**Project 1:**

Texto

Descripción generada automáticamente

**Steps:**

The header in which we restart and clean MATLAB:



We define the function "f" and add the subspace as a polynomial of degree 3 "g":

Texto

Descripción generada automáticamente

To calculate the orthogonal projection of the functions "f" on "g" we need to calculate the norm with definite integrals on the given interval:Texto

Descripción generada automáticamente

And finally plot the results in the given interval:

Interfaz de usuario gráfica, Texto

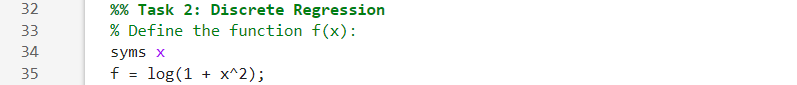
Descripción generada automáticamente

Result:

Gráfico, Gráfico de líneas

Descripción generada automáticamente

We create a symbolic variable and define the function "f":



We create 1000 points between the borders and plot the original "f":

Texto

Descripción generada automáticamente

We generate the A matrix containing the information of each point in 1 row.

And solve the system Ac = b to find in c all the approximated coefficients.

Texto

Descripción generada automáticamente

We define the approximation function, and plot it:

Texto

Descripción generada automáticamente

Result:

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**I've learned?**

I have learned that there are different types of regressions and that each one is useful in its own way by using symbolic variables, some use points to fit the function as closely as possible (discrete regression), others use integrals to get as close as possible to the main function (continuous regression).

I can also see all my work by plotting the results in Matlab.

**Project 2:**

Texto, Carta

Descripción generada automáticamente

**Steps:**

The header in which we restart and clean MATLAB:



We generate a random matrix M for a web page graph with N nodes:

Imagen que contiene Texto

Descripción generada automáticamente

We create the matrices T (each column must add 1), B and the Google one:

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

We plot the eigenvalues to show that λ = 1 is the biggest:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Result:

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

We approximate lambda by the power method and a tolerance of 10^-10:

Texto

Descripción generada automáticamente

Result:

Imagen que contiene Gráfico

Descripción generada automáticamente

We calculate the Page Rank:  
Texto

Descripción generada automáticamente

We calculate the popularity and compare it to the Page Rank:

Texto

Descripción generada automáticamente

While trying to make the program, I found interesting that, it tends to be more important to sort by inputs than by the sum of both, after trying a few times, although in some the sum works better, to a greater extent the input has the most approximate page rank. There are times when there is a lot of input and almost no output, so in the sum it is lower, and for the page rank it is very important.

Example: Columns (Rank, PageRank, Popularity by sum, Popularity by input)  
Tabla

Descripción generada automáticamenteGráfico

Descripción generada automáticamente con confianza bajaImagen que contiene Gráfico

Descripción generada automáticamente

As you can see in the **last** one, the less important **for the Page Rank is the 44 as well as for the popularity by input.**

Also in the **top 3**: The **popularity by input** and the **Page Rank have the same 3**, while the **Popularity by sum** **has the 26 in the rank 7** and **the 40 in the rank 17**.

In conclusion, I think that the **Popularity sorted by the input is more precise** than the popularity sorted by the sum in most part of the times for great quantity of pages. It is useful to have both, but the input sorted one is more powerful.

**Project 3:**

Texto

Descripción generada automáticamente

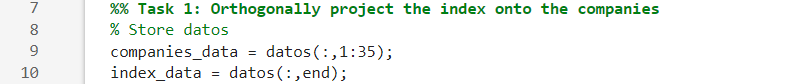
**Steps:**

The header in which we restart and clean MATLAB and upload the data:

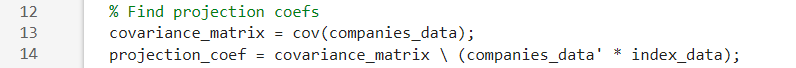
Imagen que contiene Texto

Descripción generada automáticamente

We store the data in a comfortable way:



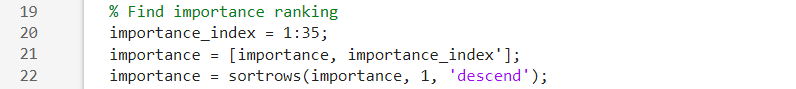
We calculate the projection coefficients with the covariance matrix:



We calculate the importance of each company:



We sort the ranking:



Plot the ranking:

Texto

Descripción generada automáticamente

Result:

|  |  |  |
| --- | --- | --- |
| Rank | Company | Importance(%) | 12 | IDR.MC | 1.8177 | 24 | FDR.MC | 0.4423 |
| 1 | UNI.MC | 27.5420 | 13 | COL.MC | 1.7396 | 25 | MTS.MC | 0.4072 |
| 2 | SCYR.MC | 19.5983 | 14 | ACS.MC | 1.7035 | 26 | FER.MC | 0.3846 |
| 3 | MAP.MC | 12.7833 | 15 | SAN.MC | 1.3919 | 27 | NTGY.MC | 0.2936 |
| 4 | RED.MC | 5.0658 | 16 | ENG.MC | 1.2598 | 28 | ITX.MC | 0.2828 |
| 5 | IAG.MC | 4.0734 | 17 | ELE.MC | 0.6458 | 29 | ROVI.MC | 0.2579 |
| 6 | BKT.MC | 3.5541 | 18 | ANE.MC | 0.6157 | 30 | MRL.MC | 0.2297 |
| 7 | IBE.MC | 3.2433 | 19 | TEF.MC | 0.5419 | 31 | CABK.MC | 0.1264 |
| 8 | SLR.MC | 3.1370 | 20 | REP.MC | 0.4966 | 32 | ACX.MC | 0.1263 |
| 9 | SAB.MC | 2.4276 | 21 | CLNX.MC | 0.4856 | 33 | AENA.MC | 0.1137 |
| 10 | LOG.MC | 2.1179 | 22 | BBVA.MC | 0.4448 | 34 | ANA.MC | 0.0758 |
| 11 | GRF.MC | 2.0937 | 23 | AMS.MC | 0.4445 | 35 | MEL.MC | 0.0358 |

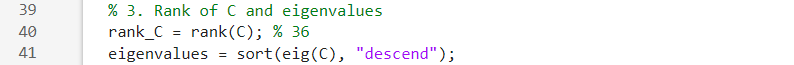
We calculate the centered matrix:



We calculate the covariance matrix of the centered matrix:



We check the rank, calculate the eigenvalues and sort them:



We calculate the eigenvectors:

Texto

Descripción generada automáticamente

We check if it is orthonormal:



Result:

Imagen que contiene Texto

Descripción generada automáticamente

We project Xc onto V, by matrix multiplication:



With the 3 principal PCA, we plot the 2D one (The index in red):

Texto

Descripción generada automáticamente

Result:

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

They appear 3 principal clusters, and I think that the maximum is 3 due to the V that has rank 3, if we take 4 eigenvectors, and plot them in a 4D space, they might be in 4 different clusters.

And we plot the 3D one (The index in red):

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Result:

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

The same happens in a 3D space, only 3 clusters appear, and if you rotate the graph, you can see that the clusters in the plane (PCA 2- PCA 3) and (PCA 1- PCA 3) are not clear. And in the 3D graph you can clearly see them.