TensorFlow

Part I

VECTOR SPACES, LINEAR TRANSFORMATION, EIGENVALUES AND EIGENVECTORS

Ayokunle Awelewa (Covenant University)

@

The 3rd Google TensorFlow College Outreach Bootcamp and FEDGEN Mini-Workshop

Date: 11th to 13th December, 2023

Sponsored By Google

Contents

Preamble

- Linear algebra is a vast branch of mathematics.

- It is the study of linear equations and functions together with their representations, operations, and manipulations using vector-matrix symbolism.

- Knowledge of linear algebra is essential in many fields—in engineering, natural and social sciences, art and humanity, etc.

- Vectors and matrices, which are structured arrays of numbers representing some objects or phenomena, are the foundation of linear algebra.

- Linear algebra tools make solutions of systems of linear equations readily available.

- For example, determining linear regression model coefficients reduces to solving a system of linear equations which can be handled smoothly using these tools.

- The weights between neurons in neural networks are represented as matrices which are operated on as the networks undergo training in a forward and backward manner.

- Eigenvalues and eigenvectors are concepts from linear algebra useful in principal component analysis, spectral clustering, etc.

Linear Vector Spaces

Background

- A vector is a finite collection of numbers.
- In a real dimensional space, a vector $v \in \mathbb{R}^n$ is called an n-vector or n-dimensional vector, and is given as

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \text{ or } \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$$

  with the dimension $\dim(v) = n$

- A vector is called a column vector if its elements are vertically arranged as given above; it is a row vector if its elements are horizontally arranged.

- A set X is called a linear vector space over a field F, which may be considered to be a field of either real or complex numbers, if the following conditions are satisfied:

$$\left.\begin{array}{lll} \text{i.} & u+v \in X & for\ all\ u,v \in X \\ ii. & \alpha u \in X & for\ all\ u \in X \ \text{ and } \alpha \in F \end{array}\right\}$$

(closure property)

$$iii.\ u+v = v+u \quad for\ all\ u,v \in X$$

(commutative)

iv. $u + (v + w) = (u + v) + w \;\; for\, all\, u, v, w \in X$

(associative)

v. $u + v = u + w \;\; for\, all\, u \in X \Rightarrow v = w$

(equivalence)

vi. $\alpha(u + v) = \alpha u + \alpha v \;\; for\, all\, u, v \in X$ and $\alpha \in F$

vii. $(\alpha + \beta)u = \alpha u + \beta u \;\; for\, all\, u \in X$ and $\alpha, \beta \in F$

viii. $\alpha\beta u = \alpha(\beta u) \;\; for\, all\, u \in X$ and $\alpha, \beta \in F$

ix. $1u = u \ for\, all\, u \in X$ and 1 is the unit element of the field F

x. $0 \in X$ and $0 + \mathrm{u} = u \ for\, all\, u \in X$

- Addition and scalar multiplication operations hold on all vector spaces.
- The simplest vector space has only one element. That is, {0} is a vector space.

Basic vector operations and properties

Transposition:

- The transpose of a vector $v \in \mathbb{R}^n$ is denoted by v^{T}:

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}$$

$$\begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

```
#Vectors
```

```
import numpy as np
 # Create row vectors
v = np.array([2, 12, 4, 5, 5])
u = np.array([1, 10, 3, 2, 3])
# Create column vectors
q = np.array([[1], [7], [4]])
r = np.array([[20], [11], [15]])
#Transpose column vectors
r_trans=r.T
v_trans=v[:, np.newaxis]
# Find the number of elements in a vector
num_v = len(v)
#Output vectors
print(« Vector v:"), print(v)
print("\nVector r:"), print(r)
print("\nVector r transposed:"), print(r_trans)
```

Addition/subtraction:

- If $\mathrm{u}, v \in \mathbb{R}^n$ then

$$u \pm v = \{w \in \mathbb{R}^n | w_i = u_i \pm v_i, i = 1, 2, \cdots n\}$$

Scalar-vector multiplication:

- For a scalar $\alpha \in \mathbb{R}$ and a vector $\mathrm{v} \in \mathbb{R}^n$,

$$\alpha v = \{w \in \mathbb{R}^n | w_i = \alpha v_i, i = 1, 2, \cdots n\}$$

#Addition/subtraction and scalar multiplication

```
import numpy as np
# Create two vectors
qq = np.array([[1], [7], [4]])
rr = np.array([[20], [11], [15]])
# Add/subtract the vectors
x = qq+rr
#Scalar multiplication
alpha= 10
xx=alpha*x
#Output vectors
print("Vector x:"), print(x)
print("\nVector xx:"), print(xx)
```

Inner product:

- The inner product, which is also called the dot product (represented symbolically as "$.$") and sometimes denoted as $\langle .\,,.\rangle$, is a scalar operation carried out between two vectors of the same dimension. If $\mathrm{u}, v \in \mathbb{R}^n$, then

$$u.v = \langle u, v\rangle \equiv u^{\mathrm{T}} v = \sum_{i=1}^{n} u_i v_i$$

Outer product:

- The outer product (represented symbolically as "×") is between two vectors of the same or different dimension. It is the set of all ordered products of the elements of both vectors. If $\mathrm{u} \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$, then the outer product
- $$u \times v = uv^{\mathrm{T}} = \begin{bmatrix} u_1 v_1 & u_1 v_2 & \cdots & u_1 v_m \\ u_2 v_1 & u_2 v_2 & \cdots & u_2 v_m \\ \vdots & \vdots & \ddots & \vdots \\ u_n v_1 & u_n v_2 & \cdots & u_n v_m \end{bmatrix}$$
- Note that: $uv^{\mathrm{T}} = \left(vu^{\mathrm{T}}\right)^{\mathrm{T}}$

```
#Inner product and outer product
```

```
import numpy as np
# Create column vectors
u = np.array([[1], [7], [4]])
v = np.array([[20], [11], [15]])
#Inner product
u_trans=u.T
InnPr=np.dot(u_trans,v)
#Outer product
u_trans=u.T
v_trans=v.T
OutPr_u_out=np.dot(v, u_trans)
OutPr_v_out=np.dot(u, v_trans)
#Output vectors
print(« Vector u:"), print(u)
print("\nVector v:"), print(v)
print("\nVector u transposed:"), print(u_trans)
print("\nVector v transposed:"), print(v_trans)
print("\nInner product of u and v:"), print(InnPr)
print("\nOuter product of v and u:"), print(OutPr_u_out)
print("\nOuter product of u and v:"), print(OutPr_v_out)
```

Norms:

- The norm of a vector is used to describe the magnitude of the vector, that is, how big the vector is.
- It is a measure of the size or length of vectors.
- The Euclidean norm (or 2-norm) of a vector $v \in \mathbb{R}^n$ is given by

$$\|v\| = \|v\|_2 = \sqrt{v^{\mathrm{T}} v} = \sqrt{\sum_{i=1}^{n} v_i^2}$$

- Note that vector v can be expressed as

$$v = \|v\|_2 \hat{v}$$

where $\hat{v}$ is the unit vector in the direction of v.

- Given vectors $\mathrm{u}, v \in \mathbb{R}^n$ and a scalar α, the properties of a norm are:
  - $\|u\| \geq 0$ (nonnegativity)
  - $\|u\| = 0$ if and only if $u = 0$ (definiteness)
  - $\|\alpha u\| = |\alpha| \|u\|$ (homogeneity)
  - $\|u + v\| \leq \|u\| + \|v\|$ (triangle inequality)

- Other norms are:

Infinity norm:

$$\|v\|_\infty = \max_{1 \le i \le n} |v_i|$$

General p-norm:

$$\|v\|_p = \left(\sum_{i=1}^{n} |v_i|^p \right)^{1/p}$$

#Norms

```
import numpy as np
# Create a vector
v = np.array([1, 5, 2])
# Euclidean norm (L2 norm)
euc_norm = np.linalg.norm(v)
# General p-norm (Lp norm) with p=5
p_norm = np.linalg.norm(v, ord=5)
# Infinity norm (L∞  norm)
inf_norm = np.linalg.norm(v, ord=np.inf)
print("Euclidean Norm:", euc_norm)
print("5-Norm:", p_norm)
print("Infinity Norm:", inf_norm)
```

Linear combination, dependence, and independence of vectors:

- Given a set of vectors $V = \{V_1, V_2, \cdots, V_m\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, m$, a vector $\mathrm{z} \in \mathbb{R}^n$ is a **linear combination** of the elements in V if there exists $\alpha_i \in \mathbb{R}$, $i = 1, 2, \cdots, m$, such that

$$z = \sum_{i=1}^{m} \alpha_i V_i$$

- A set of vectors $V = \{V_1, V_2, \cdots, V_m\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, m$, is said to be **linearly dependent** if any of its elements is a linear combination of the others. That is, if there exists $\alpha_i \in \mathbb{R}$, $i = 1, 2, \cdots, m$, such that

$$V_j = \sum_{\substack{i=1 \\ i \neq j}}^{m} \alpha_i V_i, \quad \text{for some } j \in \{1, 2, \cdots, m\}$$

- For a set of vectors $V = \{V_1, V_2, \cdots, V_n\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, n$, is said to be **linearly dependent** if the determinant of the matrix formed by stacking the vectors together is zero. That is, if

$$\det(\begin{bmatrix} V_1 & \cdots & V_n \end{bmatrix}) = 0$$

- A set of vectors $V = \{V_1, V_2, \cdots, V_m\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, m$, is said to be **linearly independent** if none of its elements is a linear combination of the others.

- Consider the set of vectors $V = \{V_1, V_2, \cdots, V_m\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, m$.

- Consider the set of vectors $V = \{V_1, V_2, \cdots, V_m\}$, $V_i \in \mathbb{R}^n$, $i = 1, 2, \cdots, m$.
  - If $m > n$, then it is a given that V is linearly dependent.
  - If $m = n$, then V is linearly independent if and only if
$$\det(\begin{bmatrix} V_1 & \cdots & V_n \end{bmatrix}) \neq 0$$
  - If $m < n$, then V is linearly independent if the only solution to the system of linear equations
$$V_1 + \alpha_2 V_2 + \cdots + \alpha_m V_m = 0$$
is trivial.

Linear Vector Subspaces

Vector subspaces

- A subspace S of $\mathbb{R}^n$ is any set of vectors which is closed under addition and scalar multiplication. Consider vectors $u, v \in S$ and a scalar α. Then S is a subspace if
    - $u + v \in S$
    - $\alpha u \in S$

Span

- If a subspace represents all linear combinations of a given set of vectors, say $\{V_1, V_2, \cdots, V_m\}$ in $\mathbb{R}^n$, then the subspace is said to be spanned by that given set of vectors, and is denoted by $\mathrm{span}\{V_1, V_2, \cdots, V_m\}$.

- If the set $\{V_1, V_2, \cdots, V_m\}$ is linearly independent, then it is said to be a **basis** for the subspace and the number of elements in the basis, m, is the dimension of the subspace.

Basis vector

- Generally, any set of linearly independent vectors is called a basis set, and each vector in a basis set is referred to as a basis vector.

Natural or standard basis

- Any set of n linearly independent vectors

$$\left\{ e_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \cdots \quad e_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \right\}$$

form a natural basis for the space $\mathbb{R}^n$.

Orthogonal basis

- If the basis vectors, in addition to being linearly independent, are mutually orthogonal, that is, for

$$V = \{V_1, V_2, \cdots, V_m\},$$

$$\langle V_i, V_j \rangle = 0, \quad i \neq j, \qquad i, j \in \{1, \cdots, m\},$$

then the basis set is referred to as an orthogonal basis.

Orthonormal basis

- If the basis vectors, in addition to being orthogonal, have unit norms, that is, for

$$V = \{V_1, V_2, \cdots, V_m\},$$

$$\|V_i\| = 1, \qquad i = 1, \cdots, m,$$

then the basis set is referred to as an orthonormal basis.

Row space of a matrix

- Consider an $m \times n$ matrix A. The matrix has m rows with n elements each, so each row is an n-vector.
- The **row space** of A is a subspace of $\mathbb{R}^n$ which is spanned by the m rows.
- The dimension of the row space is known as the row rank.

Column space of a matrix

- Consider an $m \times n$ matrix A. The matrix has n columns with m elements each, so each column is an m-vector.
- The **column space** of A is a subspace of $\mathbb{R}^m$ which is spanned by the n columns.
- The dimension of the column space is known as the column rank.

Column null space of a matrix

- Consider an m×n matrix A. The subspace generated by some column vectors **U** such that

$$\mathbf{AU} = \mathbf{0}$$

is called the column null space of A.

- The dimension of the column null space of A is called the column nullity of A, and is given as $n - r$, where r is the rank of A.

- $\text{Number of columns of A} =$

$$\text{rank of A} + \text{column nullity of A}$$

Row null space of a matrix

- Consider an $\mathrm{m} \times n$ matrix A. The subspace generated by some row vectors **P** such that

$$\mathbf{PA} = \mathbf{0}$$

is called the row null space of A.

- The dimension of the row null space of A is called the row nullity of A, and is given as $\mathrm{m} - r$, where r is the rank of A.
- Number of rows of A $=$

$$\text{rank of A} + \text{row nullity of A}$$

Linear Transformation

Background

- A matrix is a rectangular array of similar objects or things.
- A matrix $A \in \mathbb{R}^{m \times n}$ has m rows and n columns, as represented by

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

- The ith row is the n-vector:

$$\boldsymbol{a}_{i:} = \begin{bmatrix} a_{i1} \\ a_{i2} \\ \vdots \\ a_{in} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^n, \qquad i = 1, 2, \cdots, m$$

- The jth column is the m-vector:

$$\boldsymbol{a}_{:j} = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix} \in \mathbb{R}^m, \qquad j = 1, 2, \cdots, n$$

- This matrix $A \in \mathbb{R}^{m \times n}$ can be written in terms of its rows and columns as follows:

$$A = [\boldsymbol{a}_{:1} \quad \boldsymbol{a}_{:2} \quad \cdots \quad \boldsymbol{a}_{:n}] = \begin{bmatrix} \boldsymbol{a}_{1:} \\ \boldsymbol{a}_{2:} \\ \vdots \\ \boldsymbol{a}_{m:} \end{bmatrix}$$

- The same matrix $A \in \mathbb{R}^{m \times n}$ can also be written as $[a_{ij}]^{mn}$, or simply $[a_{ij}]$ (when there is no ambiguity about the dimension).

Important operations and properties

Transposition:

- The transpose of matrix $A \in \mathbb{R}^{m\times n}$, which is defined as

$$A = [\boldsymbol{a}_{:1} \quad \boldsymbol{a}_{:2} \quad \cdots \quad \boldsymbol{a}_{:n}] = \begin{bmatrix} \boldsymbol{a}_{1:} \\ \boldsymbol{a}_{2:} \\ \vdots \\ \boldsymbol{a}_{m:} \end{bmatrix},$$

is

$$A^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{a}_{:1}^{\mathrm{T}} \\ \boldsymbol{a}_{:2}^{\mathrm{T}} \\ \vdots \\ \boldsymbol{a}_{:n}^{\mathrm{T}} \end{bmatrix} = [\boldsymbol{a}_{1:}^{\mathrm{T}} \quad \boldsymbol{a}_{2:}^{\mathrm{T}} \quad \cdots \quad \boldsymbol{a}_{m:}^{\mathrm{T}}].$$

- **Note that $(ABC)^{\mathrm{T}} = C^{\mathrm{T}}B^{\mathrm{T}}A^{\mathrm{T}}$.**

Addition/subtraction:

- Consider matrices $A, B \in \mathbb{R}^{m \times n}$. Then

$$A \pm B = \left[a_{ij} \pm b_{ij}\right]$$

Scalar-matrix multiplication:

- Consider matrix $A \in \mathbb{R}^{m \times n}$ and scalar $\beta \in \mathbb{R}$. Then

$$\beta A = A\beta = \left[\beta a_{ij}\right]$$

Vector-matrix multiplication:

- Consider $A \in \mathbb{R}^{m \times n}$, $u \in \mathbb{R}^{n}$, and $v \in \mathbb{R}^{m}$. Then

$$Au = \sum_{j=1}^{n} \boldsymbol{a}_{:j}\, u_j$$

$$v^{\mathrm{T}} A = \sum_{i=1}^{m} v_i \boldsymbol{a}_{i:}$$

Matrix-matrix multiplication:

- Multiplication of matrix $A \in \mathbb{R}^{m \times n}$ by matrix $B \in \mathbb{R}^{n \times p}$ results in $C \in \mathbb{R}^{m \times p}$, i.e., $C = AB$. Then

$$c_{ij} = \langle \boldsymbol{a}_{i:}, \boldsymbol{b}_{:j} \rangle \qquad i = 1, \cdots, m; j = 1, \cdots, p$$

or

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

or

$$C = AB = \begin{bmatrix} \boldsymbol{a}_{1:} \\ \boldsymbol{a}_{2:} \\ \vdots \\ \boldsymbol{a}_{m:} \end{bmatrix} [\boldsymbol{b}_{:1} \quad \boldsymbol{b}_{:2} \quad \cdots \quad \boldsymbol{b}_{:p}]$$

Linear transformation by a matrix:

- The product of a matrix $\mathrm{A} \in \mathbb{R}^{m\times n}$ and a vector $u \in \mathbb{R}^n$, that is,

  $$\mathrm{y} = \mathrm{A}u$$

  produces a new vector $y \in \mathbb{R}^m$. This operation is known as a linear transformation from $\mathbb{R}^n$ into $\mathbb{R}^m$, which is denoted by $\{A: \mathbb{R}^n \to \mathbb{R}^m\}$.

Eigenvalues and Eigenvectors

Eigenvalues and eigenvectors of a square matrix:

- An extremely important linear transformation in linear algebra is

  $$Ax = \lambda x$$

  for $A \in \mathbb{R}^{n \times n}$ and $x \in \mathbb{R}^{n \times 1}$.
- Notice that, unlike the general linear transformation $Ax = y$, the **direction of the vector x does not change under this 'eigen' transformation**.
- x and λ form an eigenpair; x is called the eigenvalue, while λ is called the eigenvector.

- Since a square matrix $A \in \mathbb{R}^{n\times n}$ has n eigenvalues and n corresponding eigenvectors, then the eigen transformation becomes

$$\mathrm{A}\mathrm{x}_1 = \lambda_1 \mathrm{x}_1$$
$$\mathrm{A}\mathrm{x}_2 = \lambda_2 \mathrm{x}_2$$
$$\vdots$$
$$\mathrm{A}\mathrm{x}_n = \lambda_n \mathrm{x}_\mathrm{n}$$

or

$$\mathrm{AX} = \mathrm{X}\Lambda$$

- In the equation

$$AX = X\Lambda$$

X is called the matrix of eigenvectors (which are right eigenvectors), and Λ is the diagonal matrix of corresponding eigenvalues.

- Therefore,

$$A = X\Lambda X^{-1}$$

is called the eigenvalue or spectra decomposition of A.

- Further, A can be diagonalized using

$$\Lambda = X^{-1}AX$$

- Matrix A and matrix Λ are said to be similar, and X is termed the similarity transformation matrix.
- In general, suppose A, P and P^{-1} are all $n \times n$ matrices. Then, the matrix

$$\widetilde{A} = P^{-1}AP$$

- is said to be similar to A, and the operation yielding $\widetilde{A}$ is known as a similarity transformation.

- Another form of the eigen transformation is

$$\mathrm{y}\mathrm{A} = \lambda \mathrm{y}$$

for $\mathrm{A} \in \mathbb{R}^{n \times n}$ and $\mathrm{y} \in \mathbb{R}^{1 \times n}$.

where y is called the left eigenvector of A.

- The alternative form of the spectra decomposition of A is given as

$$\mathrm{A} = \mathrm{Y}^{-1} \Lambda \mathrm{Y}$$

where Y is the matrix of the left eigenvectors.

- Note that the left and right eigenvectors corresponding to different eigenvalues are orthogonal. That is, if λ_i and λ_j are two different eigenvalues, then

$$y_i \mathrm{x_j} = 0$$

- But the product of left and right eigenvectors corresponding to the same eigenvalues equals to a constant value. That is,

$$y_i \mathrm{x_i} = \mathrm{c}$$

or (by normalizing)

$$y_i \mathrm{x_i} = 1$$

Eigenvalue sensitivity:

- It is essential at times to know how the eigenvalues are influenced by the elements of the matrix.
- Recall again that

$$\mathrm{Ax}_i = \lambda_i \mathrm{x}_i$$

- Differentiate the equation with respect to some element a_{kj} of A:

$$\frac{\partial A}{\partial a_{kj}} \mathrm{x}_i + A \frac{\partial \mathrm{x}_i}{\partial a_{kj}} = \frac{\partial \lambda_i}{\partial a_{kj}} \mathrm{x}_i + \lambda_i \frac{\partial \mathrm{x}_i}{\partial a_{kj}}$$

- Multiply both sides by the left eigenvector y_i:

$$y_i \frac{\partial A}{\partial a_{kj}} \mathrm{x}_i + y_i A \frac{\partial \mathrm{x}_i}{\partial a_{kj}} = y_i \frac{\partial \lambda_i}{\partial a_{kj}} \mathrm{x}_i + y_i \lambda_i \frac{\partial \mathrm{x}_i}{\partial a_{kj}}$$

or

$$y_i \frac{\partial A}{\partial a_{kj}} \mathrm{x}_i = y_i \frac{\partial \lambda_i}{\partial a_{kj}} \mathrm{x}_i + y_i (\lambda_i I - A) \frac{\partial \mathrm{x}_i}{\partial a_{kj}}$$

- Thus,

$$\frac{\partial \lambda_i}{\partial a_{kj}} = y_{ik} \mathrm{x}_{ji}$$

Calculations of eigenvalues and eigenvectors:

- Recall the matrix equation

$$\mathrm{A}\mathrm{x}_i = \lambda_i \mathrm{x}_i$$

where λ_{i} is the i^{th} eigenvalue and x_i the associated eigenvector.

$$(\lambda_i \mathrm{I} - \mathrm{A})\mathrm{x}_i = 0$$

$$\mathrm{x}_i = \frac{\mathrm{adj}(\lambda_{\mathrm{i}}\mathrm{I}-\mathrm{A})}{\det(\lambda_{\mathrm{i}}\mathrm{I}-\mathrm{A})} 0$$

- The nontrivial solution of x_i is the one that satisfies $\det(\lambda_i I - A) = 0$.
- Thus,

Eigenvalues: $\det(\lambda_i I - A)$

Eigenvectors: $(\lambda_i I - A)x_i = 0$

Case I: Distinct eigenvalues

- For distinct eigenvalues, the eigenvectors are distinct and can be determined from the simultaneous equations

$$(\lambda_i \mathrm{I} - \mathrm{A})\mathrm{x}_i = 0$$

or from any nonzero column of the matrix

$$\mathrm{adj}(\lambda_\mathrm{i} \mathrm{I} - \mathrm{A})$$

Case II: Non-distinct (or repeated) eigenvalues

- For repeated eigenvalues, the associated eigenvectors may be fully linearly independent or they may not.
- The number of linearly independent eigenvectors associated with a given eigenvalue λ_i repeated r_i times is expressed as

$$p_i = n - rank(\lambda_i \mathrm{I} - \mathrm{A}) \qquad\qquad 1 \le p_i \le r_i$$

1) $p_i = r_i$: r_i linearly independent eigenvectors

2) $p_i = 1$: only one linearly independent eigenvector

3) $1 < p_i < r_i$

#Eigenvalues and eigenvectors

```
import numpy as np

 # Create a matrix with multiple eigenvalues
A = np.array([[2, 0, 0], [0, 2, 0], [0, 3, 1]])

# Calculate eigenvalues and eigenvectors of the matrix
eigVals, eigVecs = np.linalg.eig(A)

#The eigenvalues are 2,2,1
#find the number of distinct eigenvectors for the eigenvalue of 2
rk=np.linalg.matrix_rank((2*np.array([[1, 0,0], [0, 1, 0],[0, 0, 1]])-A))
p1=3-rk
p1
```

#Eigenvalues and eigenvectors

```
import numpy as np

 # Create a matrix
# A = np.array([[2, 1], [0, 2]])  #Multiple eigenvalues; distinct eigenvectors
A = np.array([[2, 1], [1, 2]])    # Distinct eigenvalues; distinct eigenvectors

# Calculate eigenvalues and eigenvectors of the matrix
eigVals, eigVecs = np.linalg.eig(A)

# Perform the diagonalization of the matrix
X_inv=np.linalg.inv(eigVecs)
X=eigVecs
Diag_A=np.dot((np.dot(X_inv,A)),X)

print("Matrix A:"), print(A)
print("\nEigenvalues:"), print(eigVals)
print("\nEigenvectors:"), print(eigVecs)
print("\nDiagonalized form of matrix A:"), print(Diag_A)
```

END OF PART I

THANK YOU