C106B Discussion 6: Probability and CV

1 Introduction

In anticipation for the upcoming unit on environment feedback, localization, and mapping, today we'll talk about:

1. Linearity of Expectation
2. Multivariate Random Variables
3. Hidden Markov Models
4. Low-Level Computer Vision Review

2 Linearity of Expectation

Random variables are commonly used to perform probabilistic calculations. They take on values corresponding to some distribution. The average value of some random variable X is known as the *expectation* of X.

Linearity of expectation allows us to compute the average value of a combination of multiple random variables. The theorem states:

$$\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$$
$$\mathbb{E}[cX] = c\mathbb{E}[X]$$

Problem 1: Suppose I toss all 10 (unique) pairs of my socks in the washing machine, but when I collect them from my dryer, I only have 16 socks remaining. In expectation, how many pairs can I expect to see?

We can solve this question using an indicator random variable and linearity of expectation.

Let us say we have some indicator random variable X_i for each pair of socks, such that

$$X_i = \begin{cases} 0, & \text{if the dryer eats at least 1 sock in pair i} \\ 1, & \text{if both socks survive} \end{cases}$$

We can solve for the probability that both socks in a single pair i survive with

$$P[X_i = 1] = \frac{16}{20} \times \frac{15}{19} = \frac{12}{19}$$

Because we've set the indicator to equal 1 if both socks survive, we can get the expected value of the indicator directly from the probability:

$$\mathbb{E}[X_i] = 1 \times P[X_i = 1] + 0 \times P[X_i = 1] = P[X_i = 1] = \frac{12}{19}$$

We can then use linearity of expectation to calculate the total expected number of surviving socks:

$$\mathbb{E}[\text{total surviving socks}] = \#\text{ of pairs} \times \mathbb{E}[\text{any given pair surviving}] = 10 \times \frac{12}{19} \approx 6.32 \text{ pairs}$$

3 Multivariate Random Variables

A *multivariate random variable*, also known as a *random vector*, can be thought of as a group of random variables that are associated with one another in a single mathematical system. For example, a robot's predicted (x, y, z) location might be the output of some probabilistic function of observed environment variables, and the coordinates are grouped together to represent the state.

3.1 Mean, Covariance, and Cross-Covariance

The CDF takes a vector as input with the number of entries corresponding to the number of random variables:

$$F_X(x) = P(X_1 \leq x_1, ..., X_n < x_n)$$

As a result, the mean, or expected value, of a multivariate random variable is a vector as well:

$$\mathbb{E}[X] = (\mathbb{E}[X_1], ..., \mathbb{E}[X_n])^T$$

The *variance* of a single random variable equals the average distance from the mean of the values that it can take. It can be computed in two different ways:

$$Var(X) = \sigma^2(X) = \frac{\Sigma(x_i - \bar{x})^2}{N} = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

The *covariance* of multiple random variables represents the extent to which they correspond in values. If they both increase and decrease together, this value will be positive; whereas if they move in opposite directions, the value will be negative. Independent random variables will have 0 covariance (although the converse does not hold true).

$$cov(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$$

A single multivariate random variable will have an associated covariance matrix to represent pairwise covariance. Covariance matrices are symmetric positive semidefinite. A cross-covariance matrix can be calculated to represent the covariance between two different multivariate random variables. The element in the i, j position represents the covariance between the i-th value in the first vector and the j-th value in the second.

$$cov(\mathbf{X}, \mathbf{Y}) = \mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{Y} - \mathbb{E}[\mathbf{Y}])^T] = \mathbb{E}[\mathbf{X}\mathbf{Y}^T] - \mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{Y}]^T$$

Problem 2: Interpret the following covariance matrix. Which of the values is/are invalid?

$$\begin{bmatrix} 3 & 5 & 0 \\ 5 & 12 & -7 \\ 6 & -7 & -4 \end{bmatrix}$$

1. The value of -4 on the diagonal is invalid. The diagonal of a covariance matrix is the variance of the corresponding random variable in the vector. As a result, it should be the *squared* distance to the mean, a positive number.
2. The off-diagonal terms of 0 and 6 are invalid. Covariances may be negative, but they must be commutative (i.e. $Cov(X_i, X_j) = Cov(X_j, X_i)$). As a result, the overall matrix must be symmetric.

3.2 Multivariate Gaussians

A multivariate Gaussian random variable, or normal distribution, follows the following distribution:

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{(\frac{n}{2})}|\Sigma|^{(\frac{n}{2})}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$$

The sum of Gaussians is a Gaussian:

$$X \sim N(\mu_X, \sigma_X^2)$$
$$Y \sim N(\mu_Y, \sigma_Y^2)$$
$$Z = X + Y$$
$$Z \sim N(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$$

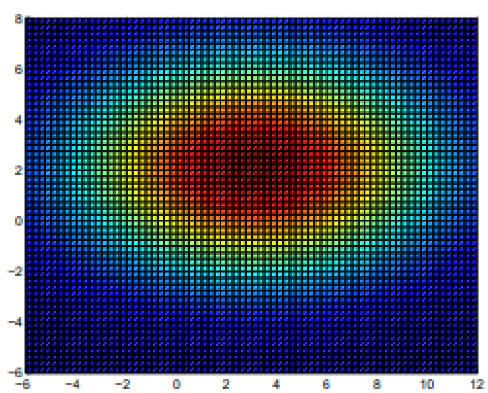
The PDF of a multivariate Gaussian with a diagonal covariance matrix will be the same as that of n independent Gaussians (uncorrelated implies independence).

Problem 3: The isocontours of a function $f : \mathbf{R}^2 \longrightarrow \mathbf{R}$ are of the form $x \in \mathbf{R}^2 : f(x) = c$. Find the isocontours of a multivariate Gaussian, both with and without a diagonal covariance matrix. What kind of intuition does this give you about Gaussians?

Setting the PDF of the Gaussian equal to some constant for a 2D Gaussian will end up with an equation that looks like

$$1 = (\frac{x_1 - \mu_1}{r_1})^2 + (\frac{x_2 - \mu_2}{r_2})^2$$

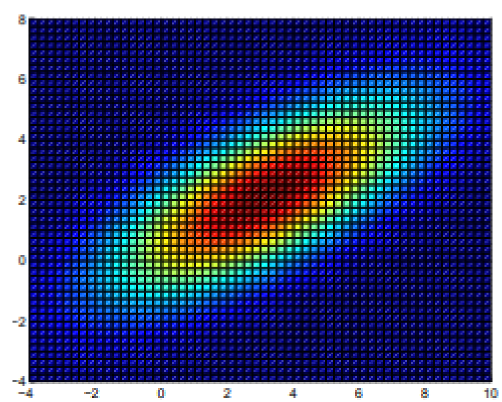
which is the equation for an ellipse! This tells us that even a multivariate Gaussian will have a single peak at the intersection of the means with probability decreasing radially around it. (A full derivation for this can be found here, also the source for the below images.)



(a)

$$\mu = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 25 & 0 \\ 0 & 9 \end{bmatrix}$$



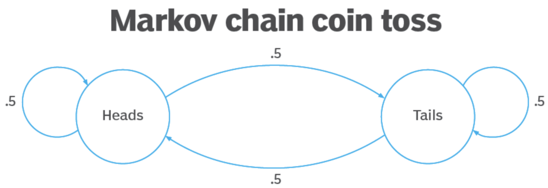
(b)

$$\mu = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 10 & 5 \\ 5 & 5 \end{bmatrix}$$

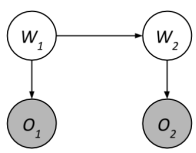
4 Hidden Markov Models

A *Markov Chain* possesses the Markov property - the next state depends only on the current state and is entirely independent of the past. For example, a coin toss can be expressed as a Markov Model:



A *Hidden Markov Model* is often used in systems where data is continuously fed in over time. HMMs assume that states themselves have the Markov property. However, they are unknown - *observations* are used to form a probabilistic distribution of your current state. The observations could be sensor readings, for example, and the state might be the (unknown) location of your robot in the space. This is quite useful for localization!

Problem 4: Suppose we observe $O_1 = a$ and $O_2 = b$. Compute the probability distribution $P(W_2|O_1 = a, O_2 = b)$. [Source: CS 188 Fa22 Discussion 9]



W_1	$P(W_1)$
0	0.3
1	0.7

W_t	W_{t+1}	$P(W_{t+1}|W_t)$
0	0	0.4
0	1	0.6
1	0	0.8
1	1	0.2

W_t	O_t	$P(O_t|W_t)$
0	a	0.9
0	b	0.1
1	a	0.5
1	b	0.5

As we make observations, the probability distribution of our current state will become more and more accurate.

1. **Observation & State Probability:** We know that we've observed a in our first state. Let's compute the joint probability distribution for our state, $P(W_1, O_1 = a)$.

   $P(W_1, O_1 = a) = P(W_1)P(O_1 = a|W_1) \rightarrow$ chain rule

   $P(W_1 = 0, O_1 = a) = (0.3)(0.9) = 0.27$
   $P(W_1 = 1, O_1 = a) = (0.7)(0.5) = 0.35$

2. **Propagation:** Using the previous calculation, we are going to propagate the current observation. That is, based on what what we know about the probability distribution for W_1 and the observation we got at time 1 ($O_1 = a$), we will predict what W_2 will be, $P(W_2, O_1 = a)$.

   $P(W_2, O_1 = a) = \Sigma_{w_1} P(w_1, O_1 = a)P(W_2|w_1) \rightarrow$ Law of Total Probability

   $P(W_2 = 0, O_1 = a) = (0.27)(0.4) + (0.35)(0.8) = 0.388$
   $P(W_2 = 1, O_1 = a) = (0.27)(0.6) + (0.35)(0.2) = 0.232$

3. **Update:** We will now update our prediction of what the next state is with the observation we make at the next step, $P(W_2, O_1 = a, O_2 = b)$

   $P(W_2, O_1 = a, O_2 = b) = P(W_2, O_1 = a)P(O_2 = b|W_2)$

   $P(W_2 = 0, O_1 = a, O_2 = b) = (0.388)(0.1) = 0.0388$
   $P(W_2 = 1, O_1 = a, O_2 = b) = (0.232)(0.5) = 0.116$

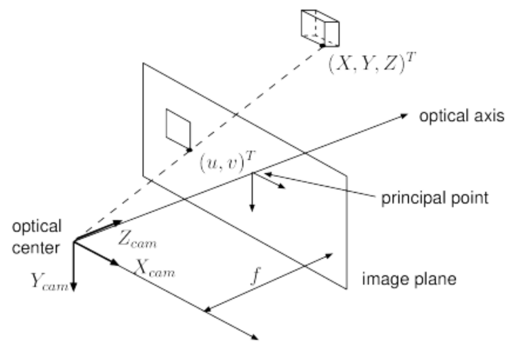
4. **Normalize:** We will normalize our probability distribution for our prediction of the next state given our observations, ensuring it sums up to 1, $P(W_2|O_1 = a, O_2 = b)$.

$$P(W_2 = 0|O_1 = a, O_2 = b) = \frac{0.0388}{0.0388 + 0.116} \approx 0.25$$

$$P(W_2 = 1|O_1 = a, O_2 = b) = \frac{0.0.116}{0.0388 + 0.116} \approx 0.75$$

5 Computer Vision

5.1 The Pinhole Camera Model



Problem 5: Using the image above, find the relationship between the 3D point $(X, Y, Z)^T$ to its corresponding 2D projection (u, v) onto the imaging plane (assume the focal length is 1).

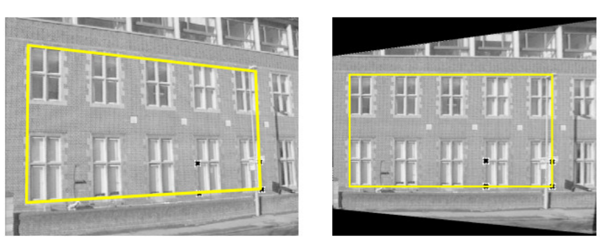
We can apply a similar triangles relationship to the pinhole camera model. The z-distances from the optical center to the image plane and the actual object must be proportional. Additionally, the x- and y-distances in the image and in the real world must also be proportional. As a result, we have the following relationship:

$$\frac{u}{f} = \frac{Y}{Z}$$

$$u = f\frac{Y}{Z}$$

5.2 Homography

The pinhole camera model has a particular center of projection, the point from which we view the world. The image plane is offset a certain focal length in some direction from that point. If we rotate the camera without moving it around, we maintain the same center of projection - we're just looking a different way! This is known as a *homography transformation* and can be thought of as a series of unprojection, rotation, and then reprojection. It's quite a useful function to have, especially when your robot is moving around. The homography transformation can be used to straighten images - if we've taken a picture with the camera pointed sideways, we can rotate it so that it looks as though the camera is pointed straight!



from Hartley & Zisserman

Problem 6: Let p correspond to a point on one image and let p' correspond to the same point in the scene, but projected onto another image. Write a general equation for how a homography matrix H maps points from one image to another. How would H be restricted if it must describe an affine transformation?

A homography is simply a matrix multiplication:

$$p' = Hp$$

Problem 7: How can you compute a homography matrix with real-world points?

This is an important part of SLAM, where we try to figure out the relationship between 2 images as we move through a space! We want to compare how we expected our robot to move with how it actually moved based on the photographs we have taken.

The actual process will be covered in a lot more detail in Discussion 7. It's considered the "front-end" of SLAM. To summarize the ideas, we first perform a feature estimation in each image - finding points of interest through an algorithm like corner detection. Then, we identify the matching points in each picture (which corners correspond to one another). Finally, we perform least squares estimation to discover the actual transformation matrix.