Exercise 1 Data fitting

Consider fitting the data

\{(0,1), (1,2), (2,-1), (3,4), (4,12), (5,10), (6,20), (7,40), (8,30), (9,60), (10,99)\},

with a quadratic polynomial. To do this we assume \( p(x) = ax^2 + bx + c \) and

setup the system

\[(0, 1) \rightarrow p(0) = 0 + 0 + c = 1,\]
\[(1, 2) \rightarrow p(1) = a + b + c = 2,\]
\[\vdots\]
\[(10, 99) \rightarrow p(10) = 100a + 10b + c = 99.\]

This system is over determined as we have 11 equations and just 3 unknowns.

Write this system in the form

\[Ac = y.\]

Use MATLAB to solve the system in the least-squares sense to find the ‘best’ \( a, b \) and \( c \). Plot the original data and your fit on the same graph. Upload your code and turn in your plot.

Exercise 2

Use the algorithm set out on the top of page 234 to code the algorithm for Gram-Schmidt. Note that when we did this in class we took the coding convention and ignored the superscripts. This is how you should code it. You start with vectors \( \{w_1, \ldots, w_n\} \) and modify them so that they become orthogonal.
\( w_1 = w_1 / \|w_1\| \% \text{Start} \)

for \( j = 2, \ldots, n \)

for \( k = j, \ldots, n \)

\( w_k = w_k - \langle w_k, w_{j-1} \rangle w_{j-1} \)

end

\( w_j = w_j / \|w_j\| \)

end

It is probably easiest to start with all \( w_j \)'s as columns of a matrix \( W \) so that you can refer to \( w_1 \) with \( W(:,1) \), \( w_2 \) with \( W(:,2) \), etc.

(a) Use your code to perform Gram-Schmidt on the vectors

\[
\begin{align*}
\mathbf{w}_1 &= \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}, \\
\mathbf{w}_2 &= \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \\
\mathbf{w}_3 &= \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \\
\mathbf{w}_4 &= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \\
\mathbf{w}_5 &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}.
\end{align*}
\]

Upload your code on the moodle page and turn in the last vector, \( \mathbf{w}_5 \), after this process.

(b) Consider adding another vector

\[
\mathbf{w}_6 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.
\]

How does the fact that we have a linearly dependent set of vectors manifest itself in the algorithm? Hint: Print out \( w_j \) before dividing by its norm.

The following exercises should be done by hand, showing all steps.

**Exercise 3**
Olver & Shakiban— 5.1.1b,d,f

**Exercise 4**
Olver & Shakiban— 5.1.5

**Exercise 5**
Olver & Shakiban— 5.2.9c
Exercise 6
Olver & Shakiban— 5.2.10

Exercise 7  EXTRA CREDIT — 5 points
Olver & Shakiban— 5.3.18 — for the second question restrict to 2 × 2 matrices
— an extra 2 pts without this restriction