10/12/23

Lecture-7

07/04/21

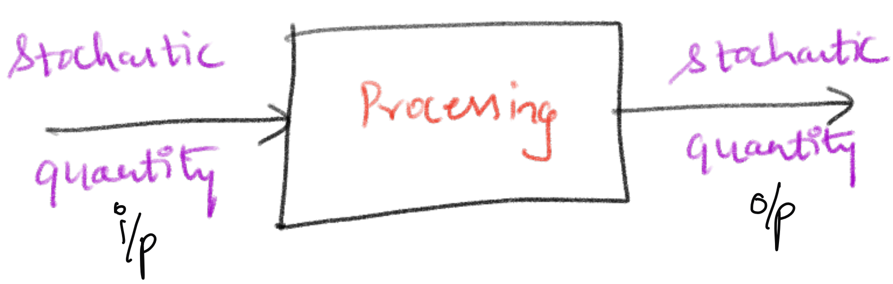
Processing:

In the last class we have studied the requirements on the Stochastic process

i.e
- Stationarity
- Ergodicity
- Mixingness

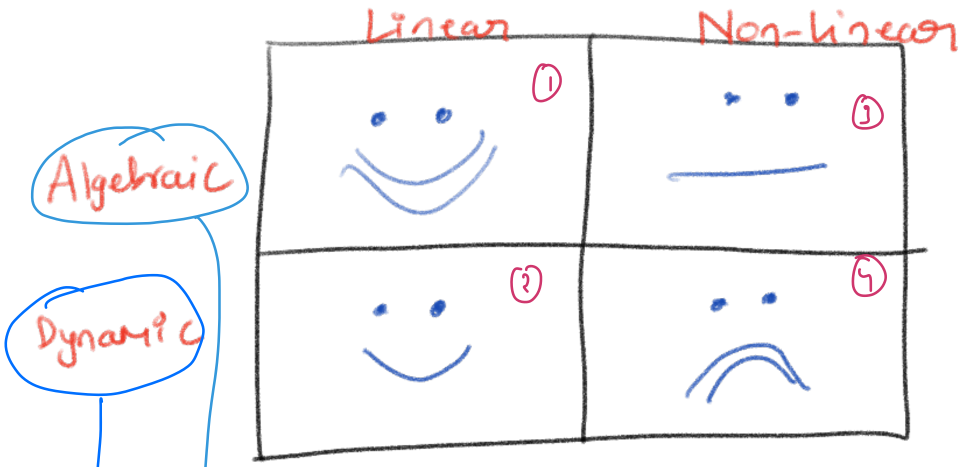
Since we have a lot of Mathematicle Model for our Stochastic quantities we want to process them.

→



Everything depends on what we put in the Box and what left and Right arrow carry.

→



present output depends on the Instantaneous value of the Input ex: Amplifier

what is the output depends on the History (Memory)

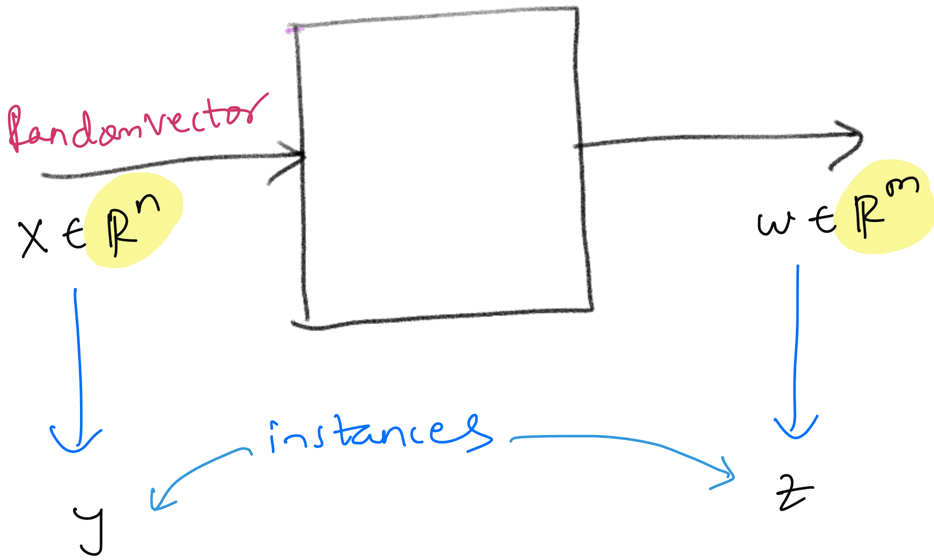
ex: Filter

Note: No s/s is completely Algebraic of course but we can model ideal s/s that way

→ Therefore, most of the time, we have to take into account History of input to determine the output.

→ In our course we will address 1, 2, 3 but not 4 even though they are very important, because of time constraints.

(I) Algebraic (+Linear) Processing of Stochastic quantities



i.e $\quad f_X(y) \rightsquigarrow f_W(z)$

PDF of X $\qquad\qquad$ PDF of W

→ Since we don't have memory issues because of lin + Algebraic process. we will use Random Variables instead of stochastic processes. because we don't have to worry about how signals change in Time.

18/12/23

- $X \in \mathbb{R}^n \longrightarrow W \in \mathbb{R}^m$

Random vector

dimension of i/p & o/p are different in general

- To take a single instance of $X \in \mathbb{R}^n$ we can call it 'y'

When we take a single instance of $W \in \mathbb{R}^m$ we call it 'z'

Instances are particular values, they are coupled with probabilities and the set of all possible instances coupled by their probabilities is the whole Random variable

→ It is clear that if we are dealing with classical task of Analog Electronics, then we are dealing with some sort of Sinusoid entering some sort of processing and exiting probably as Sinusoid.

What we would do to characterise the Box i.e the functionality of processing is

✱ Having a Model of input & Having a Model of output

Then Define how the Model of input gets transformed into the Model of the output.

For Sinusoids it's going to be Amplitude, Frequency & phase.

In Linear Filters we have a simple Rule, How to get from inputs to outputs

outputs:

a) Amplitude: we know that amplitude of the input gets multiplified by the absolute value of the Transfer function of the processing unit

b) phase: Phase of the output is altered by the argument of the transfer function

c) Frequency Doesn't change

* ∴ we know how to map the 3 quantities modelling input to 3 quantities modelling the output.

→ Things get complicated when we want to process the Stochastic quantities.

Ex: R.V at the input is not characterised by vectors but a Function i.e PDF (probability distribution function) which is an n-dimensional function.

The output is characterised by an n-dimen-sional function. i.e PDF of output.

→ Our task is determine how the input function will be transformed into output functions which depends on what is done inside the Box.

Prototype Problem:

It is particular problem which is general in some aspects and quite strange in another aspects. But is the building block on which we derive formulas we use in designing the Box.

Prototype problem assumes our Box as Algebraic but not Linear. So it is a Mapping. it assumes (input domain = output Domain) and given This we should give a formula for f_W!

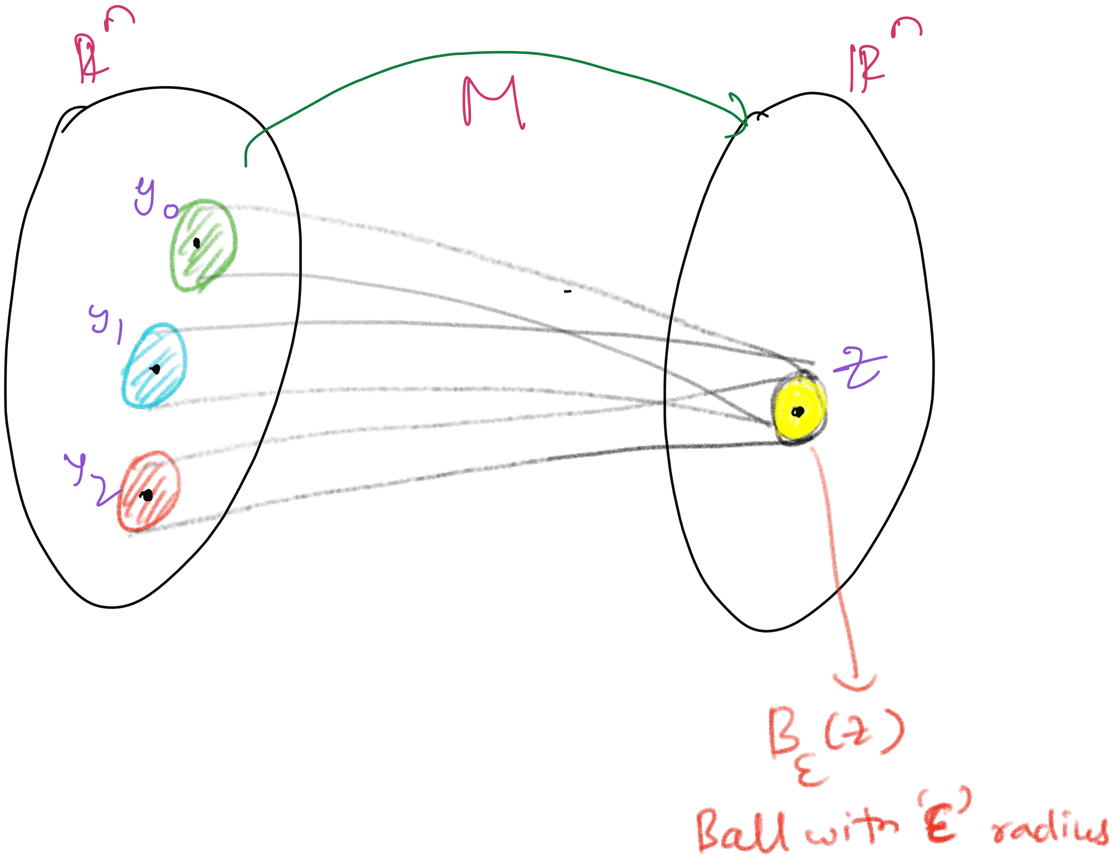
$$X \in \mathbb{R}^n \longrightarrow \boxed{M} \longrightarrow W \in \mathbb{R}^n$$

$$f_W(z) = \quad M \quad z \quad f_X$$

M: what our Box does

f_X: what is the input PDF

z: at which instance we want to calculate the output PDF



$$M(y_J) = z \quad \forall J$$

Counter Images

$$* \quad \boxed{\Pr\{w \in B_\varepsilon(z)\} \hat{\approx} f_w(z) \cdot \mu(B_\varepsilon(z))}$$

Probability of falling into Ball with Radius ϵ centered at 'z' is equal to PDF centered in 'z' multiplied by the Measure of the Ball

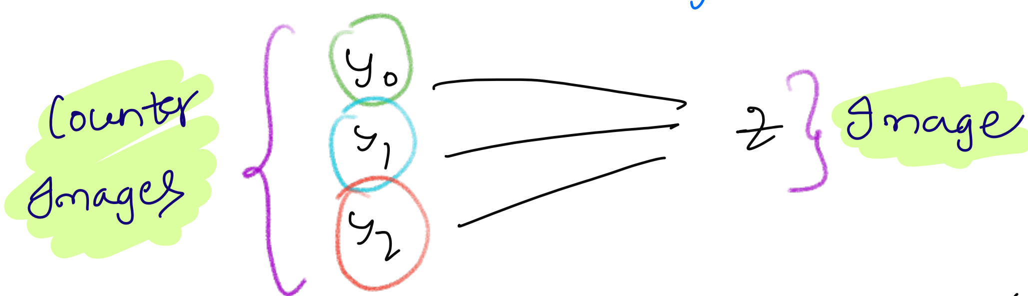
- This would be a starting point of our solution

$$\boxed{\Pr\{w \in B_\epsilon(z)\} \doteq f_w(z)\,\mu(B_\epsilon(z))}$$

$\epsilon \to 0$

Because we have $f_w(z)$ our solution in the equation.

→ In general Mappings are not invertible as we can see above. They are not Bijections.



all Three are mapping onto 'z'

∴ The Mapping is not invertible.

Here our only Assumption would be The no. of points That gets mapped onto 'z' is finite. It doesn't matter how many but they are finite.

$y_j \quad \forall j$ Counter Images

$M(y_j) = z$ Image

$\because \; y_0 \xrightarrow{M} z$

→ If we assume 'M' is sufficiently small so it's not a pathelogical function there will be a neighbourhood of y_0 that goes into the Ball around z.

The same can be assumed for y_1 & y_2

Since there is a possibility of only finite counter Images, & apart from y_0, y_1, y_2 goes into the Ball z

Therefore we can write the Equality we describe below.

$$\Pr\left\{ x \in M^{(0)^{-1}}\left(B_\epsilon(z)\right)\right\} + \Pr\left\{ x \in M^{(1)^{-1}}\left(B_\epsilon(z)\right)\right\} + \Pr\left\{ x \in M^{(2)^{-1}}\left(B_\epsilon(z)\right)\right\} \quad (1)$$

→ The probability of falling into z now is the probability of falling into either y_0, y_1, y_2 before the mapping.

$$\underbrace{f_X(y_0)}\ \underbrace{\mu\left(M^{(0)^{-1}}(B_\epsilon(z))\right)} + f_X(y_1)\,\mu\left(M^{(1)^{-1}}(B_\epsilon(z))\right) + f_X(y_2)\,\mu\left(M^{(2)^{-1}}(B_\epsilon(z))\right)$$

2

1 = 2

In General when $\epsilon \to 0$

i.e Ball should be very small.

⇒ From this we can get expression for $f_W(z)$

$$\text{i.e } f_W(z)\,\mu(B_\epsilon(z)) \cong \underbrace{\sum_{M(y_j)=z} f_X(y_j)\,\mu\left(M^{(j)^{-1}}(B_\epsilon(z))\right)}$$

Sum of each possible counter image

$$\Rightarrow \boxed{\underbrace{f_\omega(z)}_{\text{o/p PDF}} = \sum_{M(y_j)=z} \underbrace{f_x(y_j)}_{\text{i/p PDF}} \frac{\mu\left(M^{(j)^{-1}}(B_\varepsilon(z))\right)}{\mu\left(B_\varepsilon(z)\right)}}$$

This equation is in principle what we already want. With the elements in the Recepie expected to be there.

*

$$X \in \mathbb{R}^n \longrightarrow \boxed{M} \longrightarrow w \in \mathbb{R}^n$$

$$f_w(z) = \{\, M \quad z \quad f_x \,\}$$

The above expression gets simplified under the Assumption that $\varepsilon \longrightarrow 0$

→ Instead of using M our function, we can use its **Taylor Expansion** upto its first order.

The problem is that $f : \mathbb{R}^n \to \mathbb{R}^n$

not $f : \mathbb{R} \to \mathbb{R}$

Defining a Taylor expansion in Multidimensional Domain would be Tricky not Difficult.

→ The idea, is to proceed with calculations in an approximate way.

$$\mu\left(M^{(J)^{-1}}\left(B_\varepsilon(z)\right)\right) = \mu\left\{y \mid M^{(J)}(y) \in B_\varepsilon(z)\right\}$$

$$= \mu\left\{y \mid \underbrace{\left\|M^{J}(y) - z\right\|^2}_{\text{distance from the centre is less than or equal to Radius.}} \leq \varepsilon^2\right\} \quad (3)$$

Taylor expansion

↓

One-Dimensional case

$$\boxed{M^{(J)}(y) = M^{(J)}(y_J) + \nabla M^{(J)}(y_J)^T (y - y_J) + O(\|y - y_J\|^2)}$$

where

$$\nabla M^{(J)} = \left[\, \nabla M^{(J)(0)} \;\Big|\; \nabla M^{(J)(1)} \;\Big|\; \cdots \;\Big|\; \nabla M^{(J)(n-1)} \,\right]$$

This gets Transposed in the above expression

① $f : \mathbb{R} \to \mathbb{R}$ Then

$$\boxed{f(q) = f(\bar{q}) + \frac{\partial f}{\partial q}(\bar{q})(q - \bar{q}) + O(\,)}$$

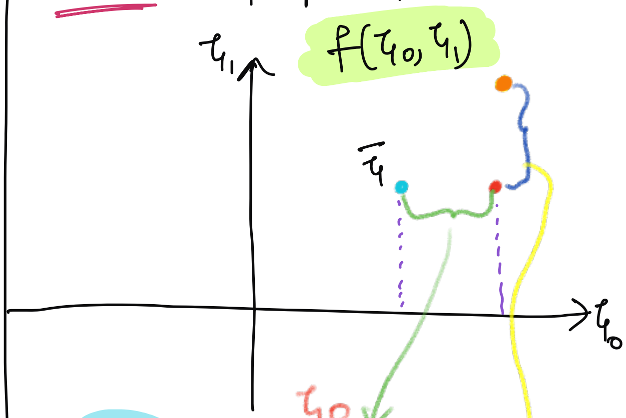
Since we are approximate only till the first order.

② $f : \mathbb{R}^n \to \mathbb{R}$

Now our function is going from $\mathbb{R}^n$ to $\mathbb{R}$

assuming all derivatives are present.

$n=2$ $\quad f : \mathbb{R}^2 \to \mathbb{R}$



$$\underbrace{f(\bar{q})}_{\text{Base point function}} + \overbrace{\frac{\partial f}{\partial q_0}(\bar{q})(q_0 - \bar{q}_0)}^{q_0} + \overbrace{\frac{\partial f}{\partial q_1}(\bar{q})(q_1 - \bar{q}_1)}^{q_1}$$

$+ o(\ )$

$$\boxed{\therefore f(\eta) = f(\bar{\eta}) + \frac{\partial f}{\partial \eta_0}(\bar{\eta})(\eta_0 - \bar{\eta}_0) + \frac{\partial f}{\partial \eta_1}(\bar{\eta})(\eta_1 - \bar{\eta}_1)}$$

This expansion could be Generalized for n-dimensions

$$f(\eta) \Rightarrow \underbrace{f(\bar{\eta}) + \begin{bmatrix} \frac{\partial f}{\partial \eta_0} \\ \frac{\partial f}{\partial \eta_1} \end{bmatrix}^T \begin{bmatrix} \eta_0 - \bar{\eta}_0 \\ \eta_1 - \bar{\eta}_1 \end{bmatrix}}_{\text{Scalar product}} + o(\ )$$

$$\boxed{f(\eta) = f(\bar{\eta}) + \nabla f(\bar{\eta})^T (\eta - \bar{\eta}) + o(\ )}$$

→ vector containing derivatives is called Gradient vector

$$\begin{bmatrix} \eta_0 - \bar{\eta}_0 \\ \eta_1 - \bar{\eta}_1 \end{bmatrix} \Rightarrow \text{vector of Deviations}$$

3) $f : \mathbb{R}^n \to \mathbb{R}^n$

$$f = \begin{bmatrix} f^{(0)} \\ f^{(1)} \\ \vdots \\ f^{(n-1)} \end{bmatrix}$$ where $f^{(J)} : \mathbb{R}^{n} \to \mathbb{R}$

We will use this in the LHS.

Using the Taylor expansion

$$\Rightarrow \underbrace{\mu\left(M^{(J)^{-1}}\left(B_\varepsilon(z)\right)\right)}_{\text{Measure of Counter Image}} = \mu\left(\left\{ y \,\middle|\, \left\| M^{(J)}(y_J) + \nabla M^{(J)}(y_J)^T (y - y_z) - z \right\|^2 \leq \varepsilon^2 \right\}\right)$$

Substituted in place of M^J its Taylor expansion

$$\Rightarrow \mu\left(M^{(J)^{-1}}\left(B_\varepsilon(z)\right)\right) = \mu\left(\left\{ y \,\middle|\, \left\| \underbrace{M^{(J)}(y_J)}_{z} + \nabla M^{(J)}(y_J)^T (y - y_z) - z \right\|^2 \leq \varepsilon^2 \right\}\right)$$

both cancel out

$$= \mu\left(\left\{ y \;\middle|\; \left\| \nabla M^{(J)}(y_J)^T (y - y_z) \right\|^2 < \varepsilon^2 \right\}\right)$$

Measure of set of points 'y' s.t where

$$\left\| \nabla M^{(J)}(y_J)^T (y - y_z) \right\|^2 < \varepsilon^2$$

To find the volume of this set we can consider integral

$$= \int_{\left\| \nabla M^{(J)}(y_J)^T (y - y_J) \right\|^2 < \varepsilon^2} 1 \, dy \qquad = \int_{\underbrace{(y-y_J)^T \nabla M^{(J)}(y_J)}_{P^T} \underbrace{\nabla M^{(J)}(y_J)^T (y-y_J)}_{P} \leq \varepsilon^2} 1 \, dy$$

our definition of 'p'

$$\boxed{p = \nabla M^{(J)}(y_J)^T (y - y_J)}$$

$$\because \quad \|p\|^2 = p^T p$$

$$\& \quad \|AV\|^2 = V^T A^T A V$$

* Now we don't want to integrate in terms of variable 'y' so we would like to change it.

$$\text{i.e } p = \underbrace{\nabla M^{(J)}(y_J)}_{\text{Matrix}}{}^T \underbrace{(y - y_J)}_{\text{vector}}$$

$$\Rightarrow \boxed{y = y_J + \left(\nabla M^{(J)}(y_J)^T\right)^{-1} P}$$

Inverse Transformation

Since we are changing the variable in Multidimensional Domain, What we are doing is altering the measure. The factor by which we will be altering the measure is it's Jacobian.

Jacobian $\left| \det\left(\nabla M^{(J)}(y_J)^T\right)^{-1} \right|$

$$\Rightarrow \frac{1}{\left|\det\left(\nabla M^{(J)}(y_J)\right)\right|}$$

Our Integral changes to

$$= \int_{p^T p \le \varepsilon^2} \frac{1}{\left|\det\left(\nabla M^{(J)}(y_J)\right)\right|} \, dp$$

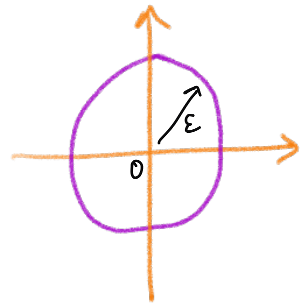
→ Definition of our domain is no longer complicated

$$= \frac{1}{\left|\det\left(\nabla M^{(J)}(y_J)\right)\right|} \int_{p^T p \le \varepsilon^2} dp$$

$p^T p = \|p\|^2 < \varepsilon^2$

$\Downarrow$

$$\boxed{\mu\left(M^{(J)-1}\left(B_\varepsilon(z)\right)\right) \Rightarrow \frac{1}{\left|\det\left(\nabla M^{(J)}(y_J)\right)\right|}\int_{\|p\|^2 \le \varepsilon^2} dp}$$



Ball of Radius ε

$$\underbrace{\sum_{j=0}^{n-1} p_J^2 \le \varepsilon^2}_{\text{centered at '0'}}$$

$$= \frac{1}{\left|\det\left(\nabla M^{(J)}(y_J)\right)\right|}\,\mu\left(B_\varepsilon(0)\right)$$

→ we did all the above calculation to simplify the below expression

$$f_W(z) = \sum_{M(y_J)=z} f_X(y_J)\,\frac{\mu\left(M^{(J)-1}\left(B_\varepsilon(z)\right)\right)}{\mu\left(B_\varepsilon(z)\right)}$$

$$\Rightarrow \boxed{f_w(z) = \sum_{M(y_J)=z} \frac{f_x(y_J)\, \mu(B_\varepsilon(0))}{\mu(B_\varepsilon(z))\, \left|\det \nabla M^{(J)}(y_J)\right|}}$$

$$\mu(B_\varepsilon(0)) = \mu(B_\varepsilon(z))$$

centered at '0' — centered at 'z'

But measure remains the same.

$$\Rightarrow \boxed{f_w(z) = \sum_{M(y_J)=z} \frac{f_x(y_J)}{\left|\det(\nabla M^{(J)}(y_J))\right|}}$$

Statistical Behaviour of output

Statistical behaviour of input

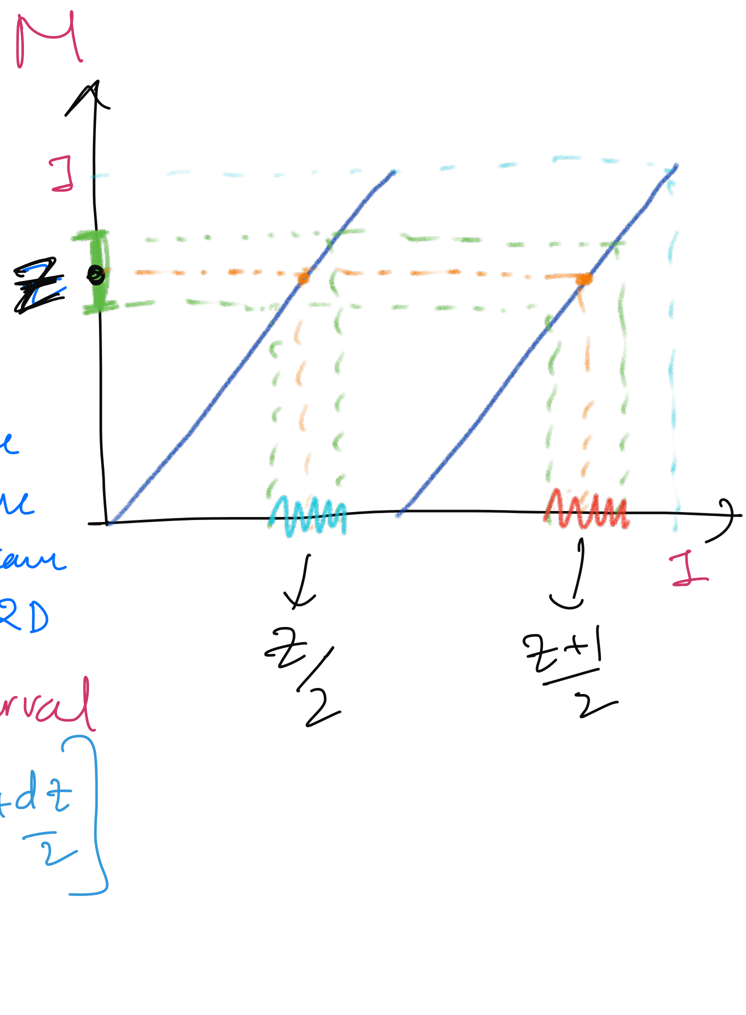
The Box function (or) Transformation function based on 'M'

This is the Fundamental Building Block we are interested in. i.e if we have a Transformation of Stochastic Vectors and we have finite no. of counter images for the point we want to calculate the output then the output PDF can be described as above.

example

$$M : [0,1] \to [0,1]$$

$$M = \begin{cases} 2\eta & \eta \in [0, \tfrac{1}{2}[\\ 2\eta - 1 & \eta \in [\tfrac{1}{2}, 1] \end{cases}$$



Let this be output and the Ball in this case is a line because we are in 2D

Interval

$$\left[z - \frac{dz}{2}, z + \frac{dz}{2}\right]$$

→ This interval has the probability

$$f_W(z)\,dz = f_X\left(\frac{z}{2}\right)\frac{dz}{2} + f_X\left(\frac{z+1}{2}\right)\frac{dz}{2}$$

These two are the Counter Images that 'z' must have

This because

$$M = \begin{cases} 2y \\ 2y - 1 \end{cases}$$

y is the value 'z' can take

$$2 = \det \left| \nabla M(y_J) \right| = \det \left| \frac{\partial}{\partial z} 2z \right|$$

$$\Rightarrow \det |2|$$

$$\Rightarrow 2$$

$$f_W(z) = f_X\left(\frac{z}{2}\right)\frac{1}{2} + f_X\left(\frac{z+1}{2}\right)\frac{1}{2}$$

This is the special case of above mapping formulae derived :

19/12/23

Now, That we have calculated the output of a stochastic signal processing with regard to a stochastic input signal with the help of Counterimages

i.e probability of the output depends on the probability of the input which could generate that particular output.

** Now we can proceed to our original Task i.e

→ Q) How linear Algebraic Things work ?

Linear Algebraic Transformations

Things would get immediately complicated due to the fact that, we derived a formula that works only when

The no. of dimensions of input & output Remain same! Therefore we might have to depart significantly from the above formula.

But, There are few cases where we can apply it!

→ So now if we have to introduce an assumption that our process is LINEAR.

our $M(x)$ is no longer generic function it is the linear mapping of 'x' into w

$$\underbrace{w}_{\in \mathbb{R}^n \;\to\; z} = A \underbrace{x}_{\in \mathbb{R}^n \;\to\; y} = M(x)$$

y → instances

- We will assume $\underbrace{\text{'A' is square}}_{\Downarrow}$ and non-singular

no. of dimensions of input & output remain same

by Non-singular 'A' we need not worry about invertibility.

∗ → In this set of above conditions we can apply the Formula we derived i.e

$$\boxed{f_w(z) = \sum_{M(y)=z} \frac{f_x(y)}{\left|\det(\nabla M(y)\right|}}$$

The reason is that 1st we have to compute the

Counter images $M(y)$ for 'z'

let's see how many counterimages are there in $w = Ax$?

$$\begin{array}{cc} w & Ax \\ \downarrow & \downarrow \\ z & y \end{array}$$

$$y \mid M(y) = z$$

$$Ay = z$$

$$y = A^{-1} z$$

we have one counter image for a 'z'

$$\mathbb{R}^n \longleftarrow \mathbb{R}^n$$

$$w = M(x) = Ax$$

$$w_J = \sum_{k=0}^{n-1} \underbrace{A_{Jk} x_k}$$

Now we have to make the derivative of this function for every possible J & K

i.e if we change J we change the output & it immediately changes 'k' that produce a particular 'J'

Therefore our Matrix

$$\nabla M = \begin{array}{c} J \\ \begin{bmatrix} \frac{\partial M_J}{\partial x_k} & & & \cdots \end{bmatrix} \end{array}$$

- every column corresponds to a different 'J'
- every Row corresponds to a different 'K'

→ differentiating $\boxed{w_J = \sum_{k=0}^{n-1} A_{Jk} x_k}$

wrto x_k

$$\frac{\partial w_J}{\partial x_k} = A_{Jk} = \frac{\partial M_J}{\partial x_k} = \nabla M_J$$

→ Since The Matrix of Gradient ∇M_J is equal to the A_{Jk}

$$\boxed{\nabla M_J = A^T}$$

J → column

K → Row

$$A_{JK} = (A_{KJ})^T$$

This proves that in case of Linear Transformations The Matrix of the Gradient is the Matrix itself:

Now that we know we have a single counter image a very easy way of determining Matrix of a Gradient

$$\boxed{f_w(z) = \frac{f_x(A^{-1}z)}{|\det(A)|}}$$

This is the most elementary Result of the course.

$$\underset{\in \mathbb{R}^m}{W} = \underset{m \times n}{A}\, \underset{\in \mathbb{R}^n}{X}$$

(i) Injection:- $y' \neq y'' \Rightarrow M(y') \neq M(y'')$

each distinct variable has a distinct mapping

(ii) Surjection:-

$$\forall z \quad \exists y \quad M(y) = z$$

all the elements have a counterimage in the Domain.

Any Function which is both Injection & Surjection is called Bijection.

Q) What if our Mapping function is Non-injective?

What we have to do is to recall what Non-injective means for Linear Transformation

They do have special properties when they are Non-injective.

Non-injective $\Rightarrow \quad y' \neq y''$ → not derivatives just indication

but $\quad Ay' = Ay''$

$\Rightarrow \quad A(y' - y'') = 0$

& $\quad y' - y'' \neq 0 \quad \because y' \neq y''$

$\Rightarrow \quad AU = 0 \qquad U = y' - y''$

A vector which is Non zero!

'U' st $AU = 0$

This the definition the vector that stays in the kernel of 'A'

$U \in$ kernel of 'A' (or) The Null Space of 'A'

Therefore when we have Non-injective linear transfor-mations, we have a kernel that contains vectors that are Non-zero.

$U \in \ker A$

let's consider a 2nd point y''' defined such that

$$y''' = y' + \alpha U$$

scalar

kernel vector of 'A'

$$Ay''' = Ay' + \alpha \underbrace{AU}_{=0}$$

$$\Rightarrow Ay''' = Ay' + 0$$

$$\Rightarrow \boxed{Ay''' = Ay'}$$

Since 'α' can be any scalar, we can have

Infinitely many y'''.

i.e There are infinitely many Counter images which goes against the conditions we set when we derived

$$\boxed{f_w(z) = \sum_{M(y)=z} \frac{f_X(y)}{|\det(\nabla M(y))|}}$$

→ we also know from Linear Algebra course that
" Ker A = Null space of 'A' is a subspace"

→ The idea is that the formula we derived is notuseful because the counter images are not finite & are difficult to enumerate

Therefore, we need a way of listing all the possible Counterimages.

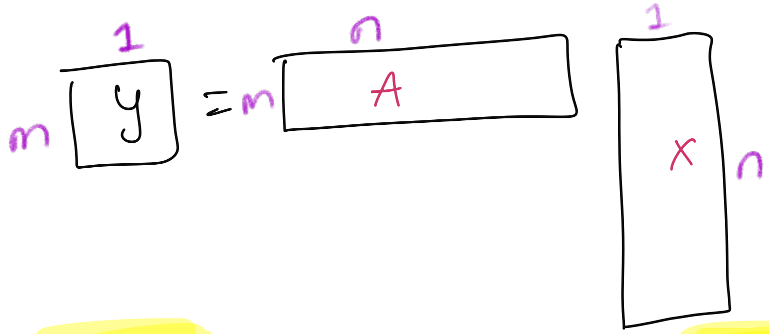
If we can find all the counter images in a Regular way, probably I can find an alternative way to summing from them (Maybe an Integral)

∴ The problem is to find on which kind of Domain I have to integrate ?

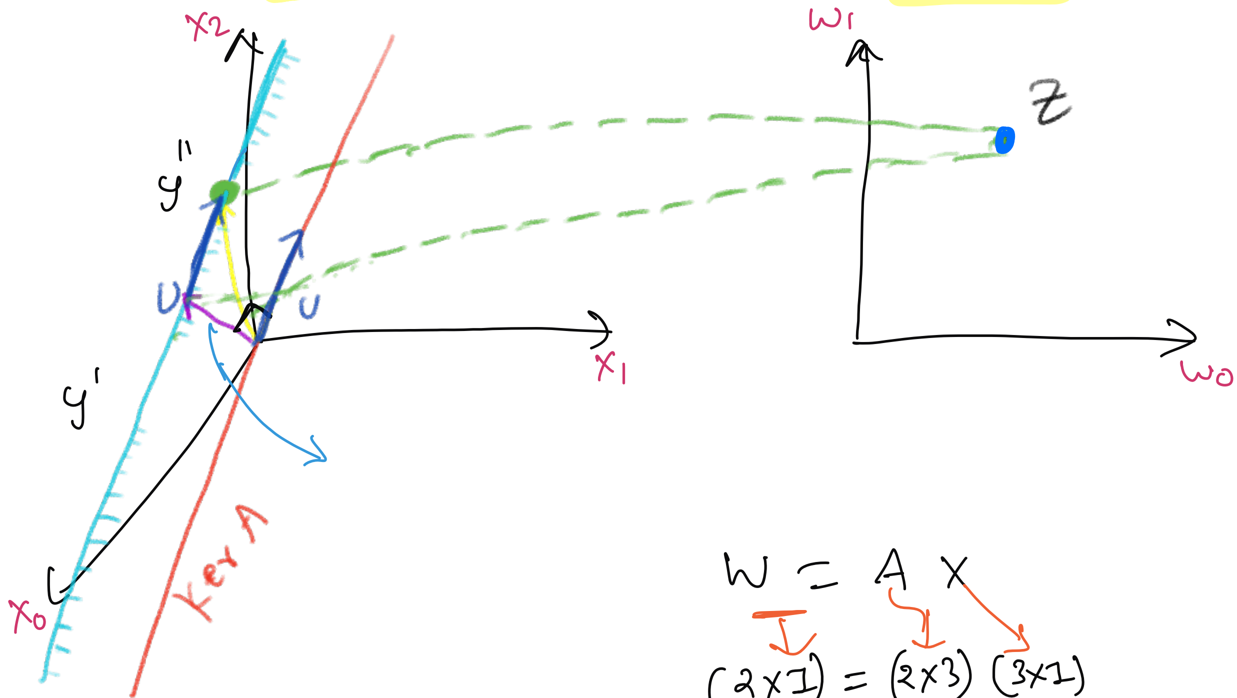
Firstly we will try and solve this problem in a 3D setting!

Slanted Matrix: It is a Matrix which has no. of Rows less than no. of columns.

$m < n$



$n = 3$ $\qquad$ $m = 2$



$$W = A\,X$$

$$(2\times1) = (2\times3)\,(3\times1)$$

For a point 'Z' we have Two counter images y', y''. If we make

$$\therefore \; v = y'' - y' \neq 0$$

$v \in$ Kernel of 'A' since Nullspace is a subspace it has to contain Origin, in this case it is a Line.

* Now that we have 'v' we can generate any Multiple of y' to get a New point y'''

$$\therefore \; \boxed{y''' = y' + \alpha v}$$

Now the set of values which can be counter images of 'Z' is the whole BLUE LINE

→ Our idea, is find a compact method of finding the counter images.

*→ THE BLUE LINE is an offset version of the kernel (REDLINE)

If we find the OFFSET, Then all the possible Counter

Images can be expressed w.r.t the OFFSET & KERNEL

* So, first we have to (i) find the OFFSET & (ii) add it to all possible vectors in the kernel

Finding that OFFSET is called PSUEDO INVERSION

(Moore – Penrose)

The idea is that, we cannot inverse the matrix that are slanted, because they are not Square Matrix and in general it is Non-injective

$$\boxed{z} = \underbrace{\boxed{\;\;A\;\;}}_{\text{Slanted}} \boxed{y}$$

every point (z) has infinite Counter Images.

→ Now, we have to find a special Counter Image which would be our OFFSET, it must the shortest one. This is why it is called psuedo Inversion. Because 'A' is a Rectangular

Matrix.

Just to lay down things in a Mathematical way.

(2) we are squaring the Norm

$$\begin{cases} \min \ \|y\|_2^{2} = y^T y \\ \text{s.t} \ Ay = z \end{cases}$$

Constraint for Counter image

It is a Vector Equality.

$$\boxed{Ay = z} \Rightarrow \begin{array}{ccccc} A & & y & & z \\ \begin{bmatrix} a_0^T \\ a_1^T \\ a_2^T \\ \vdots \\ a_{m-1}^T \end{bmatrix} & & \begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix} & = & \begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} \\ & & y & & \end{array}$$

$$A_{m\times n}\, y_{n\times 1} = z_{m\times 1}$$

- a^T because Generally $a = \begin{bmatrix} \\ \\ \end{bmatrix}$ is a vertical vector.

$$\Rightarrow \boxed{a_j^T y = z_j \quad \text{for } j = 0, 1, \ldots m-1}$$

$$a_j^T y - z_j = 0$$

∴ Lagrangian for optimization

Now we have our objective function which is the Squared Norm $\|y\|_2^2 = y^T y$

We have also our constrains ∴ we can use the Lagrangian function $L(y,\lambda)$

$$\boxed{L(y,\lambda) = y^T y - \sum_{j=0}^{m-1} \lambda_j (a_j^T y - z_j)}$$

Degrees of Freedom

vector of Lagrangian Multipliers

function that must be zero in the constraining expression.

→ We know that the stationarity points are the optimum points for the Lagrangian.

$$\boxed{\nabla_{y,\lambda} L(y,\lambda) = 0}$$

(1)

$$\nabla_y L \Rightarrow 2y - \sum_{j=0}^{m-1} \lambda_j a_j = 0$$

✱ $y^T y = \sum_{j=0}^{n-1} y_j^2$

$$\Rightarrow \frac{\partial}{\partial y_0}(y^T y) = 2y_0, \quad \frac{\partial}{\partial y_1}(y^T y) = 2y_1$$

$$\Rightarrow 2y = \begin{bmatrix} 2y_0 \\ 2y_1 \\ \vdots \\ 2y_{n-1} \end{bmatrix}$$

(ii) $$\nabla_\lambda L = \underline{Ay = z}$$

→ Now we have a s/s of eq^ns to solve for 'y'
once we have a vector 'y' we would have a solution for our original Minimization problem of OFFSET.

$$\nabla_y L \longrightarrow \left\{ \begin{array}{ll} 2y - \underline{\sum_{J=0}^{m-1} \lambda_J a_J} = 0 & \text{—(1)} \\ Ay = z & \text{—(2)} \end{array} \right.$$

$$\nabla_\lambda L \longrightarrow$$

This expression is the Linear combination of vectors

$$\sum_{j=0}^{m-1} d_j a_j = d_0 \begin{bmatrix} a_0 \end{bmatrix} + d_1 \begin{bmatrix} a_1 \end{bmatrix} + \cdots + d_{m-1} \begin{bmatrix} a_{m-1} \end{bmatrix}$$

$$\Rightarrow \underbrace{\begin{bmatrix} | & | & & | \\ a_0 & a_1 & \cdots & a_{m-1} \\ | & | & & | \end{bmatrix}}_{A^T} \underbrace{\begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{m-1} \end{bmatrix}}_{d}$$

$$\Rightarrow A^T d$$

$$\therefore \nabla_y L = 2y - A^T d = 0$$

$$\Rightarrow y = \frac{A^T d}{2}$$

Substitute in (2)

$$\nabla_d L \Rightarrow Ay = z$$

$$\Rightarrow A \frac{A^T d}{2} = z$$

$$\Rightarrow \boxed{d = 2 (AA^T)^{-1} z}$$

$$\Rightarrow y = \frac{1}{2} A^T \lambda$$

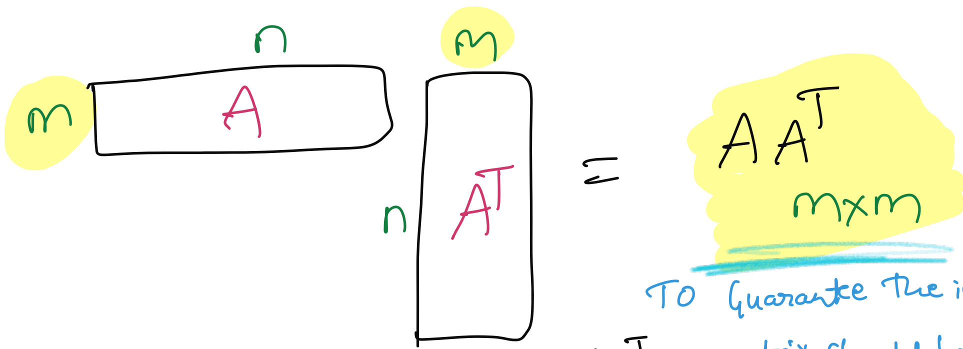
$$\Rightarrow \frac{1}{2} A^T 2 (AA^T)^{-1} z$$

**

$$\boxed{y \Rightarrow A^T (AA^T)^{-1} z}$$

This is the Moore-Penrose psuedo inverse for non-injective maximum Rank Matrices.

** we will will analyze why do you need Maximum Rank when we analyze How to Guarantee an inversion of $\boxed{AA^T}$



To Guarantee the inversion of AA^T the matrix should have Maximum Rank.

it preserves the shortest dimension
i.e 'm'

→ To prove that AA^T can be Inverted, one thing we can do is write it in a Quadratic form. The resulting Quadratic form is strictly positive Definite.

The fact that QF is strictly positive Definite depends on the Eigen values.

i.e if Eigen values > 0 then it would be positive Definite. Then our Matrix would be Invertible.

** So instead of checking Invertibility of AA^T we will check the Quadratic form.

$$\underbrace{U^T (AA^T) U > 0}_{\text{strictly positive Definite}} \qquad U \neq 0$$

$$\Rightarrow \underbrace{U^T A}_{V^T} \underbrace{A^T U}_{V} > 0$$

∵ Dot product is Associative.

$$\Rightarrow \quad V^T V = \|V\|_2^2 > 0 \qquad \|V\| = 0$$

iff $V = 0$

i.e $A^T U = 0$

$$A^T U = 0$$

$$\begin{bmatrix} | & | & & | \\ a_0 & a_1 & \cdots & a_{m-1} \\ | & | & & | \end{bmatrix} \begin{bmatrix} U_0 \\ U_1 \\ \vdots \\ U_{m-1} \end{bmatrix} = 0$$

By assumption $a_0, a_1, \cdots a_{m-1}$ are linearly independent

$$U_0 \begin{bmatrix} a_0 \end{bmatrix} + U_1 \begin{bmatrix} a_1 \end{bmatrix} + \cdots + U_{m-1} \begin{bmatrix} a_{m-1} \end{bmatrix} = 0$$

$\therefore$ The above case is possible only if

$$U = 0 = \begin{bmatrix} U_0 \\ U_1 \\ \vdots \\ U_{m-1} \end{bmatrix}$$

→ This ensures The Quadratic form is strictly positive Definite.

finally

$$\boxed{\text{OFFSET} = A^T (AA^T)^{-1} z}$$

→ Now to generate all the possible counter Images of 'z' to new would like to enumerate

To generate a formula for the PDF of the output

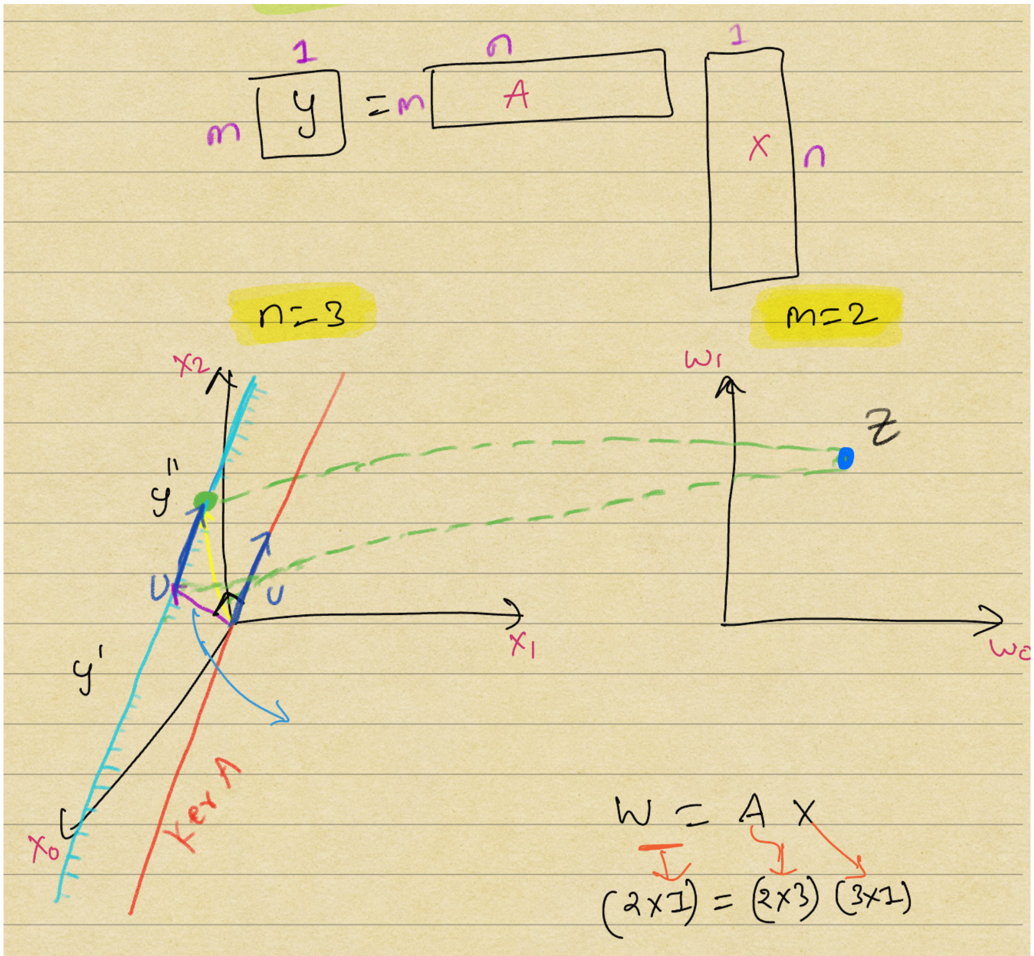
** Step-2: Is to Enumerate all the possible vectors in the kernal of Matrix 'A'

Note: our First step was to find the OFFSET

We need to find the equation for the RED LINE i.e the kernel of 'A', i.e the Nullspace of 'A' (A subspace)

(We will exploit this fact)

↑



→ Since, the kernel of 'A' is a Subspace we can express it as the linear combination of the vector in its Basis. So we will get the whole Base.

All of them will provide a different Counterimage 'U'

Dimension of the Null space

Base

$U \in \ker$ $\qquad U = \sum_{j=0}^{n-m-1} \alpha_j e_j$

$\dim \ker A = n-m$ $\qquad m\,\boxed{A}^{\,n}$ → Max Rank

→ The above conditions gives us the conditions to do the "Moore-penrose" Psuedo Inversion trick.

To do Psuedo Inversion there are many algorithms but we will choose Gram-Schmidt Algorithm

It allows us to find the Base vectors for the kernel of the Matrix.

Span → $\underbrace{a_0, a_1, \ldots\ldots\ a_{n-1}} \in \mathbb{R}^n$

- we are given a sequence of vectors &
- the task of the Gram-Schmidt mechanism is to analyze these vectors one after the other & producing a series of Orthonormal vectors.

Span → $\underbrace{b_0, b_1, \ldots\ldots}$

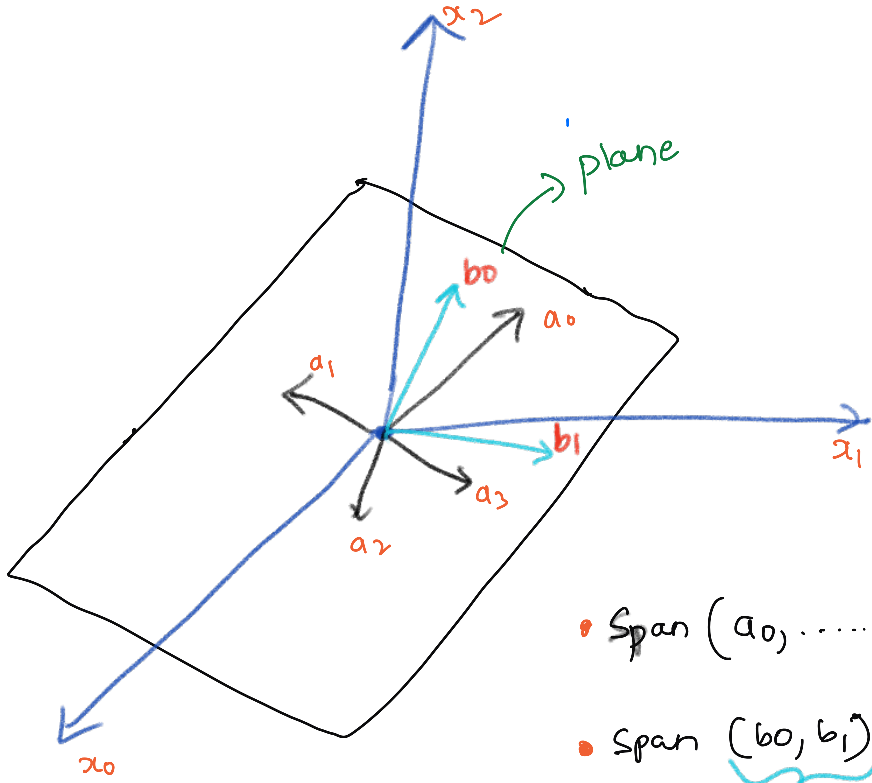
orthonormal vectors which form Basis of the Span of the vectors

→ Span of a set of vectors is set of all possible linear combinations. So all the possible linear combinations of 'A' can be produced by linear combinations of vectors .

$$b = \{b_0, b \ldots\}$$

→ The only difference is the vectors 'a' can be arbitrary, The vectors 'b' are orthonormal and in a number which are strictly necessary to produce all the linear combinations.

This behaviour is depicted in the Drawing below



- Span $(a_0, \ldots, a_{n-1})$
- Span $(\underbrace{b_0, b_1}_{\text{set of orthonormal vectors}})$

Here the plane can be spanned completely by two orthonormal vectors on that plane i.e b_0, b_1

→ We need to figure out from a_0, a_1, a_2, a_3 the Spanning vectors b_0, b_1

a_0 gets transformed only by normalization cos they remain the same otherwise.

$\therefore$ (1) $a_0 \longrightarrow \boxed{b_0 = \dfrac{a_0}{\|a_0\|}}$

unit Normal vector, which is aligned in the same direction as a_0.

(2) Things get slightly complicated for 'a_1'

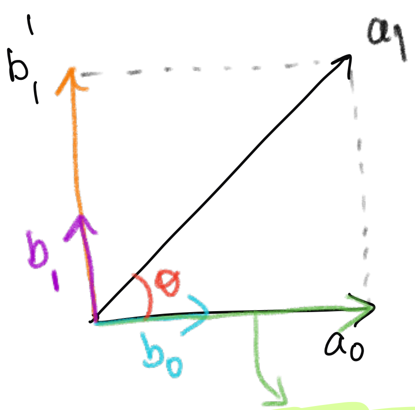
$$a_1 \longrightarrow b_1' = a_1 - \underbrace{(b_0^T a_1)}_{\text{Scalar Product of } a_1 \text{ w.r.to } b_0} b_0$$

$$\boxed{b_1 = \frac{b_1'}{\|b_1'\|}}$$

Scalar Product of a_1 w.r.to b_0

$\Downarrow$

$\|a_1\| \cos\theta$



$b_0^T a_1 = \|b_0\| \|a_1\| \cos\theta_0 = \|a_1\| \cos\theta$

$\rightarrow$

$$a_2 \longrightarrow b_2' = a_2 - (b_0^T a_2) b_0 - (b_1^T a_2) b_1$$

we remove two projections of a_2 along b_0 & b_1 & then we find b_2'

$$\boxed{b_2 = \frac{b_2'}{\|b_2'\|}}$$

Normalized

Similarly we continue with a_3

$$a_3 \longrightarrow b_3' = a_3 - (b_0^T a_3) b_0 - (b_1^T a_3) b_1 - (b_2^T a_3) b_2 = \underline{0}$$

b_0, b_1, b_2 are orthonormal

We get a zero because we have a plane to describe and only two vectors b_0, b_1 are enough i.e two dimensions are enough.

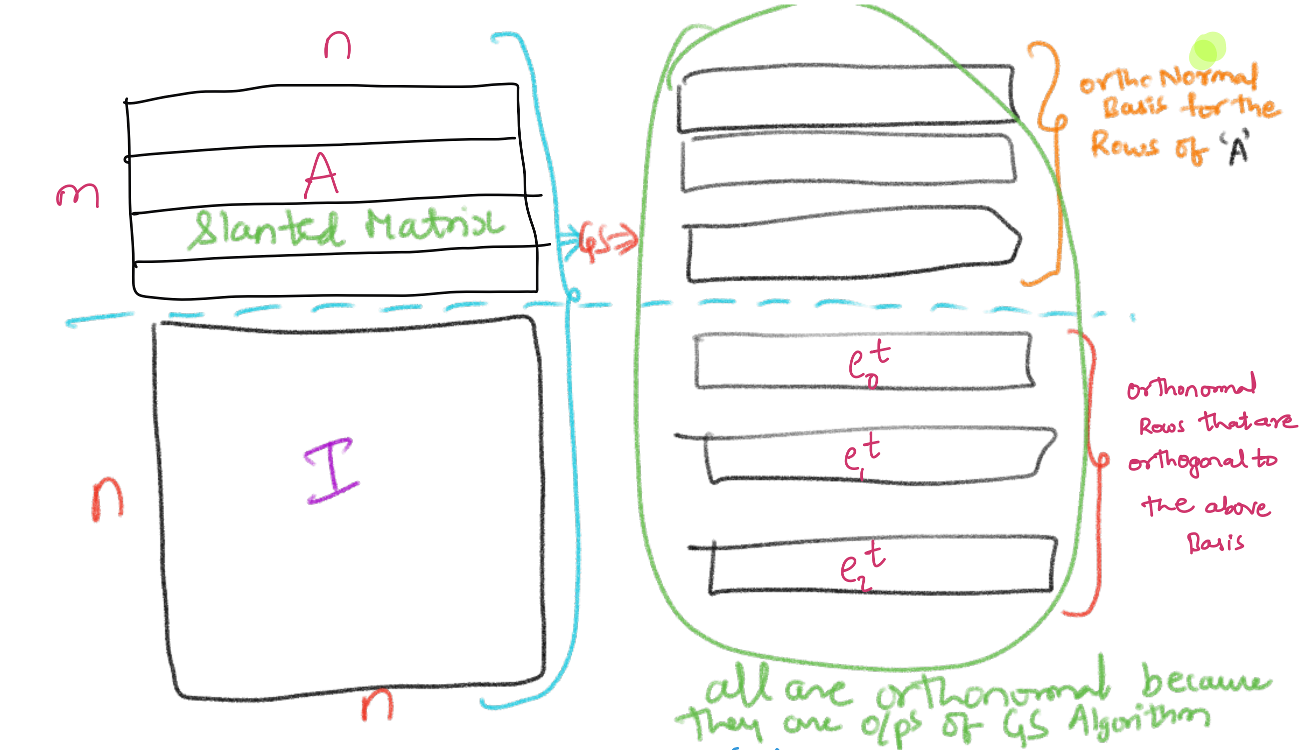
Q) How can we use this Orthonormalization to get the kernel of the Matrix 'A' ?

Ans

Application to finding the basis of $\ker A$

we have the slanted 'A' matrix i.e $m < n$
i.e rows < columns

we can see 'A' below.



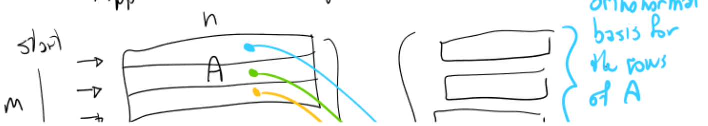
→ we are adding to the Matrix 'A' an Identity Matrix with $n \times n$ we get a New Matrix with $(m+n) \times n$

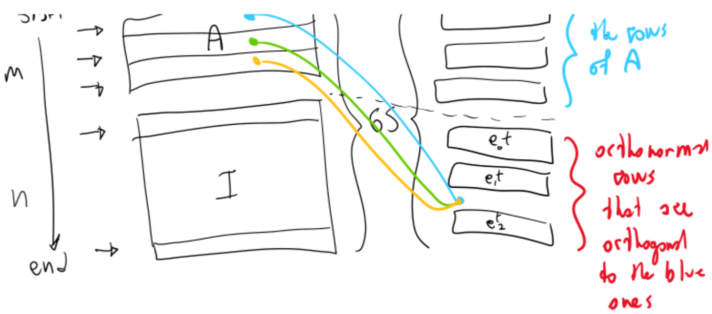
Now we will feed the New matrix to the Gram-Schmidt to find the Orthonormal Basis. & orthonormal Rows that are $\perp$lar to the orthonormal Rows.

$$e_j \in \ker A$$

$$A e_j = 0$$

Application to finding the base of ker A





They are the Basis for the kernel of 'A'

$$A e_j = \begin{bmatrix} A \end{bmatrix} \begin{bmatrix} e_j \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0$$

$$\Downarrow$$

$e_j \in \ker A$

$$W = \boxed{A^{max}_{rank}}\, X$$

$$f_W(z) = \frac{1}{\sqrt{\det(AA^t)}} \underbrace{\int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty}}_{n-m \text{ integrals}} f_X\left(\boxed{A^t(AA^t)^{-1}z} + \sum_{j=0}^{n-m-1} \alpha_j e_j\right) d\alpha_0 \cdots d\alpha_{n-m-1}$$

base vector of ker A

$$f_W(z) = \sum_{M(y)=z} \frac{f_X(y)}{|\det \nabla M(y)|}$$

Non surjective $\exists z$ do not have a counterimage

→ * Now we have all the ingredients to derive

$f_w(z)$

$$
\boxed{f_w(z) = \frac{1}{\sqrt{\det AA^T}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_X\left(A^T (AA^T)^{-1} z + \sum_{j=0}^{n-m-1} \alpha_j e_j \right) d\alpha_0 \ldots d\alpha_{n-m-1}}
$$

* it is factor that tell us How precisely the Measurements are scaled when it pass through mapping of 'A'

** We are able to identify all the possible Counter Images of any point 'z' and to characterize them we know that they are made of an OFFset which is the Psuedo Inverse of 'z' + Linear combination of basis vector of the kernel of 'A'.

→ The Formula is exactly the same as we derived earlier

i.e

$$
\boxed{f_w(z) = \sum_{M(y)=z} \frac{f_X(y)}{\left|\det(\nabla M(y))\right|}}
$$

* Therefore once we are given a Matrix i.e. Started then we can go from stochastic characterization of input $f_X(y)$ to stochastic characterization of output

$f_W(z)$ and to do so is what the Box that produces output in this case Box is Matrix A