**1a**) Q1 = 82, Median = 89, Q3 = 95

**1b**) Mean = 87.011

**1c**) Mode = 95

**1d**) The data is positively skewed. This is because the mean < median < mode. 87.011 < 89 < 95.

**2a**)



**2b**) A = (3, 1, 2) B = (-1, 0, 8)

**2b1**)



S









**2b2**)









**2b3**)











**2c**) The Euclidean distance is a direct line between 2 points in space. The Manhattan distance is the distance between 2 points in space if the path traveled from A to B is taken at right angles. If we look at these two distances in a 2-d plane, the Manhattan distance will always be the 2 shorter sides of a right triangle, whereas the Euclidean distance will be the hypotenuse. This is all assuming the points are not already on a 90 degree axis with each other, which would mean the Manhattan and Euclidean distances would be equal.

**2d1**) h = 2: 412.941

**2d2**) h = 3: 216.448

**3a**)

*Before*

Mean = 76.814

Variance = 171.396

*After*

Mean = 0

Variance = 1

**3b**)

Original Value = 90

Z-Score = 1.007

**4a**)





Correlation Coefficient = .985. The 2 vectors in the data set are positively correlated, meaning as X changes, Y changes in the same direction with a similar magnitude.

**4b**) PCA will help to reduce the data size because the correlation coefficient is high. The higher the correlation coefficient, the more redundancy exists in a data set. Because there is so much redundancy, PCA will be effective in reducing the data set size.

**4c**)

Step 1: Zero mean X and Y using eq: **xZM(i) = x(i) – xMean for i in X**. Use same for Y.



Step 2: Covariance Matrix = 



**4d**) Calculations in script below:

2 Principal components because matrix is 2x10 (MxN) where the # of PC’s are M.

Principal Components:

0.6885 -0.7253 -> P1

-0.7253 -0.6885 -> P2

Most important Principal Component:

-0.7253 -0.6885 -> P2

This is because the covariance matrix looks like this…

0.0086 -0.0000

-0.0000 1.1229 -> Variance is much higher on P2 than P1 meaning more information is retained by using P2

Code:

x = [.69, -1.31, .39, .05, 1.29, .49, .19, -.81, -.31, .71];

y = [.89, -1.11, .59, .45, 1.19, .69, .25, -.71, -.21, .71];

xMean = mean(x);

yMean = mean(y);

xNum = length(x);

% Make X and Y zero mean

for i = 1:numel(x)

xM(i) = (x(i) - xMean); % Zero Mean

yM(i) = (y(i) - yMean); % Zero Mean

end

% Combine them together to form an 2x10 (MxN) matrix

xyM = [xM; yM]

% Find the covariance matrix

xyCov = (1/(xNum-1))\*xyM\*xyM'

% Find the Eigen vectors for the covariance matrix

[e\_vec,e\_val] = eig(xyCov);

e\_vec = e\_vec

% Find the new principal component matrix

eY = e\_vec \* xyM

% Ensure Covariance of eY is diagonal matrix

cY1 = (1/(xNum-1))\*eY\*eY'

% Variance of the second vector is greater so that will be the principal

% component

**4e**)



Red line = Largest principal component direction

Orange line = Smaller principal component direction

**4f**)

Ap = P1 \* A = -.346

Bp = P1 \* B = -.830

For projecting A onto the graph I used



Where a = A (Original data point), b = P1 (Primary component of PCA)



Red circle denotes projection of A

Black circle denotes projection of B