IMPERIAL

MATH70024

BSc, MSci and MSc EXAMINATIONS (MATHEMATICS)
May 2024

This paper is also taken for the relevant examination for the
Associateship of the Royal College of Science

Computational Linear Algebra

Date: Thursday, May 30, 2024
Time: 10:00 – 12:30 (BST)
Time Allowed: 2.5 hours
This paper has 5 Questions.
Please Answer All Questions in 1 Answer Booklet
Candidates should start their solutions to each question on a new sheet of paper.
Supplementary books may only be used after the relevant main book(s) are full.
Any required additional material(s) will be provided.
Allow margins for marking.
Credit will be given for all questions attempted.
Each question carries equal weight.

DO NOT OPEN THIS PAPER UNTIL THE INVIGILATOR TELLS YOU TO

1. The classical Gram-Schmidt algorithm takes the columns of an $m \times n$ matrix $A = (a_1 a_2 \ldots a_n)$ and transforms them one by one to an orthonormal set $(q_1 q_2 \ldots q_n)$, finding an upper triangular matrix R such that

$$
\begin{aligned}
q_1 &= \frac{a_1}{r_{11}}, \\
q_2 &= \frac{a_2 - r_{12} q_1}{r_{22}}, \\
&\vdots \\
q_n &= \frac{a_n - \sum_{i=1}^{n-1} r_{in} q_i}{r_{nn}}.
\end{aligned}
$$

(a) (i) Describe how to compute r_{jj} at stage j. (2 marks)

(ii) Show that the jth step (computing q_j) of the classical Gram-Schmidt algorithm is equivalent to multiplication of a_j by a orthogonal projector P_j, followed by rescaling to obtain a unit vector. With justification, provide a formula for P_j, showing that it is indeed an orthogonal projector. (6 marks)

(b) (i) Show that P_j can be written as the product of $(j-1)$ projectors of the form $I - vv^*$ for appropriate (and different in each projector) unit vectors v. (3 marks)

(ii) Show that each of these $(j-1)$ projectors has rank $m-1$. (3 marks)

(c) Provide a rearrangement of the multiplication of these projectors that results in the modified Gram-Schmidt algorithm. Show that this rearrangement is mathematically equivalent to the projector formulation of the classical Gram-Schmidt algorithm discussed above. (6 marks)

(Total: 20 marks)

2. (a) Given a problem $f : X \to Y$, we are given

1. A floating point number system,
2. An algorithm for computing f in exact arithmetic,
3. A floating point implementation $\tilde{f}$ for f.

(i) Give the definition of stability of the algorithm $\tilde{f}$.

(4 marks)

(ii) Give the definition of backward stability of the algorithm $\tilde{f}$.

(4 marks)

(b) The backward substitution algorithm solves $Rx = y$ by computing the elements of x one by one as they become available. What does it mean for the backward substitution algorithm to be backward stable under perturbations to R? (6 marks)

(c) Given an $m \times n$ matrix A, the Householder algorithm finds $m \times m$ Q orthogonal and $m \times r$ R upper triangular such that $QR = A$. What does it mean for the Householder algorithm to be backward stable under perturbations to A? (6 marks)

(Total: 20 marks)

3. (a) Consider the pure QR algorithm applied to a real matrix A, which operates as follows:

   * Set $A^{(0)} = A$, $k = 0$.
   * While $A^{(k)}$ is not sufficiently upper triangular:
     · Find orthogonal $Q^{(k+1)}$ and upper triangular $R^{(k+1)}$ such that $A^{(k)} = Q^{(k+1)}R^{(k+1)}$.
     · Set $A^{(k+1)} = R^{(k+1)}Q^{(k+1)}$.
     · Set $k = k + 1$.

   Define orthogonal $(Q')^{(k)}$, upper triangular $(R')^{(k)}$ such that

$$(Q')^{(k)} = Q^{(k)}Q^{(k-1)} \dots Q^{(1)}, \quad (R')^{(k)} = R^{(1)}R^{(2)} \dots R^{(k-1)}R^{(k)}.$$

   (i) Show that

$$A^{(k)} = ((Q')^{(k)})^* A (Q')^{(k)}.$$

(4 marks)

   (ii) Show that

$$A^k := \underbrace{AA \dots A}_{k \text{ times}} = (Q')^{(k)}(R')^{(k)}.$$

(4 marks)

(b) Let A be a symmetric positive definite matrix, and let $A^{(k)}$ be the result of k iterations of the QR algorithm for finding eigenvalues. Show that $A^{(k)}_{mm}$ (the (m, m) entry of $A^{(k)}$) provides a Rayleigh quotient estimate of the eigenvalue based on $q^{(k)}_m$ which is the final column of $(Q')^{(k)}$.

(4 marks)

(c) Let A be a symmetric positive definite $m \times m$ matrix, and let σ be the shift parameter for the QR algorithm obtained from A using the Rayleigh quotient shift.

   Show that one iteration of Rayleigh quotient iteration applied to $x^0 = e_m = (0, 0, \dots, 0, 1)^T$ transforms to $x^1 = q_m$, where q_m is the final column of Q where $A - \sigma I = QR$, with unitary Q and upper triangular R.

(8 marks)

(Total: 20 marks)

4. (a) By appealing to the polynomial formulation of GMRES, find the maximum number of iterations required for GMRES to solve $Ax = b$ when $A = I + uv^T$ where I is the $m \times m$ identity matrix, and $u, v \in \mathbb{R}^m$ are given vectors with $u^T v \neq 0$. (You may assume that A is diagonalisable.) (6 marks)

(b) Consider the $m \times m$ matrix given by

$$A = \begin{pmatrix} 2 & -1 & \cdots & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & \vdots \\ \vdots & & & & \vdots \\ \vdots & 0 & -1 & 2 & -1 \\ 0 & 0 & \cdots & -1 & 2 \end{pmatrix}.$$

For $1 \leq \omega < 2$, the successive over relaxation (SOR) iteration for solving $Ax = b$ is given by

$$(D/\omega + L)x^{k+1} = -Ux^k + b,$$

where D, L and U are the diagonal, strictly lower triangular and strictly upper triangular parts of A, respectively. For this choice of A, provide an $\mathcal{O}(m)$ algorithm for implementing one iteration of SOR. Briefly justify why it is $\mathcal{O}(m)$.

(6 marks)

(c) Find a formula for the spectral radius of the iteration matrix C for SOR applied to this matrix. (8 marks)

(Total: 20 marks)

5. Define $\mathcal{P}^n$ as the set of real polynomials of degree n with coefficient 1 for the nth power, i.e. polynomials of the form

$$p(x) = c_0 + c_1 x + c_2 x^2 + \ldots + c_{n-1} x^{n-1} + x^n,$$

where the constants $c_0, c_1, \ldots, c_{n-1}$ are arbitrary real numbers.

Given $b \in \mathbb{R}^m$, and an $m \times m$ real matrix A, we consider the problem of finding $p \in \mathcal{P}^n$ such that $\|p(A)b\|$ is minimised.

(a) For any $p \in \mathcal{P}^n$, show that

$$p(A)b = A^n b - Q_n y,$$

for some $y \in \mathbb{R}^n$, where Q_n is the $m \times n$ matrix produced by n iterations of Arnoldi iteration on A and b.

(5 marks)

(b) (i) Show that the minimisation problem for p is equivalent to finding $y \in \mathbb{R}^n$ such that $\|A^n b - Q_n y\|$ is minimised.

(2 marks)

(ii) Hence, show that this is equivalent to finding $p \in \mathcal{P}^n$ such that $Q_n^* p(A) b = 0$. (3 marks)

(c) Hence, show that $Q_n^* Q p(H) Q^* b = 0$, where $QHQ^* = A$ is the factorisation of A that is iteratively obtained using Arnoldi iteration, with Q an $m \times m$ orthogonal matrix and H an $m \times m$ upper Hessenberg matrix. (5 marks)

(d) Assuming that $Q^* b = (|b|, 0, 0, \ldots, 0)^T$, show that the first column of $p^n(H_n) = 0$, where H_n is the $n \times n$ Hessenberg matrix corresponding to the top $n \times n$ block of H. (5 marks)

(Total: 20 marks)

BSc and MSci EXAMINATIONS (MATHEMATICS)

May 2024

This paper is also taken for the relevant examination for the Associateship.

M70024

Computational Linear Algebra (Solutions)

Setter's signature	Checker's signature	Editor's signature
....	....	....

1. (a) (i) When computing q_j, r_{jj} is the normalisation factor to ensure that $\|q_j\| = 1$. Thus at stage j, $\boxed{\text{seen} \Downarrow}$

$$r_{jj} = \|a_j - \sum_{i=1}^{j-1} r_{ij} q_i\|.$$

$\boxed{2, \text{A}}$

(ii) Looking at the general formula

$$r_{jj} q_j = a_j - \sum_{i=1}^{j-1} r_{ij} q_i,$$

taking inner products with a_i^*, $i = 1, \ldots, j-1$ gives

$$r_{ij} = q_i^* a_j.$$

Hence,

$$r_{jj} q_j = \underbrace{\left(I - \sum_{i=1}^{j-1} q_i q_i^* \right)}_{=P_j} a_j.$$

We check that P_j is a projector by showing that $\hat{P}_j = \sum_{i=1}^{j-1} q_i q_i^*$ is a projector. Then $P_j = I - \hat{P}_j$ is the complementary projector to $\hat{P}_J$, another projector. We have

$$\begin{aligned}
(\hat{P}_j)^2 &= \left(\sum_{i=1}^{j-1} q_i q_i^* \right)^2, \\
&= \sum_{i=1}^{j-1} \sum_{k=1}^{k-1} q_i \underbrace{(q_i^* q_k)}_{=\delta_{ik}} q_k^*, \\
&= \sum_{i=1}^{j-1} q_i q_i^* = \hat{P}_j,
\end{aligned}$$

so $\hat{P}_j$ is a projector. To check that $\hat{P}_j$ is an orthogonal projector, we need to show that $(\hat{P}_j v)^*(\hat{P}_j v - v) = 0$ for all v. We have

$$\begin{aligned}
(\hat{P}_j v)^*(\hat{P}_j v - v) &= v^* \hat{P}_j^* (\hat{P}_j v - v) \\
&= v^* \hat{P}_j (\hat{P}_j v - v) \\
&= v^* \underbrace{(\hat{P}_j v - \hat{P}_j v)}_{=0} = 0,
\end{aligned}$$

using the projector property and the fact that

$$\hat{P}_j^* = \left(\sum_{i=1}^{j-1} q_i q_i^* \right)^* = \sum_{i=1}^{j-1} (q_i q_i^*)^* = \sum_{i=1}^{j-1} q_i q_i^* = \hat{P}_j.$$

$\boxed{6, \text{A}}$

(b) (i) If $a^* b = 0$, then

$$(I - aa^*)(I - bb^*) = I - aa^* - bb^* + a \underbrace{a^* b}_{=0} b^*.$$

We apply this argument iteratively to obtain

$$(I - q_j q_j^*) \dots (I - q_2 q_2^*)(I - q_1 q_1^*) = I - \sum_{i=1}^{j} q_i q_i^*,$$

i.e.

$$P_{\perp q_j} \dots P_{\perp q_2} P_{\perp q_1} = P_j,$$

where

$$P_{\perp q_k} = I - q_k q_k^*,$$

which is a projector since

$$P^2 = (I - q_k q_k^*)(I - q_k q_k^*) = I - 2 q_k q_k^* + \underbrace{q_k q_k^* q_k q_k^*}_{q_k q_k^*} = I - q_k q_k^* = P.$$

$\boxed{3, \text{B}}$

(ii) The rank of a matrix is the dimension of its range. $P_{\perp q_i}$ is the orthogonal projector onto the orthogonal complement of q_i which has dimension $m-1$. $\boxed{3, \text{B}}$

(c) Gram-Schmidt orthogonalisation can thus be written as $\boxed{\text{seen} \Downarrow}$

$$r_{11} q_1 = a_1, \dots, r_{22} q_2 = P_{\perp q_1} a_2, r_{33} q_3 = P_{\perp q_2} P_{\perp q_1} a_3, \dots, r_{nn} q_n = P_{\perp q_2} P_{\perp q_1} a_n.$$

The modified Gram-Schmidt algorithm performs these calculations by first forming q_1 by normalisation, then applying $P_{\perp q_1}$ to columns 2 to n of A. Then, q_2 is completed by normalisation, and we apply $P_{\perp q_2}$ to columns 3 to n of A. Then q_3 is completed by normalisation and so on. This is equivalent because we obtain the same formulae but just build up the matrix products across all of the columns instead of working one column at a time. $\boxed{6, \text{A}}$

2. (a) (i) Stability of $\tilde{f}$ means that for each $x \in X$, there exists $\tilde{x}$ with seen $\Downarrow$

$$\frac{\|\tilde{f}(x) - f(\tilde{x})\|}{\|f(\tilde{x})\|} = \mathcal{O}(\epsilon), \text{ and } \frac{\|\tilde{x} - x\|}{\|x\|} = \mathcal{O}(\epsilon),$$

where ϵ is the "machine epsilon" for the floating point number system. 4, A

(ii) Backward stability of $\tilde{f}$ means that for each $x \in X$, there exists $\tilde{x}$ with

$$\tilde{f}(x) = f(\tilde{x}) = 0, \text{ and } \frac{\|\tilde{x} - x\|}{\|x\|} = \mathcal{O}(\epsilon).$$

4, A

(b) In this case the input to f is R, and the output of f is x, the solution to $Rx = b$. Then, backwards stability means that there exists $\tilde{R}$ such that $\tilde{R}\tilde{x} = y$, where $\tilde{x}$ is result of the backward substitution algorithm using the floating point number system, and where $\|\tilde{R} - R\| = \mathcal{O}(\epsilon)\|\tilde{R}\|$. 6, B

(c) In this case the input to f is A, and the output of f is Q and R, where $A = QR$. Backwards stability means that there exists $\tilde{A}$ with $\|A - \tilde{A}\| = \mathcal{O}(\epsilon)\|\tilde{A}\|$, and $\tilde{Q}\tilde{R} = \tilde{A}$, where $\tilde{Q}$ and $\tilde{R}$ are the result of the Householder algorithm applied to A using the floating point number system. 6, B

3. (a) (i) We prove by induction. For $k = 1$, we have seen ⇓

$$A^{(1)} = R^{(1)}Q^{(1)} = (Q^{(1)})^* A Q^{(1)} = ((Q')^{(1)})^* A (Q')^{(1)}.$$

Assuming it is true for $k > 0$, then

$$\begin{aligned} A^{(k+1)} &= R^{(k+1)}Q^{(k+1)} = (Q^{(k+1)})^* A^{(k)} Q^{(k+1)} \\ &= (Q^{(k+1)})^* ((Q')^{(k)})^* A (Q')^{(k)} Q^{(k+1)} = ((Q')^{(k+1)})^* A (Q')^{(k+1)}, \end{aligned}$$

4, B

(ii) We prove by induction. For $k = 1$, we have $A = Q^{(1)}R^{(1)} = (Q')^{(1)}(R')^{(1)}$, so the hypothesis is clear in that case.
Assuming it holds for a particular value of k, then,

$$\begin{aligned} A^{k+1} &= AA^k = A(Q')^{(k)}(R')^{(k)} \\ &= (Q')^{(k)}A^{(k)}(R')^{(k)} = (Q')^{(k)}Q^{(k+1)}R^{(k+1)}(R')^{(k)} \\ &= (Q')^{(k+1)}(R')^{(k+1)}, \end{aligned}$$

as required. 4, A

(b) Writing $e_m = (0, 0, \ldots, 0, 1)^T$, and $q_m^{(k)}$ as the last column of $(Q')^{(k)}$, seen ⇓

$$\begin{aligned} A_{mm}^{(k)} &= e_m^T A^{(k)} e_m = e_m^T (Q'^{(k)})^T A Q'^{(k)} e_m \\ &= (q_m^{(k)})^T A q_m^{(k)} = \frac{(q_m^{(k)})^T A q_m^{(k)}}{(q_m^{(k)})^T q_m^{(k)}}, \end{aligned}$$

since $(q_m^{(k)})^T q_m^{(k)} = 1$. This is precisely the Rayleigh quotient using $q_m^{(k)}$. 4, A

(c) The Rayleigh quotient shift is $\sigma = e_m^* A e_m = A_{mm}$. If $A - \sigma I = QR$, then $(A - \sigma I) = (A - \sigma I)^T = R^T Q^T$, and $(A - \sigma I)Q = R^T$. unseen ⇓
Then, $(A - \sigma I)q_n = (A - \sigma I)Qe_n = R^T e_n = r_{nn} e_n$.
Rayleigh quotient iteration solves

$$(A - \sigma I)y^{n+1} = x^n,\ x^{n+1} = \frac{y^{n+1}}{\|y^{n+1}\|},$$

and the above shows that if $x^0 = e_n$, then $x^1 = q_n$. 8, D

4. (a) We know from notes that the kth iteration of GMRES minimises $\|p(A)b\|$ over all degree k polynomials p satisfying $p(0) = 1$. If A is diagonalisable and has r distinct eigenvalues, then expanding b in the eigenbasis from A shows that the residual is guaranteed to vanish if a polynomial can be found that vanishes at those r eigenvalues, which is possible if it has degree $r + 1$. $\boxed{\text{unseen} \Downarrow}$

If $v^T x = 0$, then $Ax = x$, so there is an eigenspace of dimension $m - 1$ with eigenvalue 1. And, $Au = (1 + v^T u)u$, so u is an eigenvector with eigenvalue $1 + v^T u$. Hence, A has two distinct eigenvalues, and so GMRES can find the exact solution in 3 iterations or less. $\boxed{6, \mathrm{C}}$

(b) To perform one iteration of SOR, we apply the following steps: $\boxed{\text{unseen} \Downarrow}$

1. Multiply x^k by U,
2. Subtract the result from b to get r.
3. Apply forward substitution to obtain x^{k+1} from $(I\omega + L)x^{k+1} = r$.

Here $U_{ij} = 0$ unless $i = j - 1$, and $(I\omega + L)_{ij} = 0$ unless $i = j$ or $i = j + 1$.

In the first stage, each entry is the linear combination of at most two numbers because we only need to consider the nonzero entries of U, of which there are at most 2 in each row. This is $\mathcal{O}(m)$.

The second stage is m pairwise subtractions, $\mathcal{O}(m)$.

In the third stage, for the jth entry (starting with the first and working forwards), we divide r_j by U_{jj}, and, if $j > 1$, subtract x_{j-1}^{k+1}, to get x_j^{k+1}. This is one division and one subtraction for each j from 1 to m, $\mathcal{O}(m)$. Hence the whole algorithm is $\mathcal{O}(m)$ since it is the concatenation of three $\mathcal{O}(m)$ operations. $\boxed{6, \mathrm{C}}$

(c) We need to compute eigenvectors u and eigenvalues λ of the iteration matrix $C = -(D/\omega + L)^{-1}U$, satisfying

$$Uu + \lambda(D/\omega + L)u = 0.$$

If we write $u_0 = 0$ and $u_{m+1} = 0$, then this can be written as

$$-u_{j+1} + \lambda(2u_j/\omega - u_{j-1}) = 0.$$

Then we assume eigenvectors u^l with $u_j^l = \sin(jl\pi/(m + 1))$ (to satisfy $u_0 = u_{m+1} = 0$) for $l = 1, 2, \ldots, n$, and we get

$$\Im\left(e^{ijl\pi/(m+1)}\left(-e^{il\pi/(m+1)} + \lambda_l\left(\frac{2}{\omega} - e^{-il\pi/(m+1)}\right)\right)\right) = 0.$$

This vanishes for all j only if

$$-e^{il\pi/(m+1)} + \lambda_l\left(\frac{2}{\omega} - e^{-il\pi/(m+1)}\right) = 0,$$

i.e. if

$$\lambda_l = \frac{e^{il\pi/(m+1)}}{\frac{2}{\omega} - e^{-il\pi/(m+1)}},$$

which has magnitude

$$|\lambda_l| = \frac{1}{\frac{4}{\omega^2} - \frac{2}{\omega}\left(e^{+il\pi/(m+1)} + e^{-il\pi/(m+1)}\right) + 1}$$
$$= \frac{1}{\frac{4}{\omega^2} - \frac{4}{\omega}\sin(l\pi/(m+1)) + 1}.$$

This is maximised when $\sin(l\pi/(m+1))$ is largest i.e. when $l = n$, so the spectral radius is

$$\rho(C) = \frac{1}{\frac{4}{\omega^2} - \frac{4}{\omega}\sin(m\pi/(m+1)) + 1}.$$

$\boxed{8, \mathrm{D}}$

5. (a) $\boxed{\text{unseen} \Downarrow}$

$$p(A)b = A^n b + \sum_{i=0}^{n-1} c_0 A^i b.$$

The columns of Q_n span the Krylov subspace $\{b, Ab, \ldots, A^{n-1}b\}$ so we can change basis to the columns of Q_n, obtaining

$$p(A)b = A^n b - \sum_{i=1}^{n} q_i y_i = A^n b - Q_n y,$$

where $Q_n = (q_1\, q_2\, \ldots\, q_n)$ and $y = (y_1, y_2, \ldots, y_n)^T$. $\boxed{5, \text{M}}$

(b) (i) Since y and c are related by change of basis (and sign), y is arbitrary and we can equivalently seek to minimise $\|A^n b - Q_n y\|$ over all $y \in \mathbb{R}^n$. $\boxed{2, \text{M}}$

(ii) This minimisation is attained by finding y such that the orthogonal projection of $A^n b$ onto the Krylov subspace spanned by the columns of Q_n. The result of the projection is $Q_n y$. Hence $A^n b - Q_n = p(A)b$ needs to be orthogonal to the Krylov subspace, i.e. $Q_n^* p(A)b = 0$. $\boxed{3, \text{M}}$

(c) If $A = QHQ^*$, then $A^k = (QHQ^*)^k = QHQ^*QHQ^* \ldots QHQ^* = QH^kQ^*$. Then $p(A) = \sum_{i=0}^{n} c_i A^i = \sum_{i=0}^{n} c_i QH^iQ^* = Q\sum_{i=0}^{n} c_i H^n Q^* = Qp(H)Q^*$. Hence,

$$0 = Q_n^* p(A)b = Q_n^* Q p(H) Q^* b.$$

$\boxed{5, \text{M}}$

(d) If $Q^* b = (|b|, 0, \ldots, 0)^T$, then

$$0 = Q_n^* Q p(H)(|b|, 0, \ldots, 0)^T,$$

which says that the first column of $Q_n^* Q p(H)$ is zero. $Q_n^* Q$ is the $m \times n$ matrix B with $B_{ij} = 1$ if $i = j$ and 0 otherwise, so $Q_n^* Q p(H)$ is the first n rows of the first n columns of $p(H)$. Only the first n rows of H are nonzero, so only the first n rows of H^k are zero for $k \leq n$. Hence, $Q_n^* Q p(H)$ is equal to the first n rows of the first n columns of $p(H_n)$, where H_n is the first n rows and columns of H.

Hence, the first column of $p(H_n)$ is zero. $\boxed{5, \text{M}}$

Review of mark distribution:

Total A marks: 30 of 32 marks
Total B marks: 22 of 20 marks
Total C marks: 12 of 12 marks
Total D marks: 16 of 16 marks
Total marks: 100 of 80 marks
Total Mastery marks: 20 of 20 marks

MATH60024 Computational Linear Algebra

Question	Marker's comment
1	Students generally made a good effort of this question.Where students lost marks, it is because theyi) did not derive the formula for the projector using the formulae provided in the question, (were asked to "show")ii) quoted the form of the orthogonal projector rather than "showing" "with justification" as requested in the question.
2	Students generally performed very well in this question. The main loss of marks occurred in parts b and c when candidates got confused between inputs and outputs in the definition.
3	Students performed well on part a but several students struggled to find the right formulation in parts b and can error on the exam paper was clarified during the exam, on the order of the products for Q' and R' - the Q products should have been ascending from left to right and vice versa for the R products. This is bookwork material for this course and it did not appear to confuse the students, and was relevant only to part a.
4	Part a was challenging to several students, who didn't recognise that it was possible to compute the (two distinct) eigenvalues of the matrix.In Part b, students mostly made a good job, students who lost marks did so because they did not explain how to exploit the zeros in the matrices involved.In Part c, unfortunately there was an error in the model solution and this problem is intractable to the methods in the course (it requires additional theorems to compute the spectral radius) - this was only detected during marking. This has been reported to the exam board to take account of in scaling if necessary. There was also an error in the formula for the SOR method, which should have been written as(D/omega + L)x^{k+1} = (D(1-omega)/omega - U)x^k + b.(The D term on the RHS was missing.) This was detected and corrected during the exam with an announcement. Credit was given when students had worked with the incorrect formula.

MATH70024 Computational Linear Algebra

Question	Marker's comment
1	See comments on 60024
2	See comments for 60074
3	see comments on 60024
4	See comments for 60024
5	Most students gave good answers to a and b(i), with some losing marks in b(ii) because of not noting that p(A)b is in the column space of A at the minimum. Some students struggled in the later sections to link Q_n with the full Schur factorisation of A.