & m(1, 3) \\ m(2, 1) & m(2, 2) & m(2, 3) \end{pmatrix} \]

\[ \text{Transpose[\{x(1), x(2)\}]} \]
Putting the probabilities back together again to get the joint

\[
\begin{bmatrix}
\frac{1}{12} & \frac{1}{12} & \frac{1}{6} \\
\frac{1}{3} & \frac{1}{6} & \frac{1}{6}
\end{bmatrix}
\]

Getting the posterior from the priors and likelihoods:

One reason Bayes' theorem is so useful is that it is often easier to formulate the likelihoods (e.g. from a causal or generative-model of how the data could have occurred), and the priors (often from heuristics, or in computational vision empirically testable models of the external visual world). So let's use Mathematica to derive \( p(H|x) \) from \( p(x|H) \) and \( p(H) \), (i.e. \( pHx \) from \( pxH \) and \( pH \)).

\[
\begin{align*}
px2 &= \text{Plus} @@ (pxH pH) \\
\begin{bmatrix}
\frac{5}{12} & \frac{1}{4} & \frac{1}{3}
\end{bmatrix}
\end{align*}
\]

\[
\begin{align*}
\text{Transpose}[@\text{Transpose}[(px2 pH)]] &= \text{Plus} @@ (px2 pH) \\
\begin{bmatrix}
\frac{1}{5} & \frac{1}{3} & \frac{1}{2} \\
\frac{2}{5} & \frac{1}{3} & \frac{1}{2}
\end{bmatrix}
\end{align*}
\]

Show that this joint probability has a uniform prior (i.e. both priors equal).

\[
p = \{\{1/8, 1/8, 1/4\}, \{1/4, 1/8, 1/8\}\}
\]

\[
\{\{\frac{1}{8}, \frac{1}{8}, \frac{1}{4}\}, \{\frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}\}
\]

Figure code

\[
x = 0.2; y = 0.8;
\text{Plot}[t x + (1-t) y, \{t, 0, 1\}, \text{AxesLabel} \rightarrow \{t, "p(error)"\}];
\]

Exercises

Exercise: Use density mapping theorem to make random number generator for density \( p(y) \)

Exercise: Show that MAP minimizes the probability of error for classification

\[
p(\text{error}) = \sum_{i \neq j} p(\hat{s}_i, s_j) = \sum_{i \neq j} \int p(\hat{s}_i | x) p(s_j | x) p(x) \, dx
\]

Let \( \hat{s} \) be the MAP choice, and the error using the MAP choice is,

\[
p(\text{error}^*) = \sum_{i \neq j} p(\hat{s}_i, s_j)
\]

\[
= \sum_{i \neq j} \int p(\hat{s}_i | x) p(s_j | x) p(x) \, dx
\]

We want to show that: \( p(\text{error}) - p(\text{error}^*) \geq 0 \)
Exercise: Calculate the information capacity of the eye

Consider an m x n pixel image patch. Is there a quantum limit to the number of light levels that can be represented in a resolution cell? (The size of a resolution cell is determined by the modulation transfer function of the optical device under consideration, which in this case would be the eye. We look later at how to estimate the spatial resolution of an imaging system).

\[ m \times n \]

Resolution cell capable of encoding \( L \) levels

Let \( SN \) be the maximum number of photons that land in a resolution cell. One can’t discriminate this level from any other with an infinitely small degree of precision. Requiring a sensitivity of \( d' \), determines the next dimmest light level:

\[ SN - 1 = SN - d' \]

This effectively quantizes the dynamic range of a resolution cell. Write a small iterative program to count the number of levels down to \( S1 = 0 \). Say the number of levels is \( L \), or \( \log_{10} L = t \) bits. Of course, one has to decide a priori what is a suitable discrimination level. But once done, the information capacity can be estimated by \( \text{ln} \) bits.

References


© 2000 Daniel Kersten, Computational Vision Lab, Department of Psychology, University of Minnesota.

(http://vision.psych.umn.edu/www/kersten-lab/kersten-lab.html)