Assignment #4

Due Monday, 7 October 2024, at the start of class

From the textbook[1] please read sections 3.1, 3.2, 3.3, 4.1, 4.2, 4.3, 4.4.

DO THE FOLLOWING EXERCISE from section 3.2, page 58:
- Exercise 2.2

DO THE FOLLOWING EXERCISES from section 3.3, pages 91–93:
- Exercise 3.1
- Exercise 3.3
- Exercise 3.4

DO THE FOLLOWING EXERCISES from section 4.1, pages 98–100:
- Exercise 1.1 (i)–(iv)
- Exercise 1.2

Problem P8.   *You should be able to do Exercise 3.1 above by hand. In that Exercise, your answer to each part should literally be a* basis *for the null space, that is, you should provide a set of column vectors. For example, your part (ii) solution should be a set of three vectors from* $\mathbb{R}^4$*.*

Please use MATLAB (or similar) to check that your answers to Exercise 3.1 are correct. Specifically, for each part,
- enter the matrix A into MATLAB (or similar),   `>> A = [...`
- form the basis vectors into a matrix Z, and enter that
  matrix into MATLAB (or similar),   `>> Z = [...`
- confirm that AZ is zero to within rounding error, and   `>> A * Z`
- compute the rank of Z.   `>> rank(Z)`

For the last item, this should confirm that your Z matrix has full column rank. *Minimize the paper used to show what you did.*

Problem P9.   *This problem is built from the key paragraph at the top of page 87.*

In MATLAB, or a programming language of your choice, write a short function

```
function Z = nullmat(A,ind)
```

---

[1]Griva, Nash, and Sofer, *Linear and Nonlinear Optimization*, 2nd ed., SIAM Press 2009.

which extracts B and N from A and then returns the null matrix

$$Z = \begin{pmatrix} -B^{-1}N \\ I \end{pmatrix}.$$

If the user chooses the appropriate indices then Z will be a basis matrix for the null space of A.

In more detail, your code should take the matrix A as the first argument, and a list/array of column indices as the second (`ind`) argument. It should determine the size of A using `size()` or similar. From the `ind` list, which a good code should check for reasonableness, you will identify/construct the B and N parts of the matrix A. That is, the indices identify B. Your code could check that B is the right kind of object. Then your code builds and returns Z.

Without input checking your code could be one busy line of MATLAB.[2] But please do the input checking too! After writing the code, run it on at least two parts of Exercise 3.1 above, showing that it succeeds in generating a good Z.

Problem P10.   (*I wrote a ratio test* MATLAB *code for the solutions to Assignment # 3, and it is posted at the Codes tab of the public site. You may use it to do this problem, or write your own. In any case, be careful with the form of the inequality constraints! They need to be in the form "$Ax \geq b$" to use the ratio test as stated on page 81, and to run my code.*)

(a)   For part (ii) of Exercise 4.1 above, you should already have a sketch. Confirm that $x = (6,0)^\top$ is feasible. (*Or compute $Ax - b$ in e.g.* MATLAB, *and check!*) Confirm that $p = (0,1)^\top$ and $p = (1,0)^\top$ are feasible directions at that point. (*Graphically.*) Compute the $\bar{\alpha}$ from the ratio test for these directions. (*Use the code!*) Add the vectors $\bar{\alpha}p$ to the graph, based at x. (*Make a duplicate graph if you want.*)

(b)   Same instructions, but for part (iii) of Exercise 4.1, now using $x = (8,0)^\top$, $p = (-2,1)^\top$, and $p = (-\pi,0)^\top$.

---

[2]Hint. `setdiff()` is your friend.