Completion of assignment 2 task 3 – Numeriska metoder

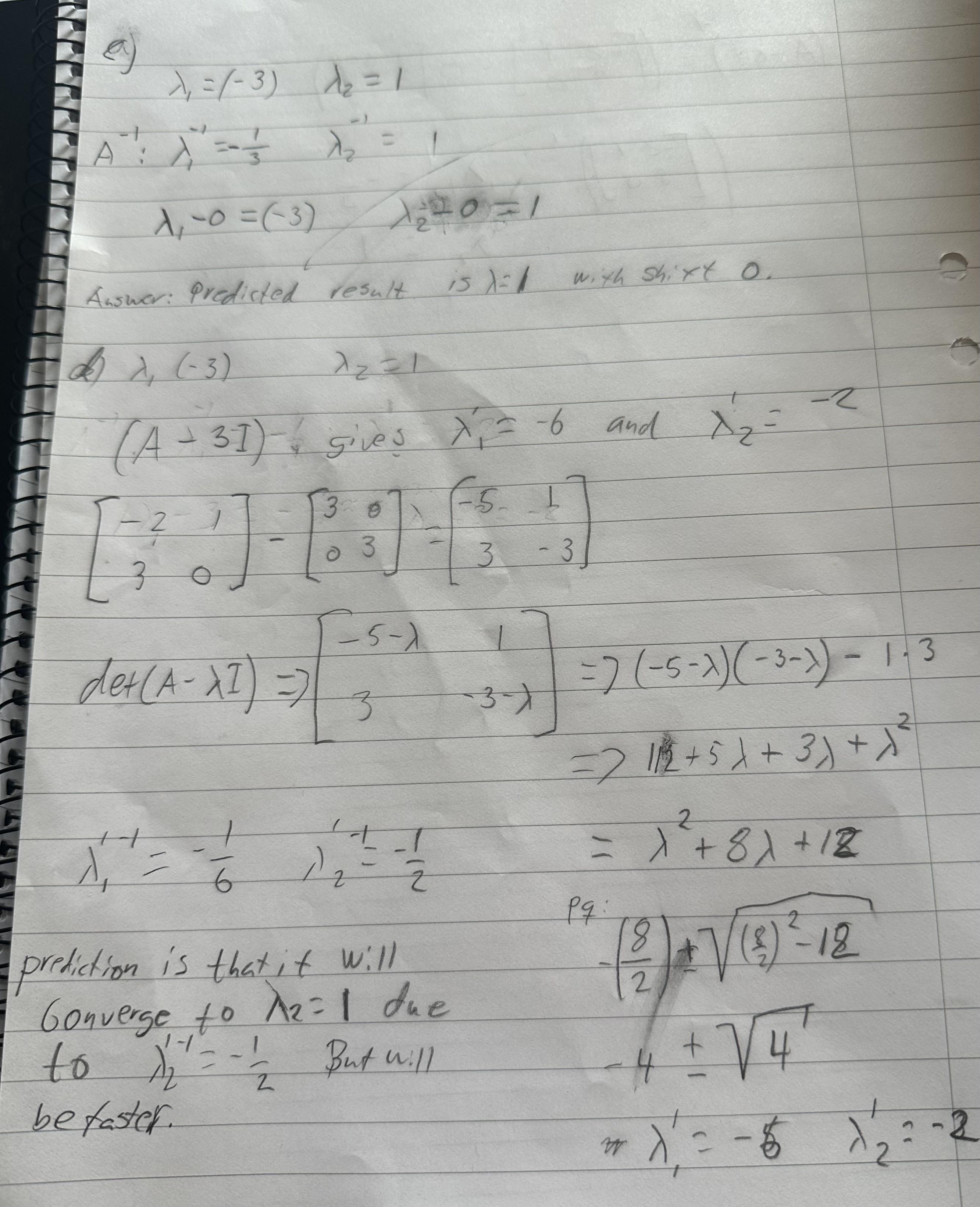
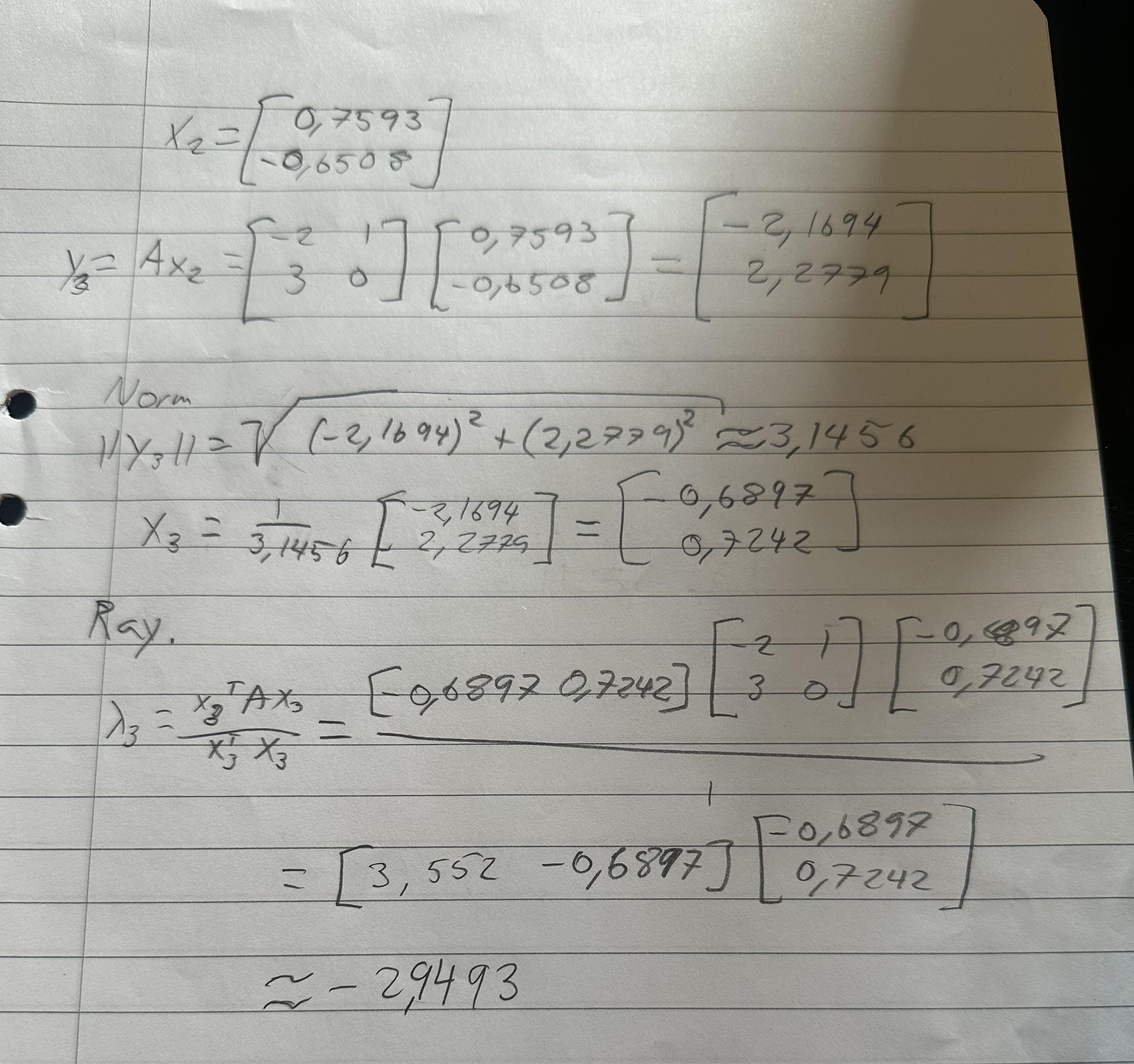
1MA930

Jesper Wingren

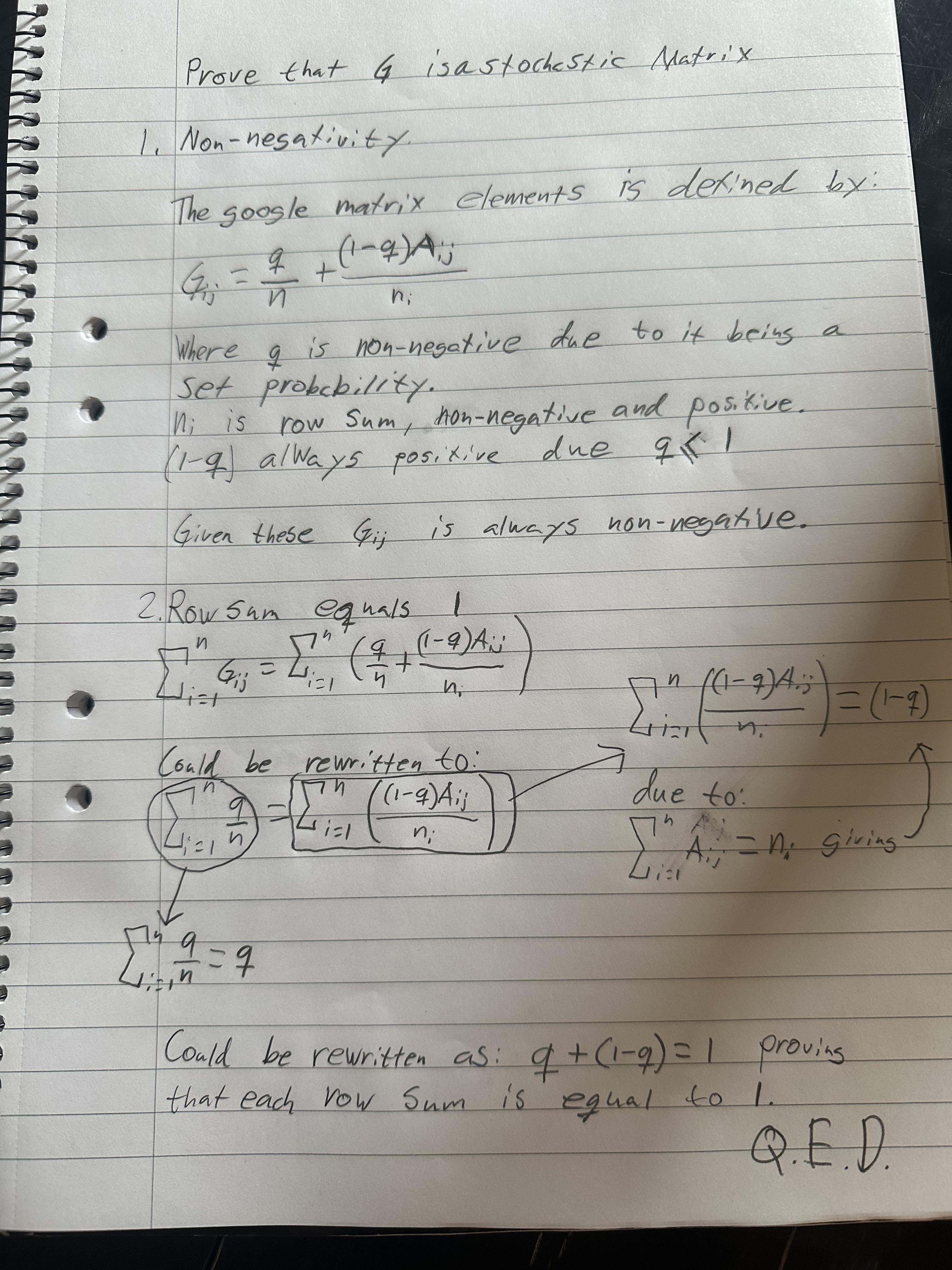
Emil Ulvagården

Samuel Berg

Task 2.



Task 6.



Task 7.

Matrix G:

0.0100 0.0100 0.0100 0.0100 0.4350 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.4350 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.0100 0.2933 0.0100 0.4350 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.4350 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.0100 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100

0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.2933 0.0100 0.0100 0.0100

0.0100 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.2933 0.0100 0.0100 0.0100

0.4350 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.4350 0.0100 0.0100

0.0100 0.0100 0.0100 0.0100 0.4350 0.4350 0.4350 0.0100 0.2933 0.0100 0.0100 0.0100 0.0100 0.2225 0.0100

0.0100 0.0100 0.0100 0.0100 0.0100 0.4350 0.4350 0.4350 0.0100 0.0100 0.0100 0.2933 0.0100 0.2225 0.0100

0.0100 0.0100 0.0100 0.4350 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.4350

0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.8600 0.0100 0.0100 0.0100 0.2225 0.0100

0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.4350 0.0100 0.4350

0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.0100 0.8600 0.0100 0.0100 0.2225 0.0100

Below is the code for verifying the dominant eigenvector p that is given in the book. Using Gp = p and comparing the results. The given program gives the output verified.

A = [

0 1 0 0 0 0 0 0 1 0 0 0 0 0 0

0 0 1 0 1 0 1 0 0 0 0 0 0 0 0

0 1 0 0 0 1 0 1 0 0 0 0 0 0 0

0 0 1 0 0 0 0 0 0 0 0 1 0 0 0

1 0 0 0 0 0 0 0 0 1 0 0 0 0 0

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0

0 0 0 1 0 0 0 0 0 0 1 0 0 0 0

0 0 0 0 1 1 0 0 0 1 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0 1 0 0

0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

0 0 0 0 0 0 1 1 0 0 1 0 0 0 0

0 0 0 0 0 0 0 0 1 0 0 0 0 1 0

0 0 0 0 0 0 0 0 0 1 1 0 1 0 1

0 0 0 0 0 0 0 0 0 0 0 1 0 1 0];

q = 0.15;

[rows, cols] = size(A);

G = zeros(rows, cols);

for i = 1:rows

% Get sum of row i

ni = sum(A(i, :));

for j = 1:cols

G(j, i) = q/15 + (((1 - q) \* A(i, j)) / ni);

end

end

G;

p = [0.0268; 0.0299; 0.0299; 0.0268; 0.0396; 0.0396; 0.0396; 0.0396; 0.0746; 0.1063; 0.1063; 0.0746; 0.1251; 0.1163; 0.1251];

result = G \* p;

tolerance = 1e-4;

if max(abs(result - p)) < tolerance

disp("Verified p = G \* p");

end

Task 8.

format short;

A = [

0 1 0 0 0 0 0 0 1 0 0 0 0 0 0;

0 0 1 0 1 0 1 0 0 0 0 0 0 0 0;

0 1 0 0 0 1 0 1 0 0 0 0 0 0 0;

0 0 1 0 0 0 0 0 0 0 0 1 0 0 0;

1 0 0 0 0 0 0 0 0 1 0 0 0 0 0;

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0;

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0;

0 0 0 1 0 0 0 0 0 0 1 0 0 0 0;

0 0 0 0 1 1 0 0 0 1 0 0 0 0 0;

0 0 0 0 0 0 0 0 0 0 0 0 1 0 0;

0 0 0 0 0 0 0 0 0 0 0 0 0 0 1;

0 0 0 0 0 0 1 1 0 0 1 0 0 0 0;

0 0 0 0 0 0 0 0 1 0 0 0 0 1 0;

0 0 0 0 0 0 0 0 0 1 1 0 1 0 1;

0 0 0 0 0 0 0 0 0 0 0 1 0 1 0];

n = size(A, 1);

qs = [0, 0.15, 0.5];

vectors = zeros(n, length(qs));

c = 0;

for q = qs

c = c + 1;

G = zeros(n, n);

for i = 1:n

ni = sum(A(i, :));

for j = 1:cols

G(j, i) = q/15 + (((1 - q) \* A(i, j)) / ni);

end

end

% Finding dominant eigenvector using eig

[V, D] = eig(G);

[~, idx] = max(diag(D));

principalEigenvector = V(:, idx);

% Normalize

principalEigenvector = principalEigenvector / sum(principalEigenvector);

vectors(:, c) = principalEigenvector;

end

fprintf("Page ranks for q equals:\n 0 0.15 0.5\n");

fprintf(" --------------------------\n")

disp(vectors);

En bild som visar text, skärmbild, Teckensnitt, nummer

Automatiskt genererad beskrivning

Jump probability is the percentual chance of a surfer jumping to a

random page instead of following a link on another page.

q = 0 means that the next jump is not random. The surfer strictly follows

from one page to another via the link meaning some pages will not be

reached giving the page rank 0.

q = 0.15 means that there is a 15 % probability of the surfer jumping to

a random page instead of following a link. This distributes the

probability of reaching pages more evenly.

q = 0.5 means that it is as likely that the surfer follows a link as it is going to a random page. This reduces the influence of the page links and pages with few, or no in-bound links has a greatly higher rank that they

have with a lower q value.

Task 9.

Code for reality check 4.

A = [

0 1 0 0 0 0 0 0 1 0 0 0 0 0 0

0 0 1 0 1 0 2 0 0 0 0 0 0 0 0

0 1 0 0 0 1 0 1 0 0 0 0 0 0 0

0 0 1 0 0 0 0 0 0 0 0 1 0 0 0

1 0 0 0 0 0 0 0 0 1 0 0 0 0 0

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0

0 0 0 0 0 0 0 0 0 1 1 0 0 0 0

0 0 0 1 0 0 0 0 0 0 1 0 0 0 0

0 0 0 0 1 1 0 0 0 1 0 0 0 0 0

0 0 0 0 0 0 0 0 0 0 0 0 1 0 0

0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

0 0 0 0 0 0 2 1 0 0 1 0 0 0 0

0 0 0 0 0 0 0 0 1 0 0 0 0 1 0

0 0 0 0 0 0 0 0 0 1 1 0 1 0 1

0 0 0 0 0 0 0 0 0 0 0 1 0 1 0];

q = 0.15;

[rows, cols] = size(A);

G = zeros(rows, cols);

for i = 1:rows

% Get sum of row i

ni = sum(A(i, :));

for j = 1:cols

G(j, i) = q/15 + (((1 - q) \* A(i, j)) / ni);

end

end

[V, D] = eig(G);

[~, idx] = max(diag(D));

principalEigenvector = V(:, idx);

% Normalize

principalEigenvector = principalEigenvector / sum(principalEigenvector);

oldp = [0.0268

0.0299

0.0299

0.0268

0.0396

0.0396

0.0396

0.0396

0.0746

0.1063

0.1063

0.0746

0.1251

0.1163

0.1251];

differences = principalEigenvector - oldp;

nodes = (1:length(oldp))';

comparisonTable = table(nodes, principalEigenvector, oldp, differences, ...

'VariableNames', {'Page', 'UpdatedPageRank', 'OldPageRank', 'Difference'});

disp(comparisonTable);

En bild som visar text, meny, skärmbild, Teckensnitt

Automatiskt genererad beskrivning

Code and outputs for reality check 5.

A = [

0 1 0 0 0 0 0 0 1 0 0 0 0 0 0;

0 0 1 0 1 0 1 0 0 0 0 0 0 0 0;

0 1 0 0 0 1 0 1 0 0 0 0 0 0 0;

0 0 1 0 0 0 0 0 0 0 0 1 0 0 0;

1 0 0 0 0 0 0 0 0 0 0 0 0 0 0;

0 0 0 0 0 0 0 0 0 0 1 0 0 0 0;

0 0 0 0 0 0 0 0 0 0 1 0 0 0 0;

0 0 0 1 0 0 0 0 0 0 1 0 0 0 0;

0 0 0 0 1 1 0 0 0 0 0 0 0 0 0;

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0;

0 0 0 0 0 0 0 0 0 0 0 0 0 0 1;

0 0 0 0 0 0 1 1 0 0 1 0 0 0 0;

0 0 0 0 0 0 0 0 1 0 0 0 0 1 0;

0 0 0 0 0 0 0 0 0 0 1 0 1 0 1;

0 0 0 0 0 0 0 0 0 0 0 1 0 1 0];

q = 0.15;

[rows, cols] = size(A);

G = zeros(rows, cols);

for i = 1:n

ni = sum(A(i, :));

if ni == 0

G(:, i) = q / n + (1 - q) / n;

else

for j = 1:n

G(j, i) = q / n + ((1 - q) \* A(i, j) / ni);

end

end

end

[V, D] = eig(G);

[~, idx] = max(diag(D));

principalEigenvector = V(:, idx);

% Normalize

principalEigenvector = principalEigenvector / sum(principalEigenvector);

oldp = [0.0268

0.0299

0.0299

0.0268

0.0396

0.0396

0.0396

0.0396

0.0746

0.1063

0.1063

0.0746

0.1251

0.1163

0.1251];

differences = principalEigenvector - oldp;

nodes = (1:length(oldp))';

comparisonTable = table(nodes, principalEigenvector, oldp, differences, ...

'VariableNames', {'Page', 'UpdatedPageRank', 'OldPageRank', 'Difference'});

disp(comparisonTable);

En bild som visar text, skärmbild, meny, Teckensnitt

Automatiskt genererad beskrivning

Reflections on reality check 4 and 5:

As for check 4, yes this succeeds we can see an increase of the probability for page 7 as intended. Meanwhile most of the other page’s ranks decrease which is reasonable if one increases. 9, 10, 13, 14 also has a small increase in page rank. But all other ranks other than 7 only differs by less than 1 % which is a small percentage. And for check 5, By removing all links to and from page 10 it decreases the chance of a surfer reaching that page by approximately 9.5 % which is a lot. This also changes the other pages ranks increasing all pages except 9, 13, 14. By having the two results of 4 and 5 we can see that by adding more links it increases for 9, 10, 13, 14 and if removing links, it decreases the page ranks for the same. This can conclude to 9, 10, 13, 14 benefiting from more links while it doesn’t benefit from removing links generally for the matrix.