The last class!

1) Review from the perspective of "equivalence of matrices"

2) An application - Perron-Frobenius Theorem.

Recall: Last time we defined the notion of an equivalence relation.

An example: similarity of matrices.

We say $A \sim B$ are similar if
A: $n\times n$-matrices

$\exists$ C - an invertible $n\times n$-matrix, s.t
$B = C^{-1}AC$.

Another example: For symmetric $n\times n$-matrices, say that A and B are equivalent if
$\exists$ P - an orthogonal matrix s.t. $P^{-1}AP = B$.

In these terms, the spectral theorem says that any symmetric matrix is equivalent to a diagonal matrix

one more equivalence relation :

- Can make an equivalence relation on $n \times m$ - matrices

say $A \sim B$ if there exists an $m \times n$ - matrix C and an $m \times m$ - matrix D both <u>invertible</u>

s.t.

$$AC = DB$$

$(m \times n) \cdot (n \times m)$ $= m \times n$ - matrix

$\underbrace{D \; B}_{n \times m \text{ - matrix.}}$ with D: $m \times m$, B: $n \times m$

This means the following :

$B, A : V \to W$

n-dim, m-dim

$$\begin{tikzcd} V \arrow[r, "A"] & W \\ V \arrow[u, "C"', "S"] \arrow[r, "B"'] & W \arrow[u, "S", "D"'] \end{tikzcd}$$

"diagram commutes" :

$DB : V \to W$
$\shortparallel$
$AC : V \to W$

$$\begin{array}{cc} & n \\ m & \begin{bmatrix} \quad & \quad \end{bmatrix} \end{array} \begin{array}{cc} n & \\ \begin{bmatrix} \quad & \quad \end{bmatrix} & n \\ C & \end{array}$$

$= m \times n$

This means: we can change the basis in V and <u>independently</u> change the basis in W.

<u>Rank Theorem</u> Under this relation, any two $m \times n$ - matrices of the same rank are equivalent !

<u>Pf</u> :

Can choose bases in V and W so that the matrix looks like:

$$\left[\begin{array}{ccc|c} 1 & & & \\ & 1 & & 0 \\ & & 1 & \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

see hw :
row operations
$\Leftrightarrow$ changes of
basis in V, W

Much harder theorem

Back to $n\times n$-matrices, $A : V \to V$

Back to similarity relation: you only choose a basis in V one time

$$A \sim B \iff \exists\, C \in GL_n(\mathbb{F}) : \quad A = C^{-1} B C$$

How much can we simplify a matrix up to similarity?

- If the characteristic polynomial of A has n distinct roots

  $$p_A(\lambda) = \det(A - \lambda I)$$

  then A has a basis of eigenvectors

  In this basis, A is diagonal!

  (for each v_i, $Av_i = \lambda_i v_i$, so the ith column of A is $\begin{bmatrix} 0 \\ 0 \\ \lambda_i \\ 0 \\ 0 \end{bmatrix} \leftarrow$ ith place)

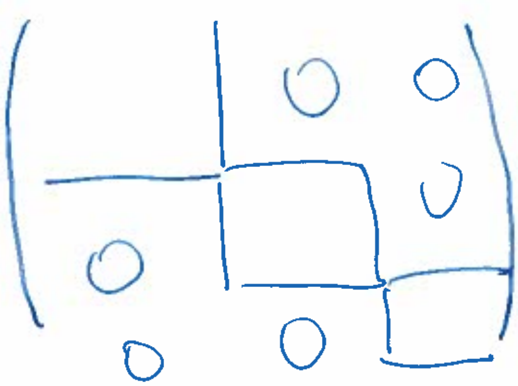
  in the eigenvector basis, the matrix of A

  is $\begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix}$.

- If $p_A(\lambda)$ has multiple roots: $p_A(\lambda) = (x-\lambda_1)^{k_1} \cdots (x-\lambda_n)^{k_n}$

  then cannot always make it diagonal:

  But can get it to Jordan Normal Form (cannot prove)



each block looks like:

$$\begin{pmatrix} \lambda & 1 & & 0 \\ & \lambda & 1 & \\ & & \lambda & 1 \\ 0 & & & \lambda \end{pmatrix}$$

the same eigenvalue

For a multiple eigenvalue, get smth like:

$$\begin{pmatrix} \lambda & 1 & & & \\ & \lambda & 1 & & \\ & & \lambda & 0 & \\ & & & \boxed{\lambda} & 0 \\ & & & & \boxed{\lambda} \end{pmatrix}$$

size 3 block

size 1 blocks

each block $\leftrightarrow$ one eigenvector.

λ has algebraic multiplicity 5. $P_A(x)$ has root of mult. 5 at λ.

here $\dim \ker (A-\lambda I)$ is 3.

Aside

not part of this course

Hard part: dealing with the situation when we do not have "enough" eigenvectors: when geometric multiplicity $\dim\ker(A-\lambda I)$ is smaller than algebraic multiplicity of λ.

Then we need to find more basis vectors $v_1, v_2, \ldots$ such that:

$(A-\lambda I)v_1 = v_0$ ← the eigenvector

$(A-\lambda I)v_2 = v_1$

$\ldots$

This gives the Jordan block

$$\begin{array}{c} \begin{matrix} v_0 & v_1 & v_2 \end{matrix} \\ \begin{pmatrix} \lambda & 1 & \\ & \lambda & 1 \\ & & \lambda \end{pmatrix} \end{array}$$

here:

$Av_0 = \lambda v_0$

$Av_1 = \lambda v_1 + v_0$

$Av_2 = \lambda v_2 + v_1$

$\ldots$

this is hard because $A-\lambda I$ is not invertible! So hard to prove they exist; also they are not unique – how to make good choices?

2) Now, Applications

Today: applications: (quick sketch)
- discrete dynamical systems
- stochastic matrices, probability, Markov chains

Idea: Many situations where you have "state" (a bunch of numbers) and apply linear operator, get a new "state".

Then do it many times.

Example: "state" could be - populations of rabbits and foxes.

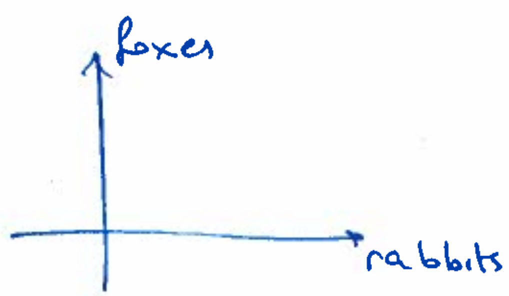
linear operator could be encoding how these populations influence each other:

- number of rabbits eaten by foxes is proportional to the number of foxes.

born: proportional to how many you have

$$\begin{array}{c|cc} & r & f \\ \hline r & \multicolumn{2}{c}{\multirow{2}{*}{$\begin{pmatrix} \overset{*}{a_{11}} & \overset{*}{a_{12}} \\ \overset{*}{a_{21}} & \overset{*}{a_{22}} \end{pmatrix}$}} \\ f & & \end{array}$$

initial state: $\begin{pmatrix} R_0 \\ F_0 \end{pmatrix}$ ← # rabbits, ← # foxes



$$\begin{pmatrix} R_1 \\ F_1 \end{pmatrix} = A \cdot \begin{pmatrix} R_0 \\ F_0 \end{pmatrix}$$

$\begin{pmatrix} R_1 \\ F_1 \end{pmatrix}$: populations after 1 year

A: 2x2 matrix based on your predictions

$$R_1 = a_{11} R_0 + a_{12} F_0$$

a_{12}: neg. number accounting for fox food.

a_{11}: positive number "percentage of births" and death of rabbits.

After one year: $\begin{pmatrix} R_1 \\ F_1 \end{pmatrix} = A \begin{pmatrix} R_0 \\ F_0 \end{pmatrix}$

After n years: $\begin{pmatrix} R_n \\ F_n \end{pmatrix} = A^n \begin{pmatrix} R_0 \\ F_0 \end{pmatrix}$

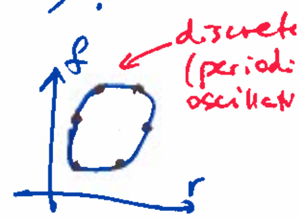
(Note: you could have seen predator-prey model in terms of differential equations:

$f(t)$, $r(t)$ — # foxes/rabbits at time t.

$$\underline{f'(t)} = \underline{a_{11}} f(t) + a_{12} r(t)$$

Our model is a discrete version of this).

(individual states, not continuous time).

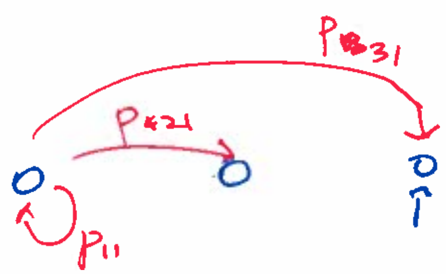


(Note: look at linear recurrences).

In many situations, such a system will approach a steady state: from any initial state, it will approach some fixed state as n gets large. ("equilibrium").

Finite automaton "vending machine"

Markov chain



Your imaginary machine is a list of probabilities p_{ij} of going from state i to state j

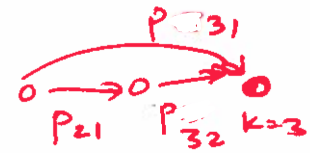
$p_{ij} \geq 0$. $\sum_{j=1}^{n} p_{ij} = 1$ ← we have to go somewhere from state i

Let $\bar{s} = (s_1, \ldots, s_n)$ ← probablities of our states. (initially)

Let $P = (p_{ij})$

The new probabilities are $P\bar{s}$

$$\begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ \vdots & & & \\ p_{n1} & - & \text{———} & p_{nn} \end{pmatrix} \begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix}$$



The k^{th} coordinate of this vector is the probability that we ended up in state k: we could have come to it <u>from</u>:

state 1 with prob. $p_{k1} \cdot s_1$

+

state 2 with prob. $p_{k2} \cdot s_2$

$-$ ————

————————————————

$$\underbrace{\sum_{j=1}^{n} p_{kj} s_j = \text{the } k^{th} \text{ entry of } P.s}$$

<u>The properties of P:</u>
- $p_{ij} \geq 0$
- columns sum to 1.

positive <u>stochastic</u> <u>matrix</u>

(<u>Note</u>: predator-prey model doesn't satisfy these conditions).

Perron - Frobenius Theorem

1) A stochastic matrix has eigenvalue 1 and unique (up to scaling) eigenvector corr. to this eigenvalue

2) All other eigenvalues satisfy $|\lambda_0| < 1$. (if $a_{ij} > 0$).

Pf: Let A be a stochastic matrix.

Consider A^T - the transpose of A.

Then the rows of A^T sum to 1.

$$A^T \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} \text{sum of row 1} \\ - \quad - \\ \text{sum of row n} \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \leftarrow \text{eigenvector of } A^T \text{ with eigenvalue } 1$$

But A and A^T have the same char. poly.

$$= \det(A - \lambda I) = \det(A^T - \lambda I^T) \underset{\substack{\uparrow \\ \text{transpose doesn't affect det.}}}{=} \det(A - \lambda I)$$

So 1 is an eigenvalue of A

So there must be an eigenvector of A.

$Aw = w$, Then $A^n w = w$. ← "steady state"

2) $Av = \lambda v$

$$\left(\sum_{j=1}^{n} a_{jk} v_j\right)_{k=1}^{n} \quad \leftarrow \text{easy to check.}$$

$$\left|\sum a_{jk} v_j\right| \le \sum |a_{jk}| |v_j| \le v_j \Big(\underbrace{\sum |a_{jk}|}_{1}\Big) \Rightarrow |\lambda| < 1.$$

When A is diagonalizable, this means:

$$A^m = C^{-1} \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix}^m C$$

$\lambda_1 = 1$.

$\to 0$ as $m \to \infty$ b/c $|\lambda_i| < 1$ for $i \neq 1$.

So: $A^m \xrightarrow[m\to\infty]{}$ projector onto the 1-dim subspace spanned by the steady state w.

Ex: Google page rank is "the steady state of the "google matrix".

(see the link on the webpage)
