Homework 0

Due: February 2, 2018 at 11:59PM

Written Questions

(30 points total + 3 extra credit)

Problem 1: Bayes' Rule

Bayes' Rule, or Bayes' Theorem is an oft-used identity coming from probability theory. If we have two events of interest, A and B, we might want to ask what the probability of B is, given that we know A happened.

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Note that this is the same as

$$P(B|A) = \frac{P(A \cap B)}{P(A)}.$$

Later in this course, the parts of this formula may be relabeled:

$$Posterior = \frac{Likelihood * Prior}{Evidence}$$

This rule will be explicitly used in Bayesian algorithms, but it is also a principle that will *implicitly* underlie almost all of our machine learning algorithms. This problem consists of four parts, each worth 3 points (1 point if the answer is correct and an additional 2 points for showing correct work). As a hint, none of the four parts have the same answer. For the purposes of this question, assume that whales have equal probability of calving a male or female, and uniform probability of being born on any day of the week. Fun whale fact: No whale has ever been observed giving birth to twins.

Solution Bonus: There is an excellent guide to Bayes' Theorem available at brilliant.org. It includes a form of each of the parts below, as well as many other problems.

Part 1

(3 points)
Suppose a whale has two offspring. What is the probability that both offspring are male?

Solution:

$$\begin{aligned} P(both\ \ male) &= P(1st\ \ is\ \ male \cap 2nd\ \ is\ \ male) && (1)\\ &= P(1st\ \ is\ \ male) \times P(2nd\ \ is\ \ male)\ \ (since\ these\ events\ are\ independent) && (2)\\ &= \frac{1}{2} \times \frac{1}{2} && (3)\\ &= \frac{1}{4} && (4) \end{aligned}$$

Part 2

(3 points)
Suppose a whale has two offspring and the eldest is male. What is the probability that both offspring are male?

Solution: Although this seem intuitive, application of Bayes' Theorem is sure to avoid mistakes.

$$P(both\ \ male|1st\ \ is\ \ male) = \frac{P(1st\ \ is\ \ male|both\ \ male)P(both\ \ male)}{P(1st\ \ is\ \ male)} \tag{5}$$

$$= \frac{1 \times \frac{1}{4}}{\frac{1}{2}} \tag{6}$$

$$= \frac{1}{2} \tag{7}$$

Part 3

(3 points)
Suppose a whale has two offspring and at least one is male. What is the probability that both offspring are male?

Solution: Begin by noting that the combinations of (M,M), (M,F), (F,M), and (F,F) are equally likely, and $\frac{3}{4}$ of these have at least one male. That is, $P(at\ \ least\ \ 1\ \ male) = \frac{3}{4}$

$$P(both\ \ male|at\ \ least\ \ 1\ \ male) = \frac{P(at\ \ least\ \ 1\ \ male|both\ \ male)P(both\ \ male)}{P(at\ \ least\ \ 1\ \ male)} \tag{8}$$

$$= \frac{1 \times \frac{1}{4}}{\frac{3}{4}} \tag{9}$$

$$= \frac{1}{3} \tag{10}$$

Part 4

(Extra Credit: 3 points)
Suppose a whale has two offspring and at least one is a male whale born on a Wednesday. What is the probability that both offspring are male?

Solution: At first glance, this problem closely resembles the one above it. However, the answer (spoiler) is $\frac{13}{27}$, which is much closer to $\frac{1}{2}$ than $\frac{1}{3}$. This surprising result may become more intuitive if we ask a related question, such as "Suppose a whale has two offspring and at least one is a male whale who won the lottery. What is the probability that both offspring are male?" Obviously, having at least one male whale win the lottery is almost twice as likely if you have two male whales. (It actually *is* twice as likely, minus the chance they both win the lottery). This nearly balances the fact that having one male and one female is twice as likely as having two males. But how do we compute this result? A careful application of Bayes' Theorem will tell us which terms we need to compute. Here we will use the set-intersection form for simplicity. Let M_W denote the event that a male is born on Wednesday

$$P(both\ \ male|at\ \ least\ \ 1\ \ M_W) = \frac{P(at\ \ least\ \ 1\ \ M_W \cap both\ \ male)}{P(at\ \ least\ \ 1\ \ M_W)} \tag{11}$$

To satisfy the condition that one male whale was born on a Wednesday, we have the following possibilities:

Two males, first male was born on Wednesday, second male was not (6 possible combinations)

Two males, first male was not born on Wednesday, second male was (6 possible combinations)

Two males, both born on Wednesday (1 combination)

One male/one female, first offspring was a male born on Wednesday, female could be born on any day (7 combinations)

One female/one male, first offspring was a female born on any day, second offspring was a male born on Wednesday (7 combinations)

There are 27 combinations that satisfy the initial condition of at least one male being born on Wednesday, but only 13 of those combinations result in both offspring being male. Therefore, the probability that both offspring are male is $\frac{13}{27}$.

Problem 2: Linear Algebra - Singular Value Decomposition

One goal of a singular value decomposition is to represent a large matrix as a product of smaller ones. See Figure 1.

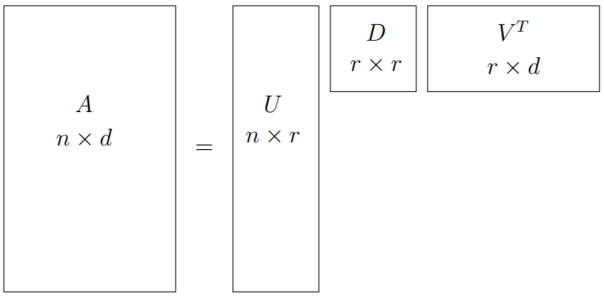

Figure 1: Singular Value Decomposition (SVD) represents a large matrix as a product of smaller ones. Note that the columns of U and V are orthonormal, and D is a real-valued diagonal matrix.

If we have an $n \times d$ matrix of n datapoints each in d dimensions, SVD finds a $d \times r$ matrix V that allows us to project the data to a smaller number of dimensions. This is the basis (pun intended) of Principal Component Analysis, as well as several other algorithms we will cover this semester. Note that V and U are orthonormal. An orthonormal matrix has the property that its columns are mutually orthogonal (the dot product of any pair of distinct columns is 0) and normalized (the dot product of any column with itself is

1). The combination of these properties means that the product of an orthonormal matrix's transpose with that matrix is an identity matrix.

$$V^T \times V = I$$
$$U^T \times U = I$$

This problem has 3 parts, each worth 3 points. Points will be awarded for clean derivations/proofs and for the compactness of the result (that is, the result should be expressed in simplest terms). Assume that expressions may freely include the transpose of any matrix, the identity matrix, and the zero matrix, in addition to any given terms.

Solution Bonus: There is an excellent guide to SVD's applications to Machine Learning available at this webpage.

Part 1

(3 points)
Let $\begin{bmatrix} A \\ A \end{bmatrix}$ be a $2n \times d$ matrix made by extending A with itself. Given a SVD of matrix A into $U \times D \times V^T$, where D is a diagonal matrix with dimension r, express SVD($\begin{bmatrix} A \\ A \end{bmatrix}$) in terms of U, D, and V. Provide an explanation for your answer.

Solution: $\begin{bmatrix} A \\ A \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}}U \\ \frac{1}{\sqrt{2}}U \end{bmatrix} \times \sqrt{2}D \times V^T$

This is easy to verify because $\begin{bmatrix} A \\ A \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}}U \\ \frac{1}{\sqrt{2}}U \end{bmatrix} \times \sqrt{2}D \times V^T = \begin{bmatrix} U \times D \times V^T \\ U \times D \times V^T \end{bmatrix} = \begin{bmatrix} A \\ A \end{bmatrix}$. The factors of $\frac{1}{\sqrt{2}}$ maintain the normalization of the new "U" matrix, and it is also orthogonal. Observe that $\left[\frac{1}{\sqrt{2}}U^T \frac{1}{\sqrt{2}}U^T\right] \times \begin{bmatrix} \frac{1}{\sqrt{2}}U \\ \frac{1}{\sqrt{2}}U \end{bmatrix} = I$. (If the notation is tripping you up, work through a few small examples on paper.) Many properties relating to duplication of datapoints or redundant columns can be proved in a similar fashion.

Part 2

(3 points)
Given $A = U_A \times D_A \times V_A^T$, use V_A to reduce the dimensionality of A. That is, let $B = A \times V_A$. Express the SVD(B) in terms of U_A, D_A, and V_A. Provide a justification for your answer.

Solution:

$$\begin{aligned} B = A \times V_A &= U_A \times D_A \times V_A^T \times V_A && (12) \\ &= U_A \times D_A \times (V_A^T \times V_A) && (13) \\ &= U_A \times D_A \times I \ (\textit{because } V_A \textit{ is orthonormal}) && (14) \\ &= U_A \times D_A && (15) \end{aligned}$$

So:

$$U_B = U_A \tag{16}$$
$$D_B = D_A \tag{17}$$
$$V_B = I \tag{18}$$

The critical step in this simplification is going from equation 13 to equation 14. In general the product of a matrix with its transpose does not necessarily equal the identity. However, V has the special property of being orthonormal; its columns are both orthogonal (the dot product of any pair of distinct columns is 0) and normal (the dot product of a column with itself is 1). It is easy to show from this that the product of an orthonormal matrix with itself is the identity matrix. Note that the identity matrix is itself orthonormal.

Part 3

(3 points)
Suppose we have $A = U_A \times D_A \times V_A^T$, and we let $B = A \times V_A$. In the same way, let $C = B \times V_B$. Write a SVD of C in terms of U_A, D_A, and V_A. Is there a maximum to the number of times the dimensionality of a dataset can be reduced using SVD?

Solution: This part should be straightforward if the minimal expression for the previous part was found.

$$\begin{aligned} B = A \times V_A &= U_A \times D_A \times V_A^T \times V_A \\ &= U_A \times D_A \ \ (see\ \ above) \end{aligned} \tag{19}$$
$$C = B \times V_B = B \times I = B = U_A \times D_A \tag{20}$$

By induction, we can show that there is no value to using SVD more than once on the same data.

Problem 3: Runtime Complexity

(12 points)
The Fibonacci sequence is defined as $F(n) = F(n-1) + F(n-2)$ for $n \geq 2$ where $F(0) = 0$ and $F(1) = 1$. We can make use of the fact (likely first noted by Edsger Dijkstra) that

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n = \begin{pmatrix} F(n+1) & F(n) \\ F(n) & F(n-1) \end{pmatrix}$$

for any positive integer n. So to compute $F(n)$ we can compute $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{n-1}$ and return the upper left number. Describe an algorithm to compute $F(n)$ for arbitrary n that uses *fewer than* $O(n)$ additions and multiplications (don't worry about the size of the integers), and prove that it satisfies this complexity bound. That is, prove a sublinear complexity bound in terms of n on the number of additions and multiplications this algorithm performs. Feel free to search for such an algorithm, but please generate the proof of the complexity bound yourself.

Solution Bonus: You can find a guide to Fibonacci-computing algorithms at: this webpage.

Solution: To compute $F(n)$ we can compute $\begin{pmatrix}1 & 1\\ 1 & 0\end{pmatrix}^{n-1}$ and return the upper left number. To compute $\begin{pmatrix}1 & 1\\ 1 & 0\end{pmatrix}^{n}$ for even n, we just need to compute $\begin{pmatrix}1 & 1\\ 1 & 0\end{pmatrix}^{\frac{n}{2}}$ and multiply it by itself - in other words, we can reduce the exponent by half with a constant amount of work. This suggests an algorithm with complexity $O(\log n)$.

Pseudo-code:

Function fib(n):
    if $n = 0$:
        return 0
    if $n = 1$:
        return 1
    mat $\leftarrow$ power$(n-1)$
    return mat[0][0]

Function power(n):
    if $n = 1$:
        return [[1, 1],[1, 0]]
    if n is even:
        mat $\leftarrow$ power$(\frac{n}{2})$
        return mat * mat
    else:
        mat $\leftarrow$ power$(n-1)$
        return mat * [[1, 1],[1, 0]]

Proof of time complexity:
A call to fib makes at most one call to power. Each call to power results in at most one recursive call. The total number of recursive calls is $O(\log n)$ because we divide n by 2 at least every other call (in the worst case, we get an odd number every time we divide by 2, but then we immediately make a recursive call with an even number and divide by 2). We do at most a constant number of arithmetic operations in each recursive call of power, and at most constant operations in fib, so the algorithm has $O(\log n)$ arithmetic operations.