DATA 505 – Computational data science and analytics

Project 1 – Latent Semantic Analysis (LSA)

Collaboration Policy: Individual Assignment

Total Points: 100 (See rubric below)

Project Topic: In this project you'll recreate the results provided in the *Deerwester et al. 1990* LSA paper (located in the table of contents, paper tab, on OAKS) that attempts to identify hidden, or latent, relationships between terms (or words) in document titles.

Approved software tools: You may only use the Python programming language and the following technologies/software libraries: 1. Python notebook, 2. NumPy, 3. MatPlotLib, and 4. SciPy.

Submission: Submit your notebook to the Dropbox setup on OAKs before the due date. The name of your notebook must be, P1_<lastname>.ipynb, where <lastname> is your last name ☺. For example, *P1_Munsell.ipynb* would be correct if I were to submit the assignment. Please, only use one notebook to complete this assignment, i.e. no other Python files (.py files) are allowed.

Grading Rubric:

Python notebook compiles and runs with no syntax errors	5 points
Your notebook must include a markdown section that describes what each cell is accomplishing. For instance, what is the cell computing or visualizing, and what mathematical, statistical, or visualization techniques is being applied.	15 points
In each notebook cell, include comments that provide more in-depth information about how the computation is performed (i.e. step-by-step approach) or how the data is stored in a particular array or matrix.	10 points
Given the term/document matrix in TABLE 2, recreate the mathematical space shown in the 2-D Plot of Terms and Docs from Example (see FIG 1).	30 points
Create five different pseudo-documents (X_q) and then insert each pseudo-document into the FIG 1 2-D Plot (see section titled "Finding Representations for Pseudo-Documents") using the provided matrix multiplication $D_q = X_q' T S^{-1}$. Note: when you insert a pseudo-document in to the plot, please use a different color for quick and easy identification.	20 points
For each pseudo-document inserted into the 2-D Plot, find the three closest documents, and the three closest terms, using the cosine distance similarity measure. Note: the results must be clearly displayed in a notebook cell (i.e. they don't have to be visualized in the 2-D plot).	20 points
	100 points