Clarification and correction

Correction

Question 1 part b should be corrected as follows:

b) Write a function which takes F and E as the inputs and returns B as the output. (Submit your function for this part)

Clarification

- The input of MDS algorithm should be squared distance.
- You can refer to "FastMap, MetricMap, and Landmark MDS are all Nystrom Algorithms" by Johan C. Platt In order to implement LMDS.
- Equation (1) in the above paper can be implemented as follows in Matlab:

```
K=-.5*(D.^2-ones(N,1)*sum(D.^2,1)/N-sum(D.^2,2)*ones(1,N)/N+sum(sum(D.^2,1))/(N^2))
```

  where $ones(N,1)$ and $ones(1,N)$ are e_j and e_i. c_i and c_j are both $1/N$ ($c_i c_j$ *is* $1/N^2$) , and $\sum_i$,($\sum_j$) and $\sum_{i,j}$ are implemented by functions $sum(\ ,1)$, $sum(\ ,2)$ and $sum(sum(\ ,1))$.
- Note that $ones(2,1) * sum(D,1)$ for example is equivalent to $repmat(sum(D,1),2,1)$.
- You can find A now using equation (13);
- B can be computed by either (14) or (15).
- The paper "Sparse multidimensional scaling using landmark points" (find it in the web page) provides Pseudocode for LMDS. You can alternatively use this Pseudocode to implement LMDS. If you chose to do so you can meagre parts *a*), *b*) and *c*) of Question 1. That is you can use pseudocode from the paper and write a function which takes E , F and d as the inputs and computes the d-dimensional representation by LMDS.