Computational Vision
U. Minn. Psy 5036
Daniel Kersten
Lecture 5

Goals

Last time: ideal observer analysis

Overview

Ideal observer
- Model the data (image) generation process
- Define the inference task
- Determine optimal performance
- Derivation of optimal performance for discrimination given additive gaussian noise model

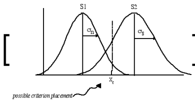
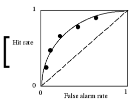
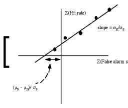
Compare human performance to the ideal
- Ideal normalizes for information available
- Statistical efficiency

Explain discrepancies in terms of:
- functional adaptation
- mechanism

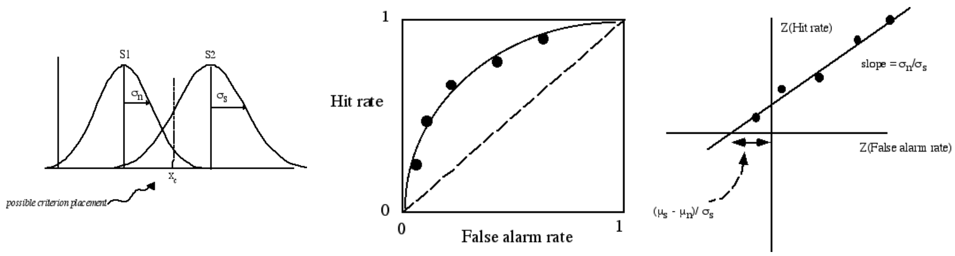
The Receiver Operating Characteristic (ROC)

Although we can't directly measure the internal distributions of a human observer's decision variable, we've seen that we can measure hit and false alarm rates, and thus d'.

By manipulating the criterion, we can generate a series of hit and false alarm rate pairs to plot an ROC curve. We can use this to see if an observer's decisions are consistent with the assumption of Gaussian distributions with equal variance. One can best test the gaussian equal variance assumption by re-plotting the ROC curve in terms of the z-scores of the hit and false alarm rates. A straight line of slope 1 is consistent with equal-variance Gaussian assumption.

In[98]:= GraphicsRow[{Image[], Image[], Image[]}]

Out[98]=



One can show that the area under the ROC curve is equal to the proportion correct in a two-alternative forced-choice experiment (Green and Swets).

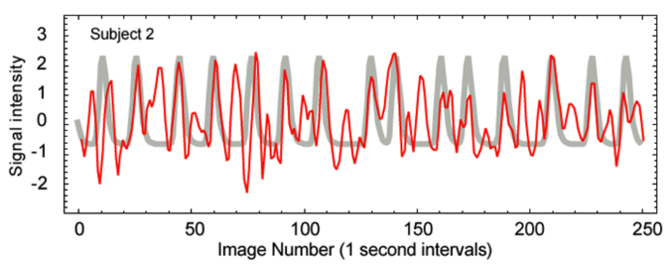
Sometimes, sensitivity is operationally defined as this area. This provides a single summary number, even if the standard definition of d' is inappropriate, for example because the variances are not equal.

Or we may not know or care whether the underlying distributions are gaussian. The area under the ROC curve provides a useful measure of sensitivity even if the additive gaussian model isn't known to be correct. It can also be thought of as a measure of how much information about signal vs. no signal can be extracted from the data.

Applications of ROC to neural measures

ROC curves can be used characterize the sensitivity of single neurons, as well as gross overall measures of activity such as comes from brain imaging data.

In the figure below, the gray lines represent a behavioral response by a human observer--i.e. when the signal is high, the observer is indicating subjective perceptual state that we treat as a "detection". The red lines represent a measured fMRI brain signal. How well does the brain signal predict what the observer is reporting?



The 2AFC (two-alternative forced-choice) method

Rather than manipulating the criterion, we can use the 2AFC method to minimize the effect of an unstable criterion.

$$d' = -\sqrt{2}\, z(\text{proportion correct})$$

Today: Probability Overview. SKE ideal observer

Review some probability and statistics
Pattern detection: The signal-known-exactly (SKE) ideal observer
Demo of 2AFC for pattern detection in noise

Motivation: What pattern does the eye see best?

Make the question precise by asking:

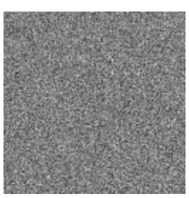
For what patterns does the human visual system have the highest detection efficiencies relative to an ideal observer?

Animals, or particularly dangerous ones?

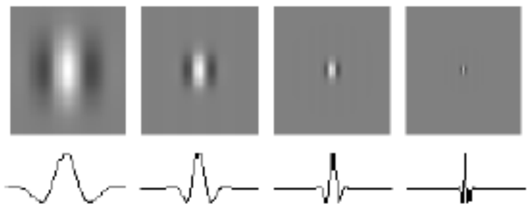
Faces, or a particular face?

Or something simple, like a spot, but perhaps of a particular size?

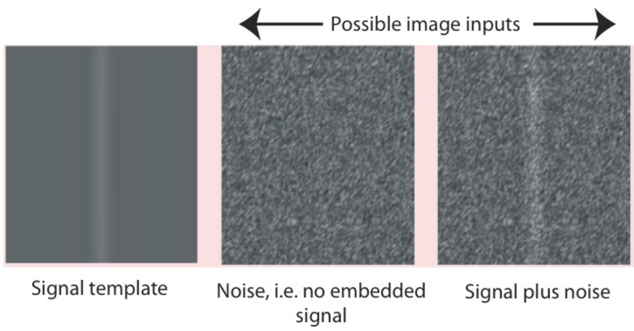
Or something complex, like a "frozen" noise image?



Or some pattern motivated by neurophysiology? E.g. the kinds of spatial patterns preferred by single neurons in the primary visual cortex ...



Answering this question requires specification of 1) a generative model that describes the variations in both the signal and the non-signal conditions, 2) the task; 3) an inference method. In general 1) and 2) hard to do, but we can do it for simple cases such as when the signal image is constant, and the data is either "white gaussian noise" or the signal added to white gaussian noise.



Some intuition: Measures of pattern similarity

The fundamental problem of pattern recognition is deciding whether an input pattern x matches a stored representation s. This decision requires some measure of comparison between the input and the stored "template" s. One might also want to know how close two input images are to each other. Given two patterns represented by vectors x and s, how can we measure how close or similar they are?

Some possibilities are: Abs[x-s], Cos[x,s], or Dot[x,s].

(See:
http://reference.wolfram.com/mathematica/guide/DistanceAndSimilarityMeasures.html)

A simple, useful measure often used in computer vision is normalized cross-correlation.

The subjective similarity between two images is a complex process, and there is no universal metric. There is substantial research on the topic, much of it addressing the need to quantify the difference between a decoded compressed image (e.g. for a lossy compression method like jpeg) and the original image.

In this lecture, we treat the very simple case of detection in additive white gaussian noise, and will see below that the ideal strategy is to compute the cross-correlation decision variable for each image (i.e. the dot product between each image data vector, say x, and an exact template of the signal, s, one is looking for), and pick the image which gives the larger cross - correlation.

Probability: overview of a few definitions

For terminology, a fairly comprehensive outline, and overview, see notebook: ProbabilityOverview.nb in the syllabus web page, and for a general introduction in the context of modeling in cognition and perception see: Griffiths and Yuille (2008).

For the section below, we'll use the properties of independence. Here is a quick overview of what we need today.

Expectation & variance

Definition of expectation or average or mean:

$$\text{Mean}[X] = \overline{X} = E[X] = \Sigma\, x[i]\; p[x[i]] \sim \sum_{i=1}^{N} x_i \,/\, N$$

Analogous to center of mass, where p(x) plays the role of mass density:

$$\mu = E[X] = \int x\, p(x)\, dx$$

The difference is that all probablistic “objects” must “weigh” 1:

$$\int p(x)\, dx = 1$$

Some rules for expections:

E[X+Y] = E[X] + E[Y]
E[aX] = aE[X]
E[X+a] = a+E[X]

Definition of variance:

$$\sigma^2 = \text{Var}[X] = E[[X\text{-}\mu]\text{^}2] = \sum_{j=1}^{N} ((p(x(j)))\,(x(j) - \mu))^2 = \sum_{j=1}^{N} (x_j - \mu)^2\, p_j$$

$$\text{Var}[X] = \int (x-\mu)^2\, p(x)\, dx \sim \sum_{i=1}^{N} (x_i - \mu)^2 \big/ N$$

Standard deviation:

$$\sigma = \sqrt{\text{Var}[X]}$$

Some rules:

$$\text{Var}[X] = E\left[X^2\right] - E[X]^2$$
$$\text{Var}[aX] = a^2\, \text{Var}[X]$$

Statistics for independent random variables

Independence means that knowledge of one event doesn't change the probability of another event.

p(X)=p(X|Y)

which is equivalent to:

p(X,Y)=p(X)p(Y) -- This is a key formula we will use below. It follows from p(X,Y) = p(X|Y)p(Y).

If p(X,Y)=p(X)p(Y), then

E[X Y] = E[X] E[Y] (i.e. X and Y are uncorrelated)

Var[X + Y] = Var[X] + Var[Y] (for uncorrelated random variables X and Y)

```
Var[c X] = c^2 Var[X], where c is a constant
```

Ideal pattern detector for a signal which is exactly known ("SKE" ideal)

The signal-known-exactly ideal (SKE) has a built-in, fixed template that precisely matches the signal that it is looking for. The signal is embedded in "white gaussian noise", or more precisely the signal is added to the noise. "white" means the pixels are not correlated with each other--intuitively this means that you can't reliably predict what one pixel's value is from any of the others. (In general tho', lack of correlation doesn't necessarily imply independence.)

Assignment 1 simulates the behavior of this ideal. In the absence of any internal noise, this ideal detector behaves as one would expect a linear neuron to behave when a target signal pattern exactly matches its synaptic weight pattern. There are some neurons in the primary cortex of the visual system called "simple cells". These cells can be modeled as ideal detectors for the patterns that match their receptive fields. In actual practice, neurons are not noise-free, and only approximately linear over a certain range. For example, simple cells show a rectifying property in which summed inputs below a threshold produce zero (not negative) response. More on this later.

Calculating the Pattern Ideal's d' based on signal-to-noise ratio

The signal + gaussian noise generative model

x = s + n, where s is a vector of image intensities, e.g. corresponding to a face, snake, spot, ...or a gabor pattern

x = n, where n is a vector representing a sample of white gaussian noise

Each element of n is assumed to have a mean of zero and standard deviation of σ. See the Exercise section below for Mathematica code of the generative process.

Overview

We are going to do two things:

1. Show that a simple decision variable for detecting a known fixed pattern in white gaussian noise is the dot product, or cross-correlation, of the observation image x with the known signal image s:

r = x • s , or alternatively written as

$r = \sum_{i=1}^{N} x(i)\, s(i)$

2. Show that d' is given by:

$$d' = \frac{\sqrt{s \bullet s}}{\sigma}$$

s and x are vectors, i.e. lists, of the image intensities, and σ is the standard deviation of the added gaussian noise. Knowing the d' for the ideal will enable us to calculate the absolute efficiency for human visual detection.

1. Cross correlation produces an ideal decision variable: Proof

What is the optimal decision variable? Starting from the maximum a posteriori rule, given equal priors, we noted that basing decisions on the likelihood ratio is ideal, in the sense of minimizing the probability of error. So the likelihood ratio is a decision variable. But it isn't the only one, because any monotonic function is still optimal. So our goal is to pick a decision variable which is simple, intuitive, and easy to compute. But first, we need an expression for the likelihood ratio:

$$\frac{p(x \mid \text{signal plus noise})}{p(x \mid \text{noise only})}$$

where x is the vector representing the image measurements actually observed

x = s + n, under signal plus gaussian noise condition
x = n, under gaussian noise only condition

First let's consider just one pixel of intensity x. Under the signal plus noise condition, the values of x fluctuate (from one trial to the next) about the average signal intensity s with a Gaussian distribution (gp[ ]) with mean s and standard deviation σ.
So under the signal plus noise condition, the likelihood p[x | s] is gp[x-s; σ]:

```
In[99]:= Clear[x,s,σ]
gp[x_,s_,σ_]:= (1/(σ*Sqrt[2 Pi])) Exp[-(x-s)^2/(2 σ^2)]
gp[x,s,σ]
```

Out[101]=
$$\frac{e^{-\frac{(-s+x)^2}{2\sigma^2}}}{\sqrt{2\pi}\ \sigma}$$

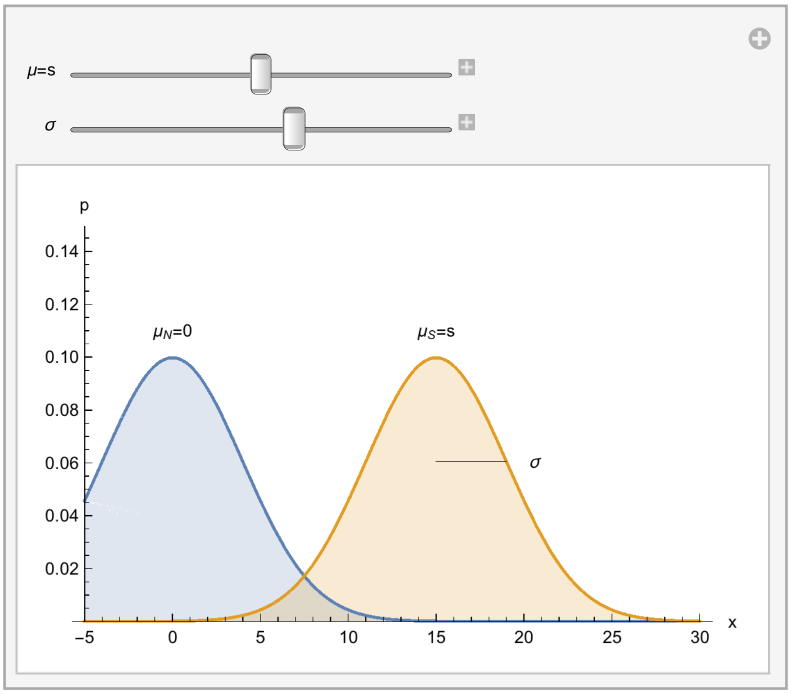
Now consider the noise only condition. Again, consider just one pixel of intensity x. Under the noise only condition, the values of x fluctuate about the average intensity corresponding to the mean of the noise, which we assume is zero.
So under the noise only condition, the likelihood p[x | n] is:

```
In[102]:= gp[x,0,σ]
```

Out[102]=
$$\frac{e^{-\frac{x^2}{2\sigma^2}}}{\sqrt{2\pi}\ \sigma}$$

In[103]:=
```
gauss[x_, mean_, std_] := E^(-(x-mean)^2/(2 std^2))/(std Sqrt[2 π]); (*Define our own gaussian distribution*)
b = 15; d = 0; sigma = 4; max = gauss[0, 0, sigma];
Manipulate[
 Plot[{gauss[x, d, sigma2], gauss[x, b1, sigma2]}, {x, -5, 30}, AxesLabel → {"x", "p"},
  Filling → Axis, PlotRange → {0, max + .05}, AxesOrigin → {-5, 0},
  Epilog → {Text["μS=s", {b1, 0.11}],
    Text["σ", {b1 + sigma2 * 1.4, (Exp[-.5] / (Sqrt[2.0 * Pi] * sigma2))}],
    Line[{{b1, (Exp[-.5] / (Sqrt[2.0 * Pi] * sigma2))}, {b1 + sigma2,
       (Exp[-.5] / (Sqrt[2.0 * Pi] * sigma2))}}], Text["μN=0", {d, 0.11}]}],
 {{b1, b, "μ=s"}, d, 30}, {{sigma2, sigma, "σ"}, 1, 6}]
```

Out[104]=



But we actually have a whole pattern of values of x, which make up an image vector x. So consider a pattern of image intensities represented by a vector x = {x[1],x[2],...x[N]} where x[i] is the measured intensity at pixel i. s[i] would be the measured intensity at pixel i if the signal was displayed with no noise. s[i] would also be the average value of that pixel with added noise.

Let the mean values of each pixel under the signal plus noise condition be given by vector s = {s[1],s[2],...,s[N]}. Under our generative model, the noise added to one pixel is independent of any other. Independence between pixels means we can multiply the individual probabilities to get the global joint image probability. (See above and ProbabilityOverview.nb)

The joint probability of an image observation x, under the signal hypothesis then is:

In[105]:= `Product[gp[x[i],s[i],σ],{i,1,N}]`

Out[105]= $$\prod_{i=1}^{N} \frac{e^{-\frac{(-s[i]+x[i])^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}$$

where i =1 to N pixels.

In general, whether we can assume independence depends on the problem. In our case, the samples are independent by definition--as "experimenters" we generate the noise as independent samples. We don't let other noise sample draws influence any other.
The joint probability of an image observation x, under the noise only hypothesis is:

In[106]:= `Product[gp[x[i],0,σ],{i,1,N}]`

Out[106]= $$\prod_{i=1}^{N} \frac{e^{-\frac{x[i]^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}$$

Now we have what we need for the likelihood ratio:

In[107]:= `Product[gp[x[i], s[i], σ], {i, 1, N}] / Product[gp[x[i], 0, σ], {i, 1, N}]`

Out[107]= $$\frac{\prod_{i=1}^{N} \frac{e^{-\frac{(-s[i]+x[i])^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}}{\prod_{i=1}^{N} \frac{e^{-\frac{x[i]^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}}$$

So at this point, we could just stop with the math and write a program to use this product to make ideal decisions. E.g. if the product is bigger than 1, choose the signal hypothesis, and if less than 1 choose the noise hypothesis. But this is inefficient, and could be problematic because of the limitations in numerical precision (What if the computer rounds off just one of the factors in the denominator to zero?).

With a little more work, we can get a much simpler rule, and one that provides insight into possible processes.

Recall that any monotonic function, f() of the likelihood ratio would give the same performance. A monotonic function simply means that whenever the likelihood ratio is bigger than 1, f(likelihood ratio) is bigger than f(1). So if we can find some monotonic function of the likelihood ratio that is simple, we'll have a simpler thing to calculate. The optimal decision rule in a yes/no experiment will be to choose "signal" if f(likelihood ratio)>f(1), and noise otherwise). Let's try one--the natural logarithm will turn the product into a sum:

In[108]:= $$\text{Log}\left[\frac{\prod_{i=1}^{N} gp[x[i], s[i], \sigma]}{\prod_{i=1}^{N} gp[x[i], 0, \sigma]}\right]$$

Out[108]= $$\text{Log}\left[\frac{\prod_{i=1}^{N} \frac{e^{-\frac{(-s[i]+x[i])^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}}{\prod_{i=1}^{N} \frac{e^{-\frac{x[i]^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}}\right]$$

which can be simplified to:

$$\text{Log}\left(\prod_{i=1}^{N} \frac{e^{-\frac{(x(i)-s(i))^2 - x(i)^2}{2\sigma^2}}}{\sqrt{2\pi}\ \sigma}\right)$$

which is monotonic with:

$$\text{Log}\left[\prod_{i=1}^{N} e^{\frac{2\,x(i)\,s(i)}{2\sigma^2}}\right]$$

which simplifies to

$$\left(1/\sigma^2\right)\sum_{i=1}^{N} x(i)\,s(i)$$

But this is monotonic with:

$$r = \sum_{i=1}^{N} x(i)\,s(i)$$

In case that went by too fast, we can use Mathematica's ability to handle symbolic expressions to see how to arrive at the same result. To be concrete, let N = 5.

In[109]:= $\text{Log}\left[\frac{\prod_{i=1}^{5} gp[x[i], s[i], \sigma]}{\prod_{i=1}^{5} gp[x[i], 0, \sigma]}\right]$

Out[109]= $\text{Log}\left[e^{\frac{x[1]^2}{2\sigma^2} - \frac{(-s[1]+x[1])^2}{2\sigma^2} + \frac{x[2]^2}{2\sigma^2} - \frac{(-s[2]+x[2])^2}{2\sigma^2} + \frac{x[3]^2}{2\sigma^2} - \frac{(-s[3]+x[3])^2}{2\sigma^2} + \frac{x[4]^2}{2\sigma^2} - \frac{(-s[4]+x[4])^2}{2\sigma^2} + \frac{x[5]^2}{2\sigma^2} - \frac{(-s[5]+x[5])^2}{2\sigma^2}}\right]$

Now use PowerExpand[] and Simplify[] to simplify the above expression:

In[110]:= **Simplify[PowerExpand[%]]**

Out[110]= $-\frac{1}{2\sigma^2}\left(s[1]^2 + s[2]^2 + s[3]^2 + s[4]^2 + s[5]^2 - 2\,s[1]\,x[1] - 2\,s[2]\,x[2] - 2\,s[3]\,x[3] - 2\,s[4]\,x[4] - 2\,s[5]\,x[5]\right)$

Notice that the terms s[i] are fixed by definition (the "signal is known exactly"), so we can lump them together as a constant c.

$$\left(1/\sigma^2\right)\sum_{i=1}^{N} x(i)\,s(i) + c$$

But because we only care that the final function is monotonic with the likelihood ratio, we can drop the $\left(1/\sigma^2\right)$ and c terms:

In[111]:= **Sum[x[i] s[i],{i,1,N}]**

Out[111]= $\sum_{i=1}^{N} s[i]\,x[i]$

$$r = \sum_{i=1}^{N} x(i)\,s(i)$$

In other words, we've proven that the dot product, r, (or cross-correlation or “matched filter”) provides a decision variable which is optimal--in the sense that if we use the rule, the probability of error will be least.

Now, let's calculate d'.

2. Derive formula for d'

By definition

In[112]:= `d' = (μ2 - μ1)/σ`

Out[112]= $\frac{-\mu 1+\mu 2}{\sigma}$

For our light discrimination example, $\mu 2$ = b, and $\mu 1$ =d, the mean photon counts under the high and low switch settings. What are $\mu 2$, and $\mu 1$ for the pattern detection case? Like the light or dot case, they are the mean values of the decision variable under the two hypotheses.

In our additive gaussian noise model with just one pixel, d'=(s - 0)/σ.

But in our pattern case, $\mu 2$ is the mean of the decision variable, r under the signal hypothesis (i.e. "switch set to send signal"), and $\mu 1$ is the mean under the noise-only hypothesis (i.e. switch set to not send signal).

To get d', we need formulas for the means and standard deviation for the decision variable, r under these two hypotheses.

First, suppose the switch is set for signal trials. What is the average and standard deviation of r? I.e. $\mu 2$ and σ?

$$\mu 2 = \text{Average}[r] =$$

$$\text{Average}\left[\sum_{i=1}^{N} x(i)\, s(i)\right] = \sum_{i=1}^{N} \text{Average}[x(i)\, s(i)] = \sum \text{Average}[x(i)]\, s(i) = \sum s(i)\, s(i) = \sum s(i)^2$$

$$\mu 2 = \sum_{i=1}^{N} s(i)^2$$

(We've used the above rules: E[X+Y] = E[X]+E[Y], E[aX]=aE[X]. And because x(i) = s(i) + n(i), Average[x(i)]=s(i), using E[X+a]=a+E[X].)

And the variance is:

$$\text{Var}\left(\sum_{i=1}^{N} x(i)\, s(i)\right) = \sum_{i=1}^{N} \text{Var}[x(i)]\, s(i)^2 = \sum_{i=1}^{N} \sigma^2 s(i)^2 = \sigma^2 \sum_{i=1}^{N} s(i)^2$$

We've used the rules: Var[Y + Z] = Var[Y] + Var[Z], and Var[x=s + n]=Var[constant + n]=Var[n]. The s(i)'s are constant.
And, recall that Var[c Y] = c^2 Var[Y]).

Second, suppose the switch is set for noise only trials. The average of the dot product is:

$$\mu 1 = \text{Average}[r] = \text{Average}\left[\sum_{i=1}^{N} x(i)\, s(i)\right] = \sum_{i=1}^{N} \text{Average}[x(i)]\, s(i) = \sum_{i=1}^{N} 0\, s(i) = 0$$

The variance is the same as for the signal case:

$$\text{Var}\left(\sum_{i=1}^{N} x(i)\, s(i)\right) = \sum_{i=1}^{N} \text{Var}[x(i)]\, s(i)^2 = \sigma^2 \sum_{i=1}^{N} s(i)^2$$

So d' is:

In[113]:=
```
Sum[s[i]^2, {i, 1, N}]/Sqrt[(σ^2 Sum[s[i]^2,{i,1,N}])]
```

Out[113]=
$$\frac{\sum_{i=1}^{N} s[i]^2}{\sqrt{\sigma^2 \sum_{i=1}^{N} s[i]^2}}$$

In[114]:=
```
FullSimplify[Sum[s[i]^2, {i, 1, N}]/Sqrt[(σ^2 Sum[s[i]^2,{i,1,N}])],σ>0]
```

Out[114]=
$$\frac{\sqrt{\sum_{i=1}^{N} s[i]^2}}{\sigma}$$

Or using dot product notation:

$$d' = \frac{\sqrt{\sum_{i=1}^{N} s(i)^2}}{\sigma} = \frac{\sqrt{s.s}}{\sigma}$$

In vision studies, s often represents contrast in luminance, which by definition has zero mean. And the dot product, s.s, is some time called contrast energy, with analogy to physical energy. It gets bigger with contrast, and because s has a dimension, varies with dimension or image size.

Calculating the Pattern Ideal's d' for a two-alternative forced-choice experiment from a z-score of the proportion correct.

Recall that for a 2AFC experiment, the observer gets two images to compare. One has the signal plus noise, and the other just noise. But the observer doesn't know which one is which. An ideal strategy is to compute the cross-correlation decision variable for each image, and pick the image which gives the larger cross-correlation. This strategy will result in a single performance number, the proportion correct, Pc. As shown earlier, d' for a 2AFC task can be calculated:

In[115]:=
```
z[p_] := Sqrt[2] InverseErf[1 - 2 p];
```

where z(*) is the z-score for Pc , the proportion correct. And

In[116]:=
```
dprime[x_] := N[-Sqrt[2] z[x]].
```

Syntax::sntxi:Incompleteexpression;moreinputis needed.

So we have what we need to compare the d’s of humans (through a measurement of Pc) and the ideal (through $\frac{\sqrt{s.s}}{\sigma}$). Now try the psychophysics demo for pattern detection in noise

DEMO: GaborSKEDetection.nb

Next time

Bayesian decision theory

Introduction to higher-level perceptual decisions as inference

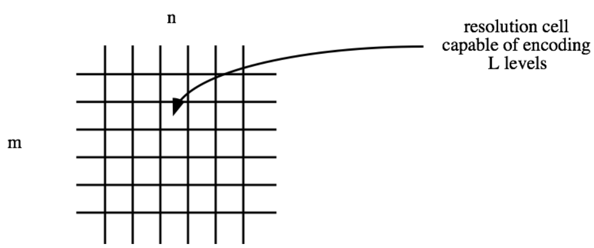
Bayesian decision theory

More types of inference tasks: synthesis, inference (detection, classification, estimation), learning

Exercises

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).



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:

$$S_{N-1} = S_N - d'\sqrt{S_N}$$

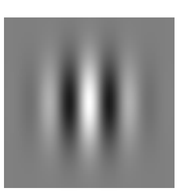
This effectively quantizes the dynamic range of a resolution cell. Write a small iterative program to count the number of levels down to S1 = zero. Say the number of levels is L, or $\log_s L = I$ bits. Of course, one has to decide a priori what is a suitable discrimination level. But once done, the information capacity can be estimated by Imn bits.

Generating gabor patch signals in additive noise

So what can you do with this particular ideal observer analysis? Take a look at:

Burgess, A. E., Wagner, R. F., Jennings, R. J., & Barlow, H. B. (1981). Efficiency of human visual signal discrimination. Science, 214, 93-94.

vs.



The signal + gaussian noise generative model

x = s + n, where s is a vector of image intensities corresponding to a gabor pattern
x = n, where n is white gaussian noise

Gabor patterns as signals

Basis set: Cartesian representation of Gabor functions:

In[116]:=
```
ndist=NormalDistribution[0,1];

cgabor[x_,y_, fx_, fy_,sx_,sy_] :=
Exp[-((x/sx)^2 +  (y/sy)^2)] Cos[2 Pi(fx x + fy y)];
```

Various frequencies , vertical orientations, and fixed width

In[119]:=
```
vtheta = Table[0, {i1,4}];
vf = {2,4};
hf = {0.0,0.0,0.0};
xwidth = {0.15,4};
ywidth = {4,4};
npoints = 128;
signalcontrast=0.15;
noisecontrast=0.2;
```

In[127]:=
```
lr = -1; ur = 1; step = (ur - lr) / (npoints - 1);
signal =
  Table[signalcontrast cgabor[y, x, vf[[1]], hf[[1]], xwidth[[1]], ywidth[[1]]],
   {x, lr, ur, step}, {y, lr, ur, step}];
noise = noisecontrast Table[Random[ndist], {npoints}, {npoints}];
```

Signal, noise, signal + noise

In[130]:=
```
sig = ArrayPlot[signal, Mesh → False,
    Frame → False, PlotRange → {-1, 1}, ColorFunction → "GrayTones"];
noi = ListDensityPlot[noise, Mesh → False, Frame → False,
    PlotRange → {-1, 1}, ColorFunction → "GrayTones"];
spn = ListDensityPlot[signal + noise, Mesh → False, Frame → False,
    PlotRange → {-1, 1}, ColorFunction → "GrayTones"];
```

In[133]:=
```
GraphicsRow[{sig, noi, spn}]
```

Out[133]=

References

Applebaum, D. (1996). Probability and Information . Cambridge, UK: Cambridge University Press.
Burgess, A. E., Wagner, R. F., Jennings, R. J., & Barlow, H. B. (1981). Efficiency of human visual signal discrimination. Science, 214, 93-94.
Cover, T. M., & Joy, A. T. (1991). Elements of Information Theory. New York: John Wiley & Sons, Inc.
Duda, R. O., & Hart, P. E. (1973). Pattern classification and scene analysis . New York.: John Wiley & Sons.
Green, D. M., & Swets, J. A. (1974). Signal Detection Theory and Psychophysics . Huntington, New York: Robert E. Krieger Publishing Company.
Kersten, D. (1984). Spatial summation in visual noise. Vision Research, 24,, 1977-1990.
Morgenstern, Y., & Elder, J. H. (2012). Local Visual Energy Mechanisms Revealed by Detection of Global Patterns. Journal of Neuroscience, 32(11), 3679–3696. doi:10.1523/JNEUROSCI.3881-11.2012
Ripley, B. D. (1996). Pattern Recognition and Neural Networks. Cambridge, UK: Cambridge University Press.
Schrater, P. R., Knill, D. C., & Simoncelli, E. P. (2000). Mechanisms of visual motion detection. Nat Neurosci, 3(1), 64-68.
Van Trees, H. L. (1968). Detection, Estimation and Modulation Theory . New York: John Wiley and Sons.
Watson, A. B., Barlow, H. B., & Robson, J. G. (1983). What does the eye see best? Nature, 31,, 419-422.