CS3081: Assignment 2

Conor McCauley - 17323203

March 25, 2020

Answers

Question 4.23

(ii)

$$L = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ -2.0 & 1.0 & 0.0 & 0.0 \\ 0.5 & 1.5 & 1.0 & 0.0 \\ -2.0 & 3.0 & -0.5 & 1.0 \end{bmatrix}$$

$$U = \begin{bmatrix} 4.0 & -1.0 & 3.0 & 2.0 \\ 0.0 & -2.0 & 3.0 & 0.5 \\ 0.0 & 0.0 & 4.0 & 2.0 \\ 0.0 & 0.0 & 0.0 & 3.0 \end{bmatrix}$$

Question 5.17

Best: teams #2 and #5 are tied. Worst: team #1.

Question 6.3

(iii) $b = 4.6931 \cdot 10^{-8}$, $m = 0.012$, $population(1985) = 1038\, million$.

Solutions

Question 4.23

The following Matlab code calculates the lower and upper triangular matrices of a given square matrix using the Gaussian method and displays them:

```
A = [
    4 -1 3 2;
    -8 0 -3 -3.5;
    2 -3.5 10 3.75;
    -8 -4 1 -0.5
];
[L, U] = LUdecompGauss(A);
disp(L);
disp(U);

function [L, U] = LUdecompGauss(A)

    % get dimensions of A
    [N, ~] = size(A);

    % initialise the triangular matrices
    L = Identity(N);
    U = A;

    % for each row in U except the last
    for i = 1:(N - 1)

        % for each subsequent row
        for j = (i + 1):N

            % constant multiplier
            m = U(j, i) / U(i, i);
```

```
            % for each column in row j
            for k = 1:N
                % update upper triangular matrix
                U(j, k) = U(j, k) - (U(i, k) * m);
            end

            % update lower triangular matrix
            L(j, i) = m;

        end

    end

end

% return the identity matrix of size N
function [Imat] = Identity(N)
    Imat = zeros(N, N);
    for i = 1:N
        Imat(i, i) = 1;
    end
end
```

Question 5.17

The following code solves parts (a) and (b) of the question.

```
A = [
    0 0 0 1 0 0;
    1 0 1 0 1 1;
    0 1 0 0 1 0;
    1 1 0 0 1 0;
    1 1 1 0 0 1;
    1 0 0 0 1 0
];

% (a)
```

```
% extract Eigenvalues
E = eig(A);
% display the real component of the values
disp(real(E));

% (b)
% extract Eigenvectors
[V, ~] = eig(A);
% only the first column contains only real vectors of the same sign
disp(V(:, [1]));
```

For part (b) the following values are displayed:

$$[0.1761, 0.5155, 0.3938, 0.4611, 0.5155, 0.2642]$$

The best teams (largest values) are teams #2 and #5. The worst team (least value) is team #1.

Question 6.3

We are given the following population table and told that population growth can be modeled using the function $p = be^{mx}$:

Year	1900	1950	1970	1980	1990	2000	2010
Population (mills)	400	557	825	981	1135	1266	1370

First we need to write the equation $p = be^{mx}$ in linear form:

$$\begin{aligned} p &= be^{mx} \\ \ln(p) &= \ln(be^{mx}) \\ \ln(p) &= \ln(b) + \ln(e^{mx}) \\ \ln(p) &= \ln(b) + mx \end{aligned}$$

Now we can use linear least-squares regressions to approximate b and m. We must first calculate the following sums: $\sum \ln(p)x$, $\sum x^2$, $\sum x$ and $\sum \ln(p)$.

$$\sum \ln(p)x = (\ln(400) \cdot 1900) + \cdots + (\ln(1370) \cdot 2010) = 93384.48848$$

$$\sum x^2 = 1900^2 + \cdots + 2010^2 = 27214000$$

$$\sum x = 1900 + \cdots + 2010 = 13800$$

$$\sum \ln(p) = \ln(400) + \cdots + \ln(1370) = 47.31855$$

Using these sums we can calculate m using the following formula (where N is the number of provided points):

$$\begin{aligned} m &= \frac{(N \cdot \sum \ln(p)x) - (\sum x \cdot \sum \ln(p))}{(N \cdot \sum x^2) - (\sum x)^2} \\ &= \frac{(7 \cdot 93384.48848) - (13800 \cdot 47.31855)}{(7 \cdot 27214000) - 13800^2} \\ &= 0.01198 \end{aligned}$$

We can then calculate $\ln(b)$ and b like so:

$$\begin{aligned} \ln(b) &= \frac{\sum p - m \cdot \sum x}{N} \\ &= \frac{47.31855 - 0.01198 \cdot 13800}{7} \\ &= -16.87458 \end{aligned}$$

$$b = \exp(-16.87458) = 4.69312 \cdot 10^{-8}$$

The population in 1985 can now be easily estimated:

$$p = (4.6931 \cdot 10^{-8}) \cdot e^{0.012 \cdot 1985} = 1038.375 = 1038\, million$$