CS2810 Mathematics of Data Models	Gold
Spring 2019	April 11, 2019

Homework 10

Due: April 17, 9PM
Point total: 60

Instructions:

- Submit your PDF and/or .py file to Blackboard by the due date and time. Please do not zip your files together, as this interferes with Blackboard's preview functionality. Always show all your work, and for full credit, you must use the method that the problem instructs you to use (unless none is mentioned). Handwritten or typeset solutions are both acceptable, but unreadable submissions will be penalized. You may discuss problems with other students, but you may not write up solutions together, copy solutions from a common whiteboard, or otherwise share your written work or code. Do not use code or language that is copied from the Internet or other students; attribute the ideas *and* rephrase in your own words.

Problem 1 (15 pts [6,4,3,2])

Suppose the following table is the result of a survey that we've sent out to our users. H is average hours online per day, D is dollars spent on software in the past month, M is household income in tens of thousands of dollars. (The number of data points is small so that you can do this problem by hand.)

H	D	M
4	50	7
16	0	2
1	400	12
2	40	9
5	10	5
3	30	8

i. Compute the 3x3 covariance matrix, using either the definition $E[(X - E[X])(Y - E[Y])]$ or the equivalent $E[XY] - E[X]E[Y]$ to compute each covariance.

Solution: We'll use the $Cov(X, Y) = E[XY] - E[X]E[Y]$ definition here. For the diagonal, that's $E[X^2] - E[X]^2$. $E[H] = (4 + 16 + 1 + 2 + 5 + 3)/6 = 31/6 = 5.17$
$E[D] = (50 + 400 + 40 + 10 + 30)/6 = 530/6 = 88.33$
$E[M] = (7 + 2 + 12 + 9 + 5 + 8)/6 = 43/6 = 7.17$
$E[HD] = (200 + 0 + 400 + 80 + 50 + 90)/6 = 136.67$
$E[DM] = (350 + 0 + 4800 + 360 + 50 + 240)/6 = 966.67$
$E[HM] = (28 + 32 + 12 + 18 + 25 + 24)/6 = 23.17$
$E[H^2] = (16 + 256 + 1 + 4 + 25 + 9)/6 = 51.83$
$E[D^2] = (2500 + 0 + 160000 + 1600 + 100 + 900)/6 = 27516.67$
$E[M^2] = (49 + 4 + 144 + 81 + 25 + 64)/6 = 61.17$

Then our covariance matrix is as follows (order is H, D, M as in the table):

$$\begin{bmatrix} 51.83-(5.17)^2 & 136.67-(5.17)(88.33) & 23.17-(5.17)(7.17) \\ 136.67-(5.17)(88.33) & 27516.67-(88.33)^2 & 966.67-(88.33)(7.17) \\ 23.17-(5.17)(7.17) & 966.67-(88.33)(7.17) & 61.17-(7.17)^2 \end{bmatrix} = \begin{bmatrix} 25.10 & -320.00 & -13.90 \\ -320.00 & 19714.48 & 333.34 \\ -13.90 & 333.34 & 9.76 \end{bmatrix}$$

ii. Use the covariance matrix to calculate the correlation coefficients between each pair of variables. (You can compute the standard deviations directly from the variances on the diagonal, without correcting for bias.)

Solution: $corr(H,D) = \frac{cov(H,D)}{\sigma_H \sigma_D} = \frac{-320.00}{\sqrt{25.10}\sqrt{19714.48}} = -0.45$
$corr(H,M) = \frac{cov(H,M)}{\sigma_H \sigma_M} = \frac{-13.9}{\sqrt{25.10}\sqrt{9.76}} = -0.89$
$corr(D,M) = \frac{cov(M,D)}{\sigma_D \sigma_M} = \frac{333.4}{\sqrt{19714.48}\sqrt{9.76}} = 0.76$

iii. Consider the largest positive or negative correlation and the old saw "correlation does not imply causation." Come up with: an explanation for how the first variable could causally affect the second; an explanation for how the second variable could causally affect the first; *and* an explanation of how a third variable could drive both.

Solution: We could believe that hours online negatively affects yearly wages if we believe people who spend all their time online neglect other work that could help them get ahead. Or, we could believe that people who make a lot of money can buy things in the real world, and therefore don't need to spend as much time online to entertain themselves. Or, we could believe that people who work longer hours both make more money and have less time to spend online; thus working hours could be a hidden variable driving both.

iv. Experimental interventions can sometimes determine causation when passive observations are inconclusive. Suggest one experiment we could perform to test one of the explanations you just gave. (You can assume we have all the money and/or influence we need to run this experiment.)

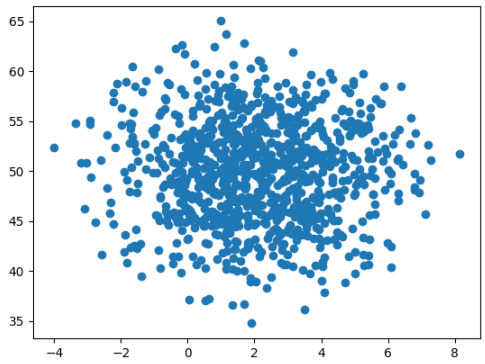
Experiments to test all three explanations (only one is necessary): We could limit one group's time online versus a control, and see whether the time-limited group's income rises after a year or two. We could give people extra money, and see whether their time online goes down relative to a control. Or we could give people extra work and working hours, and see whether their time online goes down.

Problem 2 (15 points [5 each])

For each 2D Gaussian distribution, find the covariance matrix (reporting it in your PDF), then plot 1000 random samples with `numpy.random.multivariate_normal()` and `matplotlib.pyplot.scatter()` (including the plot in your PDF). In each case, the mean of X is 2, the mean of Y is 50, the variance of X is 4, and the variance of Y is 25.

i. X and Y are independent. (Notice that matplotlib rescales each axis independently based on the data; you don't have to fix that.)

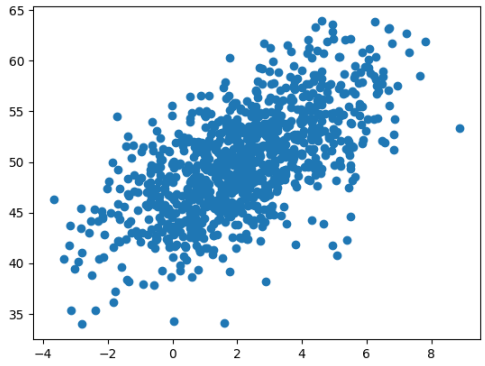
Solution: The covariance matrix is just the diagonal matrix of variances. $\begin{bmatrix} 4 & 0 \\ 0 & 25 \end{bmatrix}$



ii. X and Y are correlated, Pearson's $r = 0.6$.

Solution: We need to work backwards to figure out the covariance first. Our product of standard deviations is $2 * 5 = 10$, so if cov(X,Y)/10 = 0.6, then cov(X,Y) = 6. So the covariance matrix is

$$\begin{bmatrix} 4 & 6 \\ 6 & 25 \end{bmatrix}$$

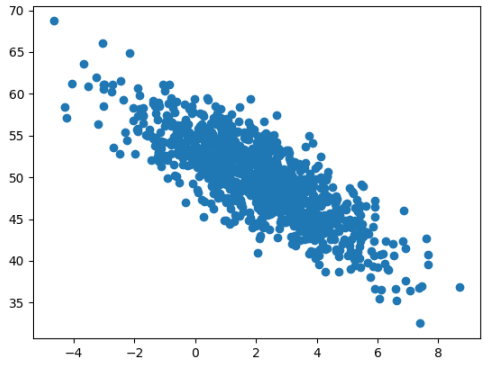


iii. X and Y are negatively correlated, Pearson's $r = -0.8$.

Solution:

Again, we need to work backwards to figure out the covariance first. Our product of standard deviations is still $2 * 5 = 10$, so if cov(X,Y)/10 = -0.8, then cov(X,Y) = -8. So the covariance matrix is

$$\begin{bmatrix} 4 & -8 \\ -8 & 25 \end{bmatrix}$$



Problem 3 (15 pts [5 each])

For each covariance matrix in the previous problem, find the eigenvalues, then find the eigenvector in the direction of greatest variance. (You may need to use the quadratic formula to solve for the eigenvalues.)

Solution:

(i) The covariance matrix here is already diagonalized, without a change of basis, so the eigenvalues are 4 and 25, and the corresponding eigenvectors are in the directions of $\begin{bmatrix}1\\0\end{bmatrix}$ and $\begin{bmatrix}0\\1\end{bmatrix}$. Since the variance in the Y direction is bigger, that's the direction of greatest variance.

(ii) We have $\det(\lambda I - A) = \det(\begin{bmatrix}\lambda-4 & -6\\ -6 & \lambda-25\end{bmatrix}) = 0$, which gives us the polynomial $\lambda^2 - 29\lambda + 64 = 0$. Quadratic equation gives us $\frac{29 \pm \sqrt{(29^2)-256}}{2}$ which has the solutions $(29 \pm 24.2)/2 = 2.4, 26.6$. We only care about the larger eigenvector, so we solve $\left[\begin{array}{cc|c} 26.6-4 & -6 & 0\\ -6 & 26.6-25 & 0\end{array}\right]$ and get a vector in (roughly) the direction $\begin{bmatrix}1\\3.75\end{bmatrix}$

(iii) We have $\det(\lambda I - A) = \det(\begin{bmatrix}\lambda-4 & 8\\ 8 & \lambda-25\end{bmatrix}) = 0$, which gives us the polynomial $\lambda^2 - 29\lambda + 36 = 0$. Quadratic equation gives us $\frac{29 \pm \sqrt{(29^2)-144}}{2}$ which has the solutions $(29 \pm 26.4)/2 = 1.3, 27.7$. We only care about the larger eigenvector, so we solve $\left[\begin{array}{cc|c} 27.7-4 & 8 & 0\\ 8 & 27.7-25 & 0\end{array}\right]$ and get a vector in the direction $\begin{bmatrix}1\\-2.96\end{bmatrix}$

Problem 4 (15 pts [3, 3, 4, 5])

Use the provided `read_two_columns()` function in `ratings.py` to answer the following questions. (The function assumes that whenever two columns of data are involved, we only care about users where both fields are populated and nonzero.)

The data comes from `https://archive.ics.uci.edu/ml/datasets/Travel+Reviews`. (Their documentation is 1-indexed when referring to columns, but my instructions are zero-indexed.)

They're average Google ratings on a 5-point scale of art galleries, museums, and so on (see the webpage for complete details), where a particular entry is a user's average rating for that category.

i. Find the correlation between users' juice bar ratings (column 3) and their restaurant ratings (column 4). (`numpyp.corrcoef()` returns a matrix where entry (i, j) is the correlation between variable i and variable j.)

Solution: 0.396

ii. Test the hypothesis that users give museums (column 5) significantly higher ratings than they do restaurants (column 4). Your target p-value is 0.05 as usual; however, since we are really just examining a within-user difference instead of looking across groups, the variance is less and we can use `scipy.stats.ttest_rel()`. What is the effect size (difference between means)?

Solution: This is significant (p is 2.2×10^{-25}) and the effect size is 2.96-2.80=0.16.

iii. Fit a least-squares linear model that predicts average beach rating (column 8) from average park rating (column 7). Report the correlation coefficient as well as the equation $y = mx + b$.

Solution: The correlation is 0.41, and the linear fit is $0.32x + 1.47$.

iv. Find a confidence interval for the true mean of all average user ratings of restaurants (column 4; use `scipy.stats.t.interval()` if you like; you can use the column for both arguments to `read_two_columns()` to omit the zeros).

Solution: scipy.stats.t.interval(0.95, len(rest)-1, loc=np.mean(rest), scale=scipy.stats.sem(rest)) gives [0.276, 0.283].

Problem 5 (30 points extra credit [10,20])

The other data file posted for the homework is a series of data points to which you will fit a curve of the form $f(x) = 1/(ax + b)$. (This curve, called a *hyperbolic discounting curve*, appears in models of how people value future rewards; the value drops off rapidly at first, then stabilizes.)

i. Assume we care about minimizing the squared error as our loss function. Find the derivative with respect to a of the loss as we try to fit a function of the form $f(x) = 1/(ax + b)$. Then find the derivative with respect to b of the loss. (You will need to use the chain rule. Study the backpropagation derivation from lecture as a guide for what to do.)

Solution: The derivative with respect to a is $\frac{-2xErr}{(ax+b)^2}$ while the derivative with respect to b is $\frac{-2Err}{(ax+b)^2}$. ("Err" is the error that is squared for the loss function.)

ii. Implement gradient descent to find the best fit of a function of the form $f(x) = 1/(ax + b)$ to the noisy data that has been provided. This should be your own implementation, using the equations you derived in the previous part. (I had success with starting values of $a = 1$ and $b = 1$, a learning rate of 0.1, and quite a lot of iterations, where a single iteration consisted of sampling a point at random and applying gradient descent to the error.) In your PDF, turn in the best values for a and b that your approach produced, and a plot showing the fit. Turn in your Python code as well.

Solution: See Python file. The true parameters were $a = 2.5, b = 3.1$.