Column space, Left Null space, control applications

Computational Intelligence, Lecture 3

by Sergei Savin

Spring 2021

Content

- Column space
- Column space basis
- Column space and null space
- Projector onto column space
- Left null space
- Finding fixed points
- Checking fixed points
- Correcting fixed points
- Finding fixed points for affine systems
- Minimize one of the control inputs

Column space

Consider the following task: find all vectors $\mathbf{y}$ such that $\mathbf{y} = \mathbf{A}\mathbf{x}$.

It can be re-formulated as follows: find all elements of the *column space* of $\mathbf{A}$.

Definition 1

Column space of $\mathbf{A}$ is the set of all outputs of the matrix $\mathbf{A}$, for all possible inputs

We will denote column space as $\mathcal{C}(\mathbf{A})$. In the literature, it is often called an *image* of $\mathbf{A}$.

Column space basis

The problem of finding orthonormal basis in the column space of a matrix is often called *orthonormalization* of that matrix. Hence in MATLAB and Python/Scipy the function that does it is called `orth`:

- `C = orth(A).`
- `C = scipy.linalg.orth(A).`

That is how one finds all the outputs of the matrix $\mathbf{A}$: as $\{\mathbf{Cz} : \forall \mathbf{z}\}$.
Notice that $\{\mathbf{Ax} : \forall \mathbf{x}\}$ might contain repeated entries if $\mathbf{A}$ has a non-trivial null space.

Let $\mathbf{A}$ be a square matrix, a map from $\mathbb{X} = \mathbb{R}^n$ to $\mathbb{Y} = \mathbb{R}^n$. Notice that if it has a non-trivial null space, it follows that multiple unique inputs are being mapped by it to the same output:

$$\begin{aligned} \mathbf{y} = \mathbf{A}\mathbf{x}_r = \mathbf{A}(\mathbf{x}_r + \mathbf{x}_n), \\ \mathbf{x}_r \in \mathcal{R}(\mathbf{A}) \\ \forall \mathbf{x}_n \in \mathcal{N}(\mathbf{A}) \end{aligned} \tag{1}$$

In fact, if null space of $\mathbf{A}$ has k dimensions, it implies that an n-dimentional subspace of $\mathbb{X}$ is mapped to a single element of $\mathbb{Y}$.

It follows that in this case the dimensionality of the column space could not exceed $n - k$.

Projector onto column space

Given vector $\mathbf{y}$ and matrix $\mathbf{A}$, let us find projector of $\mathbf{y}$ onto the column space of $\mathbf{A}$.

This is done in the same manner as we did with the null space:

$$\mathbf{y}_c = \mathbf{A}\mathbf{A}^{+}\mathbf{y} \in \mathcal{C}(\mathbf{A}) \tag{2}$$

This feels nice, as it only requires the matrix itself, no need for the orthonormal basis as before. However, you need to remember that the pseudoinverse is base on SVD decomposition, same as operations of finding a basis in the null space or column space.

Projector onto row space

In the same we can define a projector onto row space. Given vector $\mathbf{x}$ and matrix $\mathbf{A}$, let us find projector of $\mathbf{x}$ onto the row space of $\mathbf{A}$:

$$\mathbf{x}_r = \mathbf{A}^+\mathbf{A}\mathbf{x} \in \mathcal{R}(\mathbf{A}) \tag{3}$$

You can think of this in the following terms: first we find what output $\mathbf{x}$ makes, then we find the smallest norm vector that produces this same output, and this vector has to 1) have the same row space projector (because output is the same), 2) has to lie in the row space, hence it is the row space projector of $\mathbf{x}$.

Left null space

The subspace, orthogonal to the column space is called *left null space*.

Definition

Space of all vectors that can't be produced as outputs of matrix $\mathbf{A}$ is called *left null space*. Zero vector is included, as in linear spaces.

If we want to project vector $\mathbf{y}$ onto the left null space of $\mathbf{A}$, we do it as:

$$\mathbf{y}_l = (\mathbf{I} - \mathbf{A}\mathbf{A}^+)\mathbf{y} \in \mathcal{C}^\perp(\mathbf{A}) \tag{4}$$

Finding fixed points

Given LTI system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, 1) find if there are states that can be made into fixed points, 2) find all states that can be made into fixed points with a constant control law.

Solution:

1. Yes, state $\mathbf{x} = \mathbf{0}$ becomes a fixed point under control law $\mathbf{u} = \mathbf{0}$ or $\mathbf{u} = -\mathbf{K}\mathbf{x}$.
2. Let us find null space of the matrix $\begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix}$ as $\mathbf{N} = \text{null}(\begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix})$. We can find all $\mathbf{x}$, $\mathbf{u}$ pairs that produce fixed points as follows: $\begin{bmatrix}\mathbf{x} \\ \mathbf{u}\end{bmatrix} = \mathbf{N}\mathbf{z}$, $\forall \mathbf{z}$. Let $\mathbf{N}_x$ be the first n rows of $\mathbf{N}$. Then all states that can be made into fixed points are given as $\mathbf{x}^* = \mathbf{N}_x\mathbf{z}_x$, $\forall \mathbf{z}_x$

Checking fixed points

Given LTI system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, 1) check if $\mathbf{x}^*$ can be made into a fixed point, 2) find control constant $\mathbf{u}^*$ that does it, given control law $\mathbf{u} = -\mathbf{K}\mathbf{x} + \mathbf{u}^*$.

Solution:

1. We can check that $(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^* + \mathbf{B}\mathbf{u}^* = \mathbf{0}$ has a solution, in other words that $-(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^* \in \text{col}(\mathbf{B})$. Resulting condition is given via projection into the left null space of $\mathbf{B}$: $(\mathbf{I} - \mathbf{B}\mathbf{B}^+)(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^* = \mathbf{0}$
2. This means finding such $\mathbf{u}^*$ that $(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^* + \mathbf{B}\mathbf{u}^* = \mathbf{0}$. This is done via pseudo-inverse, which provides exact solution, as long as it exists: $\mathbf{u}^* = -\mathbf{B}^+(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^*$.

Given LTI system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, and a state $\mathbf{x}^d$ which can not be made into a fixed point under constant control law, find the closest to it state $\mathbf{x}^f$ which can be made into a fixed point.

As we know from the first example, for this LTI system all fixed points under constant control are given as $\mathbf{x}^* = \mathbf{N}_x \mathbf{z}_x$. To find the closest point to a given vector in a subspace, you project the vector into that subspace. In this case, we project $\mathbf{x}^d$ to the column space (span) of $\mathbf{N}_x$:

$$\mathbf{x}^f = \mathbf{N}_x \mathbf{N}_x^+ \mathbf{x}^d \tag{5}$$

Finding fixed points for affine systems

Given LTI system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{c}$, where $\mathbf{x} \in \mathbb{R}^n$, and control law $\mathbf{u} = -\mathbf{K}\mathbf{x} + \mathbf{u}^*$, find all states that can be made fixed points by choosing appropriate $\mathbf{u}^*$.

We are required to find all solutions to the equation $(\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}^* + \mathbf{B}\mathbf{u}^* + \mathbf{c} = \mathbf{0}$. Let us define state-control pairs $\mathbf{v} = \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix}$.

We can easily find particular solution to this linear system: $\mathbf{v}^p = -\begin{bmatrix} (\mathbf{A} - \mathbf{B}\mathbf{K}) & \mathbf{B} \end{bmatrix}^+ \mathbf{c}$.

Finding null space basis $\mathbf{N}$ for the matrix of this linear system: $\mathbf{N} = \text{null}(\begin{bmatrix} (\mathbf{A} - \mathbf{B}\mathbf{K}) & \mathbf{B} \end{bmatrix})$ we get the general solution as follows: $\mathbf{v}^* = \mathbf{v}^p + \mathbf{N}\mathbf{z}$. First n equations in the expression defining $\mathbf{v}^*$ give us $\mathbf{x}^*$, the rest - $\mathbf{u}^*$.

Minimize one of the control inputs

Now that we have such powerful tools, we can solve difficult problems easily. Consider this one. Linear time-invariant (LTI) dynamical system is described as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_2\mathbf{u}_2 \tag{6}$$

Find such control inputs $\mathbf{u}_1^*$, $\mathbf{u}_2^*$ that state $\mathbf{x}^*$ becomes a fixed point. Additionally, assume that $\mathbf{u}_1^*$ is free to use, while $\mathbf{u}_2^*$ should be used as sparingly as possible.

This can be formulated in the language of optimization as follows:

$$\begin{aligned} &\underset{\mathbf{u}_1,\mathbf{u}_2}{\text{minimize}} && ||\mathbf{u}_2||, \\ &\text{subject to} && \mathbf{A}\mathbf{x}^* + \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_2\mathbf{u}_2 = \mathbf{0} \end{aligned} \tag{7}$$

Minimize one of the control inputs

part 2

In order to check that the problem has at least one solution we need to make sure that there are such inputs $\mathbf{u}_1^*$, $\mathbf{u}_2^*$ that state $\mathbf{x}^*$ becomes a fixed point. In other words, vector $\mathbf{A}\mathbf{x}^*$ should lie in the span of the columns of the matrix $[\mathbf{B}_1\ \mathbf{B}_2]$. Which is the same as saying that its projection on the compliment on this column space (left null space of $[\mathbf{B}_1\ \mathbf{B}_2]$) is zero:

$$(\mathbf{I} - [\mathbf{B}_1\ \mathbf{B}_2][\mathbf{B}_1\ \mathbf{B}_2]^+)\mathbf{A}\mathbf{x}^* = \mathbf{0} \tag{8}$$

All solutions to the problem $\mathbf{A}\mathbf{x}^* + \mathbf{B}_1\mathbf{u}_1 + \mathbf{B}_2\mathbf{u}_2 = \mathbf{0}$ can be found as:

$$\mathbf{u} = -[\mathbf{B}_1\ \mathbf{B}_2]^+\mathbf{A}\mathbf{x}^* + \text{null}([\mathbf{B}_1\ \mathbf{B}_2])\mathbf{z},\ \forall\mathbf{z} \tag{9}$$

Minimize one of the control inputs

part 3

Now we need to pick one solution out of all of them, based on the criteria that it minimizes $\mathbf{u}_2$. We can solve it as an optimization (by thinking about the derivatives of the objective/cost function), but it can also be solved as a projection.

Let us define projector $\mathbf{P} = \mathbf{B}_1\mathbf{B}_1^+$. We can prove that $\mathbf{PB}_2\mathbf{u}_2^* = \mathbf{0}$. Assume $\mathbf{PB}_2\mathbf{u}_2^* = \mathbf{a}$, then $\mathbf{u}_2^* = \mathbf{u}_2^0 + \mathbf{u}_2^a$, where $\mathbf{u}_2^0$ is in the null space of $\mathbf{PB}_2$ and $\mathbf{u}_2^a$ is in the row space of $\mathbf{PB}_2$, or equivalently $\mathbf{PB}_2\mathbf{u}_2^0 = \mathbf{0}$,$\mathbf{PB}_2\mathbf{u}_2^a = \mathbf{a}$, $(\mathbf{I} - \mathbf{P})\mathbf{B}_2\mathbf{u}_2^a = \mathbf{0}$. This gives us solution:

$$\mathbf{Ax}^* + \mathbf{B}_1\mathbf{u}_1^* + \mathbf{B}_2\mathbf{u}_2^0 + \mathbf{B}_2\mathbf{u}_2^a = \mathbf{0} \tag{10}$$

But since $\mathbf{a} \in \text{span}(\mathbf{B}_1)$ we can also provide an alternative solution:

$$\mathbf{A}\mathbf{x}^* + \mathbf{B}_1\mathbf{u}_1^a + \mathbf{B}_2\mathbf{u}_2^0 = \mathbf{0} \tag{11}$$

where $\mathbf{u}_1^a = \mathbf{u}_1^* + \mathbf{B}_1^+\mathbf{a}$.

Since $\mathbf{u}_2^0$ and $\mathbf{u}_2^a$ and belong to orthogonal subspaces, $||\mathbf{u}_2^0 + \mathbf{u}_2^a|| \geq ||\mathbf{u}_2^0||$. Therefore the alternative solution provides smaller norm $\mathbf{u}_2$, hence $\mathbf{a} = \mathbf{0}$ and $\mathbf{u}_2^a = \mathbf{0}$ and $\mathbf{P}\mathbf{B}_2\mathbf{u}_2^* = \mathbf{0}$.

Since $\mathbf{P}\mathbf{B}_2\mathbf{u}_2^* = \mathbf{0}$, $(\mathbf{I} - \mathbf{P})\mathbf{B}_2\mathbf{u}_2^* = \mathbf{B}_2\mathbf{u}_2^*$, and while $(\mathbf{I} - \mathbf{P})\mathbf{B}_1 = \mathbf{0}$.

Multiplying our constraint by $(\mathbf{I}-\mathbf{P})$ we attain:

$$(\mathbf{I}-\mathbf{P})\mathbf{A}\mathbf{x}^* + (\mathbf{I}-\mathbf{P})\mathbf{B}_2\mathbf{u}_2^* = \mathbf{0} \tag{12}$$

Which has an exact solution:

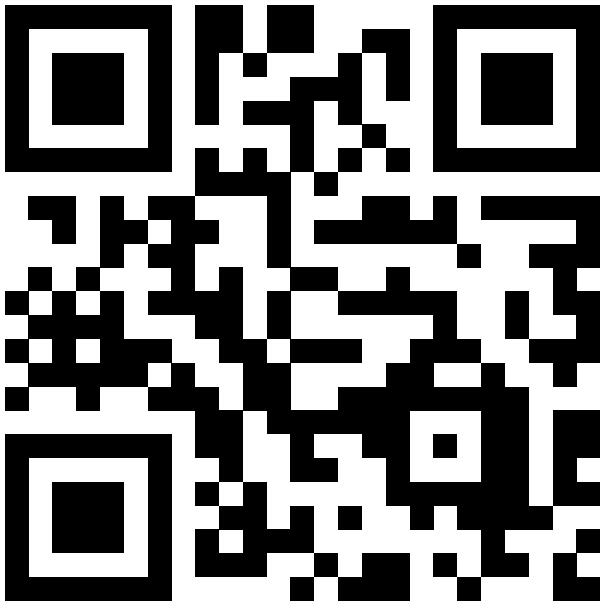
$$\mathbf{u}_2^* = -((\mathbf{I}-\mathbf{P})\mathbf{B}_2)^+(\mathbf{I}-\mathbf{P})\mathbf{A}\mathbf{x}^* \tag{13}$$

Which is the solution to the original problem; $\mathbf{u}_1^*$ can be obtained as follows:

$$\mathbf{u}_1^* = -\mathbf{B}_1^+(\mathbf{A}\mathbf{x}^* + \mathbf{B}_2\mathbf{u}_2^*) \tag{14}$$

Lecture slides are available via Moodle.

You can help improve these slides at:
github.com/SergeiSa/Computational-Intelligence-Slides-Spring-2021



Check Moodle for additional links, videos, textbook suggestions.