COMP 3206: Machine Learning
Linear Algebra Background

Srinandan Dasmahapatra

2017/8
A vector space is a set with additional structure
From sets to vectors

- A vector space is a set with additional structure
- The structure allows you to

  to get other elements of the set
From sets to vectors

- A vector space is a set with additional structure
- The structure allows you to
  - multiply elements by scalars and

  to get other elements of the set
A vector space is a set with additional structure. The structure allows you to:
- multiply elements by scalars and
- add elements together

to get other elements of the set.
From sets to vectors

- A vector space is a set with additional structure
- The structure allows you to
  - multiply elements by scalars and
  - add elements together
  to get other elements of the set
- Impose transformation rules on all elements – linear transformations
From sets to vectors

- A vector space is a set with additional structure
- The structure allows you to
  - multiply elements by scalars and
  - add elements together
    to get other elements of the set
- Impose transformation rules on all elements – linear transformations
- Discover hidden parts/patterns in collections
Information storage matrix is $A$ may be viewed as map from space of *users* to space of *movies*
Matrix representation of associations: storage and access

- Information storage matrix is $A$ may be viewed as map from space of users to space of movies.

\[
A : U \rightarrow V \\
A_{uv} = \begin{cases} 
1 & \text{if } (u, v) \text{ connected} \\
0 & \text{otherwise}
\end{cases}
\]
Matrix representation of associations: storage and access

- Information storage matrix is $A$ may be viewed as map from space of **users** to space of **movies**

$$A : U \rightarrow V$$

$$A_{uv} = \begin{cases} 
1 & \text{if } (u, v) \text{ connected} \\
0 & \text{otherwise} 
\end{cases}$$

- Retrieval of information is by matrix-vector operations
Matrix representation of associations: storage and access

Data as matrix

\[
A = \begin{pmatrix}
0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 0
\end{pmatrix}
\]
Matrix representation of associations: storage and access

Data as matrix

\[ A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \]

Information retrieval by matrix-vector ops
Matrix representation of associations: storage and access

- Data as matrix
  
  \[
  \mathbf{A} = \begin{pmatrix}
  0 & 1 & 0 & 0 & 1 \\
  0 & 0 & 1 & 0 & 1 \\
  1 & 0 & 0 & 1 & 0
  \end{pmatrix}
  \]

- Information retrieval by matrix-vector ops

- Given: preferences of 3 subscribers to Netflix, predict: what movies would this new user rent?
Represent elements of sets as vectors

\[ \mathbf{\hat{u}}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{u}}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{u}}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \]

\[ \mathbf{\hat{v}}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{v}}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{v}}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{v}}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{\hat{v}}_5 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \]

- Retrieval: movie preferences of $k$-th user $\mathbf{\hat{u}}_k$ obtained by performing $A^T \mathbf{\hat{e}}_k$
Retrieval of information by matrix-vector multiplication

For user 2,

\[ \mathbf{A}^T \hat{\mathbf{u}}_2 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \]
Retrieval of information by matrix-vector multiplication

For user 2,

\[ \mathbf{A}^T \hat{\mathbf{u}}_2 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \]

\[ \mathbf{A}^T \hat{\mathbf{u}}_2 = \hat{\mathbf{v}}_3 + \hat{\mathbf{v}}_5 \]
Information Retrieval, Lexical Semantics: meaning of word determined by company it keeps

- $\mathcal{D}$, a collection of documents $d_j \in \mathcal{D}$, $j = 1, \ldots, n = |\mathcal{D}|$. 

$W$, the vocabulary, i.e., all the words $w_i \in W$ contained in $\mathcal{D}$, $i = 1, \ldots, m = |W|$. Construct $(m \times n)$ matrix $T$ with entries $(T)_{ij} = t_{ij}$, where $t_{ij}$ counts the number of times word $w_i$ appears in document $d_j$: 

$$ T = \begin{pmatrix} 
T_{11} & \cdots & T_{1n} \\
\vdots & \ddots & \vdots \\
T_{m1} & \cdots & T_{mn} 
\end{pmatrix} $$

Column view: document retrieval (library catalogue)
Row view: lexical semantics (distributional similarity): if 2 words $a, b$ appear in same documents, they are similar: $w_a$ close to $w_b$. 

Information Retrieval, Lexical Semantics: meaning of word determined by company it keeps

- $\mathcal{D}$, a collection of documents $d_j \in \mathcal{D}, j = 1, \ldots, n = |\mathcal{D}|$.
- $\mathcal{W}$, the vocabulary, i.e. all the words $w_i \in \mathcal{W}$ contained in $\mathcal{D}$, $i = 1, \ldots, m = |\mathcal{W}|$. 

Construct $(m \times n)$ matrix $T$ with entries $(T)_{ij} = t_{ij}$, where $t_{ij}$ counts the number of times word $w_i$ appears in document $d_j$:

$$T = \begin{pmatrix} \vdots & \vdots & \vdots \\ d_1 & d_n & \vdots \\ \vdots & \vdots & \vdots \end{pmatrix} = \begin{pmatrix} 0 & 0 & \vdots \\ \vdots & \vdots & \vdots \\ 0 & 0 & \vdots \end{pmatrix}$$

Column view { document retrieval (library catalogue)
Row view { lexical semantics (distributional similarity): if 2 words $a$; $b$ appear in same documents, they are similar: $w_a$ close to $w_b$.}
Information Retrieval, Lexical Semantics:
meaning of word determined by company it keeps

- \( \mathcal{D} \), a collection of documents \( d_j \in \mathcal{D}, j = 1, \ldots, n = |\mathcal{D}|. \)
- \( \mathcal{W} \), the vocabulary, i.e. all the words \( w_i \in \mathcal{W} \) contained in \( \mathcal{D} \), \( i = 1, \ldots, m = |\mathcal{W}| \)
- Construct \((m \times n)\) matrix \( T \) with entries \((T)_{ij} = t_{ij}\), where \( t_{ij} \) counts the number of times word \( w_i \) appears in document \( d_j \):
Information Retrieval, Lexical Semantics: meaning of word determined by company it keeps

- $\mathcal{D}$, a collection of documents $d_j \in \mathcal{D}$, $j = 1, \ldots, n = |\mathcal{D}|$.
- $\mathcal{W}$, the vocabulary, i.e. all the words $w_i \in \mathcal{W}$ contained in $\mathcal{D}$, $i = 1, \ldots, m = |\mathcal{W}|$.
- Construct $(m \times n)$ matrix $\mathcal{T}$ with entries $(\mathcal{T})_{ij} = t_{ij}$, where $t_{ij}$ counts the number of times word $w_i$ appears in document $d_j$.
Information Retrieval, Lexical Semantics: meaning of word determined by company it keeps

- $\mathcal{D}$, a collection of documents $d_j \in \mathcal{D}$, $j = 1, \ldots, n = |\mathcal{D}|$.
- $\mathcal{W}$, the vocabulary, i.e. all the words $w_i \in \mathcal{W}$ contained in $\mathcal{D}$, $i = 1, \ldots, m = |\mathcal{W}|$
- Construct $(m \times n)$ matrix $\mathcal{T}$ with entries $(\mathcal{T})_{ij} = t_{ij}$, where $t_{ij}$ counts the number of times word $w_i$ appears in document $d_j$:

$$
\mathcal{T} = \begin{pmatrix}
\vdots & \vdots & \vdots \\
\mathbf{d}_1 & \cdots & \mathbf{d}_n \\
\vdots & \vdots & \vdots
\end{pmatrix} = 
\begin{pmatrix}
\vdots & \mathbf{w}_1 & \vdots \\
\vdots & \vdots & \vdots \\
\vdots & \mathbf{w}_m & \vdots
\end{pmatrix}
$$

- Column view – document retrieval (library catalogue)
Information Retrieval, Lexical Semantics: meaning of word determined by company it keeps

- $\mathcal{D}$, a collection of documents $d_j \in \mathcal{D}$, $j = 1, \ldots, n = |\mathcal{D}|$.
- $\mathcal{W}$, the vocabulary, i.e. all the words $w_i \in \mathcal{W}$ contained in $\mathcal{D}$, $i = 1, \ldots, m = |\mathcal{W}|$
- Construct $(m \times n)$ matrix $\mathcal{T}$ with entries $(\mathcal{T})_{ij} = t_{ij}$, where $t_{ij}$ counts the number of times word $w_i$ appears in document $d_j$:

$$
\mathcal{T} = \begin{pmatrix}
\mathbf{d}_1 & \cdots & \mathbf{d}_n
\end{pmatrix} = \begin{pmatrix}
\ldots & \mathbf{w}_1 & \ldots \\
\ldots & \vdots & \ldots \\
\ldots & \mathbf{w}_m & \ldots
\end{pmatrix}
$$

- Column view – document retrieval (library catalogue)
- Row view – lexical semantics (distributional similarity): if 2 words $a, b$ appear in same documents, they are similar: $\mathbf{w}_a$ close to $\mathbf{w}_b$. 

---

<table>
<thead>
<tr>
<th>Document</th>
<th>Document</th>
<th>Document</th>
<th>Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathbf{d}_1$</td>
<td>$\mathbf{d}_2$</td>
<td>$\mathbf{d}_3$</td>
<td>$\mathbf{d}_4$</td>
</tr>
</tbody>
</table>

---

- Use of $t_{ij}$ to represent the number of times $w_i$ appears in $d_j$.
Representing images as vectors

\[
\mathbf{x} = (x_{11}, x_{12}, \ldots, x_{LL})^T = (x(1), x(2), \ldots, x(D))^T.
\]
Matrices

You should all know the following:

- Matrix notation
- Matrix transpose
- Scalar multiplication
- Matrix addition & multiplication
- Matrix inverse
- System of linear equations in matrix form
- Matrix determinant
Reminder: Solving Linear Equations – Geometrical Picture

- Solve set of equations:

\[
\begin{pmatrix}
  a & b \\
  c & d
\end{pmatrix}
\begin{pmatrix}
  x \\
  y
\end{pmatrix}
=
\begin{pmatrix}
  r \\
  s
\end{pmatrix}
\]

- Geometrically viewed as intersection of linear linear combination of vectors:

\[
\begin{align*}
\text{rows of matrix} & : \\
ax + by & = r \\
cx + dy & = s \\
\end{align*}
\leftrightarrow
\begin{align*}
\text{columns of matrix} & : \\
x\begin{pmatrix}
a \\
c
\end{pmatrix} + y\begin{pmatrix}
b \\
d
\end{pmatrix} & = \begin{pmatrix}
r \\
s
\end{pmatrix}
\end{align*}
\]
The Geometrical Picture: An example

- Solve set of equations:

\[
\begin{pmatrix}
-2 & 1 \\
-1 & 3
\end{pmatrix}
\begin{pmatrix}
x \\
y
\end{pmatrix}
=
\begin{pmatrix}
4 \\
-2
\end{pmatrix}
\]

- Solution of \( y - 2x = 4, 3y - x = -2 \), is \((x, y) = (-2.8, -1.6)\).
Fundamental operations on vectors – multiply by scalars and perform addition

Multiply red vectors by numbers (elements of a field) and add vectors together
Fundamental operations on vectors – multiply by scalars and perform addition

Multiply red vectors by numbers (elements of a field) and add vectors together

For **linear regression**: find linear combination of columns of design matrix to get vector closest to output
Examples of vector spaces

- Add vectors $(1.0, -2.0) + (3.0, 4.0) = (4.0, 2.0)$, where the entries are in this case real.
Examples of vector spaces

- Add vectors $(1.0, -2.0) + (3.0, 4.0) = (4.0, 2.0)$, where the entries are in this case real.
- Multiply vectors by numbers (scalars) $3.2(1.0, -3.0) = (3.2, -9.6)$
Examples of vector spaces

- Add vectors $(1.0, -2.0) + (3.0, 4.0) = (4.0, 2.0)$, where the entries are in this case real.
- Multiply vectors by numbers (scalars) $3.2(1.0, -3.0) = (3.2, -9.6)$
- For any field $\mathbb{F}$ (such as the reals $\mathbb{R}$ or complex numbers $\mathbb{C}$), $\mathbb{F}$-valued $n$-tuples

$$\mathbb{F}^n = \{(a_1, \ldots, a_n) | a_i \in \mathbb{F}, i = 1, \ldots, n\}$$

form a vector space; $\mathbb{R}$-valued 3-tuples such as $\mathbf{v}_1 = (-1.2, 2.0, 5.5)$ locate points in 3D. Written as rows or columns $\mathbf{v}_1^T = \begin{pmatrix} -1.2 \\ 2.0 \\ 5.5 \end{pmatrix}$. 
Matrices form a vector space: you can multiply \( n \times m \) matrices \( A \) over \( \mathbb{F} \) with entries \( a_{ij} \in \mathbb{F}, \ i = 1, \ldots, n, \ j = 1, \ldots, m \) by scalars and add any two such matrices together:

\[
3 \begin{pmatrix} -2 & 1 \\ -1 & 4 \end{pmatrix} - 2 \begin{pmatrix} 2 & 2 \\ -1 & 6 \end{pmatrix} = \begin{pmatrix} -10 & -1 \\ -1 & 0 \end{pmatrix}.
\]
Functions constitute vector spaces

- \( \mathbb{F}[x] \), the space of polynomials \( \sum_m a_m x^m \), where \( a_m \in \mathbb{F} \) forms a vector space:

\[
(a_0 + a_1 x + a_2 x^2) + (b_0 + b_1 x) = (a_0 + b_0) + (a_1 + b_1)x + a_2 x^2
\]

\[
=: (c_0 + c_1 x + c_2 x^2).
\]

Monomials as basis elements:

\[
(a_0, a_1, a_2) + (b_0, b_1, 0) = (a_0 + b_0, a_1 + b_1, a_2)
\]
Functions constitute vector spaces

- \( \mathbb{F}[x] \), the space of polynomials \( \sum_m a_m x^m \), where \( a_m \in \mathbb{F} \) forms a vector space:

\[
(a_0 + a_1 x + a_2 x^2) + (b_0 + b_1 x) = (a_0 + b_0) + (a_1 + b_1) x + a_2 x^2
\]

\[
=: (c_0 + c_1 x + c_2 x^2).
\]

Monomials as basis elements:

\[
(a_0, a_1, a_2) + (b_0, b_1, 0) = (a_0 + b_0, a_1 + b_1, a_2)
\]

- Similarly, the set \( \mathbb{F}[x_1, \ldots, x_k] \) of polynomials in \( k \) variables forms a vector space.
Functions constitute vector spaces

- $\mathbb{F}[x]$, the space of polynomials $\sum_m a_m x^m$, where $a_m \in \mathbb{F}$ forms a vector space:

$$
(a_0 + a_1 x + a_2 x^2) + (b_0 + b_1 x) = (a_0 + b_0) + (a_1 + b_1) x + a_2 x^2
$$

$$
=: (c_0 + c_1 x + c_2 x^2).
$$

Monomials as basis elements:

$$
(a_0, a_1, a_2) + (b_0, b_1, 0) = (a_0 + b_0, a_1 + b_1, a_2)
$$

Similarly, the set $\mathbb{F}[x_1, \ldots, x_k]$ of polynomials in $k$ variables forms a vector space.

- Set of functions of the form $\sum_{|n|<N} a_n e^{in\theta}$ (Fourier series).
- Extension – replace sums (where the summation index is from a discrete set) by integrals (where the index being summed over is now continuous)
Formal definition of vector space

A vector space $V$ over a field $\mathbb{F}$ is a collection of objects (vectors) upon which two operations can be performed – addition amongst the vectors, and multiplication by elements of the field $\mathbb{F}$ (scalars). Upon addition of any two vectors $v_1 \in V, v_2 \in V$, the resulting vector $v_1 + v_2$ must also belong to $V$ (closure). The binary product for $a \in \mathbb{F}$ (scalar) and $v \in V$ (vector)

$$m : \mathbb{F} \times V \rightarrow V$$

$$m(a, v) \mapsto av$$

is defined, and satisfies
Formal definition of vector space

A vector space \( V \) over a field \( \mathbb{F} \) is a collection of objects (vectors) upon which two operations can be performed – addition amongst the vectors, and multiplication by elements of the field \( \mathbb{F} \) (scalars). Upon addition of any two vectors \( \mathbf{v}_1, \mathbf{v}_2 \in V \), the resulting vector \( \mathbf{v}_1 + \mathbf{v}_2 \) must also belong to \( V \) (closure). The binary product for \( a \in \mathbb{F} \) (scalar) and \( \mathbf{v} \in V \) (vector)

\[
m : \mathbb{F} \times V \rightarrow V
\]

\[
m(a, \mathbf{v}) \mapsto a\mathbf{v}
\]

is defined, and satisfies

- \( 1\mathbf{v} = \mathbf{v}, \ 1 \in \mathbb{F}, \ \mathbf{v} \in V \)
A vector space $V$ over a field $\mathbb{F}$ is a collection of objects (vectors) upon which two operations can be performed – addition amongst the vectors, and multiplication by elements of the field $\mathbb{F}$ (scalars). Upon addition of any two vectors $\mathbf{v}_1 \in V, \mathbf{v}_2 \in V$, the resulting vector $\mathbf{v}_1 + \mathbf{v}_2$ must also belong to $V$ (closure). The binary product for $a \in \mathbb{F}$ (scalar) and $\mathbf{v} \in V$ (vector)

$$m : \mathbb{F} \times V \rightarrow V$$

$$m(a, \mathbf{v}) \mapsto a\mathbf{v}$$

is defined, and satisfies

- $1\mathbf{v} = \mathbf{v}$, $1 \in \mathbb{F}$, $\mathbf{v} \in V$
- $(ab)\mathbf{v} = a(b\mathbf{v})$, $a, b \in \mathbb{F}$, $\mathbf{v} \in V$
Formal definition of vector space

- A vector space \( V \) over a field \( \mathbb{F} \) is a collection of objects (vectors) upon which two operations can be performed – addition amongst the vectors, and multiplication by elements of the field \( \mathbb{F} \) (scalars). Upon addition of any two vectors \( \mathbf{v}_1 \in V, \mathbf{v}_2 \in V \), the resulting vector \( \mathbf{v}_1 + \mathbf{v}_2 \) must also belong to \( V \) (closure). The binary product for \( a \in \mathbb{F} \) (scalar) and \( \mathbf{v} \in V \) (vector)

\[
m : \mathbb{F} \times V \rightarrow V
\]
\[
m(a, \mathbf{v}) \mapsto a\mathbf{v}
\]

is defined, and satisfies

- \( 1\mathbf{v} = \mathbf{v}, 1 \in \mathbb{F}, \mathbf{v} \in V \)
- \( (ab)\mathbf{v} = a(b\mathbf{v}), a, b \in \mathbb{F}, \mathbf{v} \in V \)
- \( (a + b)\mathbf{v} = a\mathbf{v} + b\mathbf{v} \) and \( a(\mathbf{v} + \mathbf{w}) = a\mathbf{v} + aw, a, b \in \mathbb{F}, \mathbf{v}, \mathbf{w} \in V \)
On notation:

$A \times B$ is the Cartesian product of the sets $A$ and $B$. This means that, for example, if $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2\}$, then

$$A \times B = \{(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2), (a_3, b_1), (a_3, b_2)\}.$$ 

In the definition of $m$, the symbol $\rightarrow$ refers to the map between the sets, while $\mapsto$ takes a particular pair from $\mathbb{F} \times V$ to produce an output vector from $V$. 
Reminder: Linear combination and dependence

Linear combination of vectors:
\[ \mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i, \quad \alpha_i \in \mathbb{F}, \mathbf{v}_i \in V. \]

- The vectors in the figure are linear combinations of \( \mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \) and \( \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \). They are in the span of \( \{ \mathbf{e}_1, \mathbf{e}_2 \} \).
- \( \mathbf{v} = a_1 \mathbf{e}_1 + a_2 \mathbf{e}_2 = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \) can be zero iff \( a_1 = 0 = a_2 \).
A set of vectors $v_1, v_2, \ldots, v_n$ are called **linearly independent** if none of them can be represented as a linear combination of the others:

$$v_k \neq \sum_{i \neq k} c_i v_i, \text{ for any } c_i \in F$$
Reminder: Linear independence & Basis

- A set of vectors $v_1, v_2, \ldots, v_n$ are called **linearly independent** if none of them can be represented as a linear combination of the others:

  $$v_k \neq \sum_{i \neq k} c_i v_i, \text{ for any } c_i \in F$$

- Equivalently, condition for a set of vectors $\{v_i\}_i$ to be linearly independent:

  $$\text{If } \sum_{i=1}^{n} \alpha_i v_i = 0, \text{ then } \alpha_i = 0 \text{ for all } i$$
Reminder: Linear independence & Basis

- A set of vectors \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n \) are called **linearly independent** if none of them can be represented as a linear combination of the others:

  \[
  \mathbf{v}_k \neq \sum_{i \neq k} c_i \mathbf{v}_i, \text{ for any } c_i \in \mathbb{F}
  \]

- Equivalently, condition for a set of vectors \( \{\mathbf{v}_i\}_i \) to be linearly independent:

  \[
  \text{If } \sum_{i=1}^{n} \alpha_i \mathbf{v}_i = 0, \text{ then } \alpha_i = 0 \text{ for all } i
  \]

- A **basis** for \( V \) is a set \( B \subset V \) which is both spanning and independent. A finite dimensional vector space has a finite basis, and its dimension \( \text{dim } V \) is the number of elements in \( B \).
Dot Products, Orthogonality and Norms

- We can associate, with two vectors \( \mathbf{v} \) and \( \mathbf{w} \) an element of \( \mathbb{F} \) called their **scalar** (or **dot**) product:

\[
\text{dot} : V \times V \rightarrow \mathbb{F} \\
\text{dot}(\mathbf{v}, \mathbf{w}) = \mathbf{v} \cdot \mathbf{w} = \mathbf{v}^T \mathbf{w} \mapsto a
\]

- Two vectors \( \mathbf{v}_1 \) and \( \mathbf{v}_2 \) are called **orthogonal** if their dot product is zero, i.e. \( \mathbf{v}_1 \cdot \mathbf{v}_2 = 0 \). If \( k \) vectors \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k \) are mutually orthogonal, i.e. \( \mathbf{v}_i \cdot \mathbf{v}_j = 0 \) for \( i \neq j \), they are called an **orthogonal set**.

- Euclidean norm: for \( \mathbf{v} \in V \), \( \dim V = N \),

\[
\|\mathbf{v}\|_2 := \sqrt{\mathbf{v}^T \mathbf{v}} = \sqrt{v_1^2 + v_2^2 + \ldots + v_N^2} \]

If all vectors are of unit length \( \|\mathbf{v}_i\| = 1 \), the set is called an **orthonormal** set.
Dot Products, Orthogonality and Norms

- We can associate, with two vectors \( \mathbf{v} \) and \( \mathbf{w} \) an element of \( \mathbb{F} \) called their **scalar** (or **dot**) **product**:

\[
\text{dot} : V \times V \rightarrow \mathbb{F}
\]

\[
\text{dot}(\mathbf{v}, \mathbf{w}) = \mathbf{v} \cdot \mathbf{w} = \mathbf{v}^T \mathbf{w} \rightarrow a
\]

- Two vectors \( \mathbf{v}_1 \) and \( \mathbf{v}_2 \) are called *orthogonal* if their dot product is zero, i.e. \( \mathbf{v}_1 \cdot \mathbf{v}_2 = 0 \). If \( k \) vectors \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k \) are mutually orthogonal, i.e. \( \mathbf{v}_i \cdot \mathbf{v}_j = 0 \) for \( i \neq j \), they are called an **orthogonal set**.

- Euclidean norm: for \( \mathbf{v} \in V \), \( \text{dim } V = N \),

\[
\|\mathbf{v}\|_2 := \sqrt{\mathbf{v}^T \mathbf{v}} = \sqrt{v_1^2 + v_2^2 + \cdots + v_N^2} = |\mathbf{v}|
\]

- If all vectors are of unit length \( |\mathbf{v}_i| = 1 \), the set is called **orthonormal**.
Using dot products to introduce projections

- **Project** a vector \( \mathbf{y} \) on a direction given by a vector \( \mathbf{u} \)

![Diagram of projection](image)

- The **projection** is given by the value \( w \) (length \( \overline{0w} \)). Note, \( w \) could be negative if \( \alpha \) is bigger than 90°. From the figure, we see that

\[
w = |\mathbf{y}| \cos \alpha = \frac{\mathbf{y} \cdot \mathbf{u}}{|\mathbf{u}|},
\]

because the **dot** product is \( \mathbf{y} \cdot \mathbf{u} = |\mathbf{u}||\mathbf{y}| \cos \alpha \).
Expanding a vector in a set of orthogonal vectors – an example

Let $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. We would like to expand $v = \begin{pmatrix} -5 \\ 3 \end{pmatrix}$ as a linear combination of the set $\{e_i\}$, ie. we would like to find numbers $\alpha_1, \alpha_2$ such that

$$v = \sum_{i=1}^{n} \alpha_i e_i$$
Expanding a vector in a set of orthogonal vectors — an example

Let \( \mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \). We would like to expand \( \mathbf{v} = \begin{pmatrix} -5 \\ 3 \end{pmatrix} \) as a linear combination of the set \( \{ \mathbf{e}_i \} \), i.e. we would like to find numbers \( \alpha_1, \alpha_2 \) such that

\[
\mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{e}_i
\]  

(1)

Solution: Multiply \( \mathbf{v} \) by \( \mathbf{e}_j \) and use the orthogonality (\( \mathbf{e}_1 \cdot \mathbf{e}_2 = 0 \)):

\[
\mathbf{e}_1 \cdot \begin{pmatrix} -5 \\ 3 \end{pmatrix} = -5, \quad \mathbf{e}_2 \cdot \begin{pmatrix} -5 \\ 3 \end{pmatrix} = 3.
\]

\[
\begin{pmatrix} -5 \\ 3 \end{pmatrix} = (-5) \begin{pmatrix} 1 \\ 0 \end{pmatrix} + (3) \begin{pmatrix} 0 \\ 1 \end{pmatrix}
\]
Expanding a vector in a set of orthogonal vectors

Suppose, $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ are an orthogonal set of $n \times 1$ column vectors and $\mathbf{v}$ an arbitrary $n \times 1$ column vector. We would like to expand $\mathbf{v}$ as a linear combination of the set $\{\mathbf{v}_i\}$, ie. we would like to find numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$ such that

$$\mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i.$$  \hspace{1cm} (2)

Solution: Multiply by $\mathbf{v}_j$ and use the orthogonality to get

$$\mathbf{v}_j \cdot \mathbf{v} = \alpha_1 \mathbf{v}_j \cdot \mathbf{v}_1 + \cdots + \alpha_j \mathbf{v}_j \cdot \mathbf{v}_j + \cdots \alpha_n \mathbf{v}_j \cdot \mathbf{v}_n$$
Expanding a vector in a set of orthogonal vectors

Suppose, $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ are an orthogonal set of $n \times 1$ column vectors and $\mathbf{v}$ an arbitrary $n \times 1$ column vector. We would like to expand $\mathbf{v}$ as a linear combination of the set $\{\mathbf{v}_i\}$, i.e. we would like to find numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$ such that

$$\mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i.$$  \hspace{1cm} (2)

Solution: Multiply by $\mathbf{v}_j$ and use the orthogonality to get

$$\mathbf{v}_j \cdot \mathbf{v} = \alpha_1 \mathbf{v}_j \cdot \mathbf{v}_1 + \cdots + \alpha_j \mathbf{v}_j \cdot \mathbf{v}_j + \cdots + \alpha_n \mathbf{v}_j \cdot \mathbf{v}_n$$

$$= \alpha_j \mathbf{v}_j \cdot \mathbf{v}_j$$

Hence

$$\alpha_j = \frac{\mathbf{v}_j \cdot \mathbf{v}}{\mathbf{v}_j \cdot \mathbf{v}_j} = \frac{\mathbf{v}_j \cdot \mathbf{v}}{|\mathbf{v}_j|^2}$$ \hspace{1cm} (3)
Expanding a vector in a set of orthogonal vectors

Suppose, \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n \) are an orthogonal set of \( n \times 1 \) column vectors and \( \mathbf{v} \) an arbitrary \( n \times 1 \) column vector. We would like to expand \( \mathbf{v} \) as a \textit{linear combination} of the set \( \{ \mathbf{v}_i \} \), ie. we would like to find numbers \( \alpha_1, \alpha_2, \ldots, \alpha_n \) such that

\[
\mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i.
\]

(2)

Solution: Multiply by \( \mathbf{v}_j \) and use the orthogonality to get

\[
\mathbf{v}_j \cdot \mathbf{v} = \alpha_1 \mathbf{v}_j \cdot \mathbf{v}_1 + \cdots + \alpha_j \mathbf{v}_j \cdot \mathbf{v}_j + \cdots + \alpha_n \mathbf{v}_j \cdot \mathbf{v}_n
\]

\[
= \alpha_j \mathbf{v}_j \cdot \mathbf{v}_j
\]

Hence

\[
\alpha_j = \frac{\mathbf{v}_j \cdot \mathbf{v}}{\mathbf{v}_j \cdot \mathbf{v}_j} = \frac{\mathbf{v}_j \cdot \mathbf{v}}{|\mathbf{v}_j|^2}
\]

(3)

Orthonormal bases (where all basis vectors have length 1) are very useful.
Linear mappings between vector spaces

- For $V, W$ vector spaces over $\mathbb{F}$, a map $T : V \rightarrow W$ is **linear** if for all vectors $v_i \in V$ and scalars $a_i \in \mathbb{F}$,

$$T(a_1 v_1 + a_2 v_2) = a_1 T v_1 + a_2 T v_2.$$ 

- Example (derivative): $T \equiv \left( \frac{d}{dx} \right)$

$$\left( \frac{d}{dx} \right)(af(x) + bg(x)) = a\left( \frac{d}{dx} \right)f(x) + b\left( \frac{d}{dx} \right)g(x)$$
Linear mappings between vector spaces

- For $V, W$ vector spaces over $\mathbb{F}$, a map $T : V \rightarrow W$ is **linear** if for all vectors $v_i \in V$ and scalars $a_i \in \mathbb{F}$,

  $$T(a_1v_1 + a_2v_2) = a_1 T v_1 + a_2 T v_2.$$ 

- Example (derivative): $T \equiv \left( \frac{d}{dx} \right)$

  $$\left( \frac{d}{dx} \right)(af(x) + bg(x)) = a\left( \frac{d}{dx} \right)f(x) + b\left( \frac{d}{dx} \right)g(x)$$

- Example (verify):

  $$T : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2 \mapsto \begin{pmatrix} 5x_1 - x_2 \\ x_1 + x_2 \\ x_1 - x_2 \end{pmatrix} \in \mathbb{R}^3$$
Linear mappings between vector spaces

- For $V, W$ vector spaces over $\mathbb{F}$, a map $T : V \rightarrow W$ is \textbf{linear} if for all vectors $v_i \in V$ and scalars $a_i \in \mathbb{F},$

  $T(a_1v_1 + a_2v_2) = a_1T(v_1) + a_2T(v_2).$

- Example (derivative): $T \equiv \left( \frac{d}{dx} \right)$

  $\left( \frac{d}{dx} \right)(af(x) + bg(x)) = a\left( \frac{d}{dx} \right)f(x) + b\left( \frac{d}{dx} \right)g(x)$

- Example (verify):

  $T : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \mathbb{R}^2 \mapsto \begin{pmatrix} 5x_1 - x_2 \\ x_1 + x_2 \\ x_1 - x_2 \end{pmatrix} \in \mathbb{R}^3$

- \textit{Hint:} Let $v_1 = (x_1, x_2)^T$ and $v_2 = (y_1, y_2)^T$. Thus, $a_1v_2 + a_2v_2 = (a_1x_1 + a_2y_1, a_1x_2 + a_2y_2)^T.$
Linear mappings between vector spaces

For $V, W$ vector spaces over $\mathbb{F}$, a map $T : V \rightarrow W$ is **linear** if for all vectors $v_i \in V$ and scalars $a_i \in \mathbb{F}$,

$$T(a_1 v_1 + a_2 v_2) = a_1 T v_1 + a_2 T v_2.$$  

By induction, this extends over any (finite) sum of vectors.

Thus, a linear map from $V$ to $W$ is completely defined by values it assigns to basis elements of $V$, and these values can be arbitrary vectors in $W$.

$$T : x_1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \cdots ?$$
Linear mappings as matrices

- For $V, W$ vector spaces over $\mathbb{F}$ with bases $\{v_1, \ldots, v_q\}$ and $\{w_1, \ldots, w_p\}$, a map $T : V \to W$ is completely specified by scalars $a_{ij} \in \mathbb{F}$, $(i = 1, \ldots, p, j = 1, \ldots, q)$, such that

$$Tv_j = \sum_{i=1}^{p} a_{ij}w_i.$$ 

Let $x \in \mathbb{F}^q$ with components $x_1, \ldots, x_q$, $y \in \mathbb{F}^p$ with components $y_1 \ldots, y_p$. Then, $Tx$ becomes

$$T \left( \sum_{j=1}^{q} x_j v_j \right) = \sum_{i=1}^{p} \left( \sum_{j=1}^{q} x_j a_{ij} \right) w_i = \sum_{i=1}^{p} y_i w_i \iff T_A x = y$$

- The entries $a_{ij} \in \mathbb{F}^{p \times q}$ are determined by the action of $T$ on the basis vectors.
Column space and Range of a matrix

Thus $Ax = y$ can be solved if and only if $y$ is a linear combination of columns of $A$.

- The column space of a matrix $A$ (denoted col $A$) is the subspace spanned by all linear combinations of the columns of $A$. 

![Graph showing column space and range of a matrix](image)
Thus \( Ax = y \) can be solved if and only if \( y \) is a linear combination of columns of \( A \).

- The column space of a matrix \( A \) (denoted \( \text{col } A \)) is the subspace spanned by all linear combinations of the columns of \( A \).
- This is also the range of the linear map: \( \text{range}(A) = \text{range}(AV) = \{ w \in W : w = Av \text{ for some } v \in V \} \)
Examples illustrating linear dependence and nullspace

Let \( B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \). For vector \( \mathbf{v} \) in direction \( \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \), \( B\mathbf{v} = 0 \).
Examples illustrating linear dependence and nullspace

Let \( B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \). For vector \( \mathbf{v} \) in direction \( \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \), \( B\mathbf{v} = 0 \).
Examples illustrating linear dependence and nullspace

- Let $B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$. For vector $v$ in direction $\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, $Bv = 0_v$ in nullspace or kernel of $B$.

- For $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, $\text{col}(1) + \text{col}(2) = \text{col}(3)$, so

$$\begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} - 1\begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Rightarrow \ker(A) = c \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$$
Examples illustrating linear dependence and nullspace

- Let \( B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \). For vector \( \mathbf{v} \) in direction \( \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \), \( B\mathbf{v} = 0 \). \( \mathbf{v} \) in nullspace or kernel of \( B \).

- For \( A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \), \( \text{col}(1) + \text{col}(2) = \text{col}(3), \) so

\[
\begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Rightarrow \ker(A) = c \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}
Examples illustrating linear dependence and nullspace

- Let $B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$. For vector $\mathbf{v}$ in direction $\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, $B\mathbf{v} = 0$. $\mathbf{v}$ is in nullspace or kernel of $B$.

- For $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, $\text{col}(1) + \text{col}(2) = \text{col}(3)$, so

  \[
  \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} 3 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \implies \ker(A) = c \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}
  \]

  Show $\ker(A^T) = c \begin{pmatrix} -1 \\ 1 \\ 2 \end{pmatrix}$.
Kernel or Null space of a matrix

In the previous example $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, there are 3 variables $v$ in $Av = y$ but only two independent equations.
Kernel or Null space of a matrix

- In the previous example $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, there are 3 variables $v$ in $Av = y$ but only two independent equations.

- If $Av = y$ and $x \in \ker(A)$ then $A(v + x) = y$. Either there are no solutions or there are (infinitely) many solutions.
Kernel or Null space of a matrix

- In the previous example $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, there are 3 variables $\mathbf{v}$ in $A\mathbf{v} = \mathbf{y}$ but only two independent equations.

- If $A\mathbf{v} = \mathbf{y}$ and $\mathbf{x} \in \ker(A)$ then $A(\mathbf{v} + \mathbf{x}) = \mathbf{y}$. Either there are no solutions or there are (infinitely) many solutions.
Kernel or Null space of a matrix

- In the previous example $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, there are 3 variables $v$ in $Av = y$ but only two independent equations.

- If $Av = y$ and $x \in \ker(A)$ then $A(v + x) = y$. Either there are no solutions or there are (infinitely) many solutions.

- The kernel of a map (or matrix) $\ker(A) = \text{nullspace } A = \{v \in V : Av = 0\}$.
Kernel or Null space of a matrix

- In the previous example $A = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$, there are 3 variables $v$ in $Av = y$ but only two independent equations.
- If $Av = y$ and $x \in \ker(A)$ then $A(v + x) = y$. Either there are no solutions or there are (infinitely) many solutions.
- The kernel of a map (or matrix) $\ker(A) = \text{nullspace } A = \{v \in V : Av = 0\}$.
- Let $A$ be a $3 \times q$ matrix.

$$A = \begin{pmatrix} \_ & \_ & u \\ \_ & \_ & v \\ \_ & \_ & w \end{pmatrix},$$

where $u$, $v$, and $w$ are $q$-dim row vectors. Then, $x \in \ker(A) \iff Ax = 0$. This means $x \perp \{u, v, w\}$. 
The rank (column rank) of $A$ is the dimension of the column space of $A$. 

- Rank of a matrix $= \text{number of independent equations}$
The **rank** (column rank) of A is the dimension of the column space of A.

A vector space is partitioned into its range and null spaces:

\[
\dim V = \dim \ker(A) + \dim \range(A).
\]

- **nullity**
- **rank**
The rank (column rank) of $A$ is the dimension of the column space of $A$.

A vector space is partitioned into its range and null spaces:

$$\dim V = \dim \ker(A) + \dim \text{range}(A).$$

We can do the same for the transpose: $\text{col}(A^T)$ and $\ker(A^T)$. 

**Rank of a matrix** = number of independent equations
The **rank** (column rank) of $A$ is the dimension of the column space of $A$.

A vector space is partitioned into its range and null spaces:

$$\dim V = \dim \ker(A) + \dim \text{range}(A).$$

We can do the same for the transpose: $\text{col}(A^T)$ and $\ker(A^T)$.

4 fundamental subspaces: $\text{col}(A)$, $\ker(A^T)$, $\text{col}(A^T)$ and $\ker(A^T)$.
Four fundamental subspaces of a matrix

Multiplying vectors by matrices iteratively introduces linear dependence

Let $A$ be a $n \times n$ (square) matrix. The multiplication of $A$ with vectors

$$ w = Av $$

defines a mapping (or transformation) of vectors $v \in V$ into vectors $w \in W$. For square matrix $A$, $V = W$. 
Multiplying vectors by matrices iteratively introduces linear dependence

- Let $A$ be a $n \times n$ (square) matrix. The multiplication of $A$ with vectors
  \[ w = Av \]
defines a mapping (or transformation) of vectors $v \in V$ into vectors $w \in W$. For square matrix $A$, $V = W$.

- For $v \neq 0$, the $(n + 1)$ vectors $v, Av, A^2v, \ldots A^n v$ cannot all be linearly independent if the rank of $A$ is $n$. 
Multiplying vectors by matrices iteratively introduces linear dependence

- Let \( A \) be a \( n \times n \) (square) matrix. The multiplication of \( A \) with vectors
  
  \[ w = Av \]

  defines a mapping (or transformation) of vectors \( v \in V \) into vectors \( w \in W \). For square matrix \( A \), \( V = W \).

- For \( v \neq 0 \), the \((n + 1)\) vectors \( v, Av, A^2v, \ldots A^n v \) cannot all be linearly independent if the rank of \( A \) is \( n \).

- Therefore there must be scalars \( a_0, a_1, \ldots, a_n \) such that
  
  \[
  (a_0I + a_1A^1 + \cdots + a_nA^n)v = 0, \text{ for } v \neq 0.
  \]
Linear dependence determines eigenvalues from matrix polynomials

- For a polynomial $f(x) = \sum_{i=0}^{n} a_i x^n$, $x \in \mathbb{F}$ define a **matrix polynomial** $f(A) = \sum_{i=0}^{n} a_i A^n$ where $A$ is a square matrix.
Linear dependence determines eigenvalues from matrix polynomials

- For a polynomial \( f(x) = \sum_{i=0}^{n} a_i x^n, x \in \mathbb{F} \) define a matrix polynomial \( f(A) = \sum_{i=0}^{n} a_i A^n \) where \( A \) is a square matrix.
- Therefore if \( f(x) = (x - \lambda_1)(x - \lambda_2) \cdots (x - \lambda_n) \), we can express \( f(A) = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I) \) where \( I \) is an \( n \times n \) identity matrix.
Linear dependence determines eigenvalues from matrix polynomials

- For a polynomial \( f(x) = \sum_{i=0}^{n} a_i x^n, x \in \mathbb{F} \) define a **matrix polynomial** \( f(A) = \sum_{i=0}^{n} a_i A^n \) where \( A \) is a square matrix.
- Therefore if \( f(x) = (x - \lambda_1)(x - \lambda_2) \cdots (x - \lambda_n) \), we can express \( f(A) = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I) \) where \( I \) is an \( n \times n \) identity matrix.
- Therefore, since

\[
0 = (a_0 I + a_1 A^1 + \cdots + a_n A^n)v = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I)v,
\]

at least one of \( (A - \lambda_k I) \) maps a non-zero vector in that space to 0.
Linear dependence determines eigenvalues from matrix polynomials

- For a polynomial $f(x) = \sum_{i=0}^{n} a_i x^i$, $x \in \mathbb{F}$ define a matrix polynomial $f(A) = \sum_{i=0}^{n} a_i A^i$ where $A$ is a square matrix.
- Therefore if $f(x) = (x - \lambda_1)(x - \lambda_2) \cdots (x - \lambda_n)$, we can express $f(A) = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I)$ where $I$ is an $n \times n$ identity matrix.
- Therefore, since

$$0 = (a_0 I + a_1 A^1 + \cdots + a_n A^n)v = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I)v,$$

at least one of $(A - \lambda_k I)$ maps a non-zero vector in that space to 0.
- This means that $\ker(A - \lambda_k I) \neq \{0\}$ and $(A - \lambda_k I)$ is not invertible.
Linear dependence determines eigenvalues from matrix polynomials

- For a polynomial $f(x) = \sum_{i=0}^{n} a_i x^i$, $x \in \mathbb{F}$ define a **matrix polynomial** $f(A) = \sum_{i=0}^{n} a_i A^i$ where $A$ is a square matrix.
- Therefore if $f(x) = (x - \lambda_1)(x - \lambda_2) \cdots (x - \lambda_n)$, we can express $f(A) = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I)$ where $I$ is an $n \times n$ identity matrix.
- Therefore, since

$$0 = (a_0 I + a_1 A^1 + \cdots + a_n A^n)\mathbf{v} = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_n I)\mathbf{v},$$

at least one of $(A - \lambda_k I)$ maps a non-zero vector in that space to 0.
- This means that $\ker(A - \lambda_k I) \neq \{0\}$ and $(A - \lambda_k I)$ is not invertible.
- $\lambda_1, \ldots, \lambda_n$ are the **eigenvalues** of $A$. 
For polynomials $f_1(x) = x^2 - 5x - 2$ and $f_2(x) = x^2 - 2x - 5$
compute $f_1(A)$ and $f_2(A)$ for $A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}$.
Matrix polynomials and eigenvalues: example

- For polynomials \( f_1(x) = x^2 - 5x - 2 \) and \( f_2(x) = x^2 - 2x - 5 \)
  compute \( f_1(A) \) and \( f_2(A) \) for \( A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} \):

- Check that \( A^2 = \begin{pmatrix} 7 & 4 \\ 6 & 7 \end{pmatrix} \)
Matrix polynomials and eigenvalues: example

- For polynomials \( f_1(x) = x^2 - 5x - 2 \) and \( f_2(x) = x^2 - 2x - 5 \) compute \( f_1(A) \) and \( f_2(A) \) for \( A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} \):

- Check that \( A^2 = \begin{pmatrix} 7 & 4 \\ 6 & 7 \end{pmatrix} \)

- \( f_1(A) = \begin{pmatrix} 0 & -6 \\ -9 & 0 \end{pmatrix} \) and \( f_2(A) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \)
Matrix polynomials and eigenvalues: example

For polynomials $f_1(x) = x^2 - 5x - 2$ and $f_2(x) = x^2 - 2x - 5$
compute $f_1(A)$ and $f_2(A)$ for $A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}$:

Check that $A^2 = \begin{pmatrix} 7 & 4 \\ 6 & 7 \end{pmatrix}$

$f_1(A) = \begin{pmatrix} 0 & -6 \\ -9 & 0 \end{pmatrix}$ and $f_2(A) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$

$f_2(A)$ is the characteristic polynomial of $A$.
$f_2(x) = (x - \lambda_1)(x - \lambda_2)$. $\lambda_{1,2}$ are the eigenvalues of the matrix $A$.
(In this example, $\lambda_{1,2} = (1 \pm \sqrt{6})$.)
Matrix polynomials and eigenvalues: example

- For polynomials $f_1(x) = x^2 - 5x - 2$ and $f_2(x) = x^2 - 2x - 5$
  compute $f_1(A)$ and $f_2(A)$ for $A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}$:

- Check that $A^2 = \begin{pmatrix} 7 & 4 \\ 6 & 7 \end{pmatrix}$

- $f_1(A) = \begin{pmatrix} 0 & -6 \\ -9 & 0 \end{pmatrix}$ and $f_2(A) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$

- $f_2(A)$ is the **characteristic polynomial** of $A$.
  $f_2(x) = (x - \lambda_1)(x - \lambda_2)$. $\lambda_{1,2}$ are the **eigenvalues** of the matrix $A$.
  (In this example, $\lambda_{1,2} = (1 \pm \sqrt{6})$.)

- In general, the characteristic polynomial of a matrix $A$ is denoted $\chi_A(x)$. So $\chi_A(x) = f_2(x)$ for this example.
Determinants and characteristic polynomials of matrices

- The **determinant** of a matrix \( T \), denoted \( \det(T) \) or \( |T| \), is the product of its eigenvalues.
- The characteristic polynomial \( \chi_T(x) \) of a matrix \( T \) equals \( \det(xI - T) \).
- You should all know the Laplace expansion of \( \det(T) \).
- Check that \( |xI - A| \) for \( A = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} \) is indeed \( f_2(A) \).
Eigenvalues: vectors $\mathbf{x}$ whose lengths are scaled by eigenvalue $\lambda$ upon action of $\mathbf{A}$

- The eigenvalue problem $\mathbf{A}\mathbf{x} = \lambda \mathbf{x}$: find, for a matrix $\mathbf{A}$, the eigenvectors $\mathbf{x}$ and eigenvalues $\lambda$.
- Example: Find the eigenvalues and eigenvectors of

$$\begin{pmatrix} 1 & 1 & -2 \\ -1 & 2 & 1 \\ 0 & 1 & -1 \end{pmatrix}$$
Eigenvectors: vectors $\mathbf{x}$ whose lengths are scaled by eigenvalue $\lambda$ upon action of $\mathbf{A}$

- The eigenvalue problem $\mathbf{Ax} = \lambda \mathbf{x}$: find, for a matrix $\mathbf{A}$, the eigenvectors $\mathbf{x}$ and eigenvalues $\lambda$.
- Example: Find the eigenvalues and eigenvectors of

$$
\mathbf{A} = \begin{pmatrix}
1 & 1 & -2 \\
-1 & 2 & 1 \\
0 & 1 & -1
\end{pmatrix}
$$

- **STEP I:** Compute the characteristic polynomial of $\mathbf{A}$ and find its roots:

$$
-\chi_{\mathbf{A}}(\lambda) = \begin{vmatrix}
1 - \lambda & 1 & -2 \\
-1 & 2 - \lambda & 1 \\
0 & 1 & -1 - \lambda
\end{vmatrix} = (1 - \lambda)(2 - \lambda)(-1 - \lambda)
$$
Eigenvectors: vectors $\mathbf{x}$ whose lengths are scaled by eigenvalue $\lambda$ upon action of $\mathbf{A}$

- The eigenvalue problem $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$: find, for a matrix $\mathbf{A}$, the eigenvectors $\mathbf{x}$ and eigenvalues $\lambda$.
- Example: Find the eigenvalues and eigenvectors of

$$
\mathbf{A} = \begin{pmatrix}
1 & 1 & -2 \\
-1 & 2 & 1 \\
0 & 1 & -1
\end{pmatrix}
$$

**STEP I:** Compute the characteristic polynomial of $\mathbf{A}$ and find its roots:

$$
-\chi_\mathbf{A}(\lambda) = \begin{vmatrix}
1 - \lambda & 1 & -2 \\
-1 & 2 - \lambda & 1 \\
0 & 1 & -1 - \lambda
\end{vmatrix} = (1 - \lambda)(2 - \lambda)(-1 - \lambda)
$$

$\chi_\mathbf{A}(\lambda) = 0 \Rightarrow \lambda_1 = 2, \lambda_2 = 1$ and $\lambda_3 = -1.$
STEP II:
For each eigenvalue, we need to compute the corresponding eigenvectors. We demonstrate this only for $\lambda_1 = 2$. Setting $\lambda = 2$, denoting the corresponding eigenvector by $v_1 = (x \ y \ z)^T \in \ker(A - \lambda_1 I)$, compute

$$(A - 2I)v_1 = \begin{pmatrix} -1 & 1 & -2 \\ -1 & 0 & 1 \\ 0 & 1 & -3 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$$

This is equivalent to the set of equations

\[-x + y - 2z = 0 \]
\[-x + z = 0 \]
\[y - 3z = 0 \]

Solution $v_1 = c(1 \ 3 \ 1)^T$
Often, one computes the *normalized* eigenvectors \( \hat{\mathbf{v}} \), which have unit length \( |\hat{\mathbf{v}}| = 1 \). In our case,

\[
|\hat{\mathbf{v}}_1| = \frac{\mathbf{v}_1}{|\mathbf{v}_1|} = \frac{\mathbf{v}_1}{\sqrt{11}} = \frac{1}{\sqrt{11}} (1 \; 3 \; 1)^T
\]

Proceeding in a similar way with the two other eigenvalues, we get the set of eigenvectors \( \mathbf{v}_1 = \begin{pmatrix} 1 \\ 3 \\ 1 \end{pmatrix}, \mathbf{v}_2 = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix} \) and \( \mathbf{v}_3 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \).

**In python (seek the online Help):**

```
np.linalg.eig?
```
Example: eigenvectors of repeated eigenvalues

You may not always get distinct eigenvalues (like we did in the previous case). Let

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 2 \\ 0 & 2 & 1 \\ -1 & 2 & 2 \end{pmatrix} \quad (4)$$

Solving $\chi_{\mathbf{A}}(\lambda) = 0$ gives the eigenvalues $\lambda_1 = \lambda_2 = 2$ and $\lambda_3 = 1$. For the eigenvalue $\lambda_3$ we find $\mathbf{v}_3 = (1 \ 1 \ -1)^T$ as an eigenvector. If we set $\lambda_{1,2} = 2$ and $\mathbf{v}_{1,2} = (x \ y \ z)^T$ the equation $(\mathbf{A} - 2\mathbf{I})\mathbf{v}_{1,2} = 0$ gives

$$-x + 2y + 2z = 0$$
$$z = 0$$
$$-x + 2y = 0$$

yielding $z = 0$ and $x = 2y$ and $\mathbf{v}_{1,2} = (2 \ 1 \ 0)^T$. Hence, we get only two linear independent eigenvectors.
Linear independence of eigenvectors

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.
Linear independence of eigenvectors*

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.

- **Proof (by contradiction):** Assume linear dependence: $\exists a_1, \ldots, a_m$, all $a_i \neq 0$ such that $w := a_1 v_1 + \cdots + a_m v_m = 0$. 

Details of the proof are omitted here.
Linear independence of eigenvectors*

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.

- **Proof (by contradiction):** Assume linear dependence: $\exists a_1, \ldots, a_m$, all $a_i \neq 0$ such that $w := a_1 v_1 + \cdots a_m v_m = 0$. 

Linear independence of eigenvectors*

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.

- **Proof (by contradiction):** Assume linear dependence: $\exists a_1, \ldots, a_m$, all $a_i \neq 0$ such that $w := a_1 v_1 + \cdots a_m v_m = 0$. For $k = 1, \ldots, m$ we apply to the zero vector $w$ the operators $\tilde{A}_k$ defined thus (missing $k$-th factor):

  $\tilde{A}_k = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_{k-1} I)(A - \lambda_{k+1} I) \cdots (A - \lambda_m).$
Linear independence of eigenvectors

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.

- **Proof (by contradiction):** Assume linear dependence: $\exists a_1, \ldots, a_m$, all $a_i \neq 0$ such that $w := a_1 v_1 + \cdots a_m v_m = 0$. For $k = 1, \ldots, m$ we apply to the zero vector $w$ the operators $\tilde{A}_k$ defined thus (missing $k$-th factor):

  $$\tilde{A}_k = (A - \lambda_1 I)(A - \lambda_2 I) \cdots (A - \lambda_{k-1} I)(A - \lambda_{k+1} I) \cdots (A - \lambda_m).$$

  For example, for $k = 1$ we have

  $$0 = \tilde{A}_1 w = a_1 (\lambda_1 - \lambda_2)(\lambda_1 - \lambda_3) \cdots (\lambda_1 - \lambda_m)v_1 \Rightarrow a_1 = 0.$$
Linear independence of eigenvectors*

- If matrix $A$ has $m$ distinct eigenvalues $\lambda_1, \ldots, \lambda_m$, then the corresponding eigenvectors $v_1, \ldots, v_m$ are linearly independent.

- **Proof (by contradiction):** Assume linear dependence: $\exists a_1, \ldots, a_m$, all $a_i \neq 0$ such that $w := a_1 v_1 + \cdots + a_m v_m = 0$. For $k = 1, \ldots, m$ we apply to the zero vector $w$ the operators $\tilde{A}_k$ defined thus (missing $k$-th factor):

  $$
  \tilde{A}_k = (A - \lambda_1 I)(A - \lambda_2 I)\cdots(A - \lambda_{k-1} I)(A - \lambda_{k+1} I)\cdots(A - \lambda_m).
  $$

  For example, for $k = 1$ we have

  $$
  0 = \tilde{A}_1 w = a_1(\lambda_1 - \lambda_2)(\lambda_1 - \lambda_3)\cdots(\lambda_1 - \lambda_m)v_1 \Rightarrow a_1 = 0.
  $$

  Similarly, $0 = \tilde{A}_k w \Rightarrow a_k = 0$. For $w = 0$ all $a_k = 0$ necessarily. *Contradicts assumption.* □
Complex eigenvalues*

Eigenvalues may not be real numbers, even if the matrix elements are real:

$$A = \begin{pmatrix} 1 & -2 \\ 2 & 0 \end{pmatrix}$$

Setting $|A - \lambda I| = \lambda^2 - \lambda + 4 = 0$, we obtain

$$\lambda_{1,2} = (1 \pm \sqrt{-15})/2 = (1 \pm i\sqrt{15})/2$$

with $i = \sqrt{-1}$. 
Real symmetric and hermitian matrices

- A matrix $A$ is called real symmetric if all matrix elements are real numbers and $A^T = A$, where $A^T$ is the transpose of $A$.
Real symmetric and hermitian matrices

- A matrix $\mathbf{A}$ is called *real symmetric* if all matrix elements are real numbers and $\mathbf{A}^T = \mathbf{A}$, where $\mathbf{A}^T$ is the transpose of $\mathbf{A}$.
- For a matrix $\mathbf{A}$ with elements $(\mathbf{A})_{ij} = a_{ij}$, $\cdot^T$ is defined as $(\mathbf{A}^T)_{ij} = a_{ji}$.
Real symmetric and hermitian matrices

- A matrix $\mathbf{A}$ is called *real symmetric* if all matrix elements are real numbers and $\mathbf{A}^T = \mathbf{A}$, where $\mathbf{A}^T$ is the transpose of $\mathbf{A}$.
- For a matrix $\mathbf{A}$ with elements $(\mathbf{A})_{ij} = a_{ij}$, $\cdot^T$ is defined as $(\mathbf{A}^T)_{ij} = a_{ji}$.
- $\cdot^\dagger$ is an operation called hermitian conjugation, defined as $(\mathbf{A}^\dagger)_{ij} = a_{ji}^*$, and $\mathbf{A}^\dagger$ is referred to as “$\mathbf{A}$-dagger.”
A matrix $A$ is called *real symmetric* if all matrix elements are real numbers and $A^T = A$, where $A^T$ is the transpose of $A$.

For a matrix $A$ with elements $(A)_{ij} = a_{ij}$, $\cdot^T$ is defined as $(A^T)_{ij} = a_{ji}$.

$\cdot^\dagger$ is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “$A$-dagger.”

**NB:** For real matrices, $A^\dagger = A^T$. 

Real symmetric and hermitian matrices
Real symmetric and hermitian matrices

- A matrix $A$ is called *real symmetric* if all matrix elements are real numbers and $A^T = A$, where $A^T$ is the transpose of $A$.
- For a matrix $A$ with elements $(A)_{ij} = a_{ij}$, $\cdot^T$ is defined as $(A^T)_{ij} = a_{ji}$.
- $\cdot^\dagger$ is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “A-dagger.”
- **NB:** For real matrices, $A^\dagger = A^T$.
- A matrix $A$ is called *hermitian* if all matrix elements are complex numbers and $A^\dagger := (A^*)^T = A$, where $A^*$ is the matrix whose elements are complex conjugates of those in $A$. 

-y is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “A-dagger.”

**NB:** For real matrices, $A^\dagger = A^T$.

A matrix $A$ is called *hermitian* if all matrix elements are complex numbers and $A^\dagger := (A^*)^T = A$, where $A^*$ is the matrix whose elements are complex conjugates of those in $A$. 

-y is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “A-dagger.”

**NB:** For real matrices, $A^\dagger = A^T$.

A matrix $A$ is called *hermitian* if all matrix elements are complex numbers and $A^\dagger := (A^*)^T = A$, where $A^*$ is the matrix whose elements are complex conjugates of those in $A$. 

-y is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “A-dagger.”

**NB:** For real matrices, $A^\dagger = A^T$.
Real symmetric and hermitian matrices

- A matrix $A$ is called *real symmetric* if all matrix elements are real numbers and $A^T = A$, where $A^T$ is the transpose of $A$.
- For a matrix $A$ with elements $(A)_{ij} = a_{ij}$, $\cdot^T$ is defined as $(A^T)_{ij} = a_{ji}$.
- $\cdot^\dagger$ is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “A-dagger.”
- **NB:** For real matrices, $A^\dagger = A^T$.
- A matrix $A$ is called *hermitian* if all matrix elements are complex numbers and $A^\dagger := (A^*)^T = A$, where $A^*$ is the matrix whose elements are complex conjugates of those in $A$.
- *The eigenvalues of a hermitian matrix are real.*
Real symmetric and hermitian matrices

- A matrix $A$ is called **real symmetric** if all matrix elements are real numbers and $A^T = A$, where $A^T$ is the transpose of $A$.
- For a matrix $A$ with elements $(A)_{ij} = a_{ij}$, $\cdot^T$ is defined as $(A^T)_{ij} = a_{ji}$.
- $\cdot^\dagger$ is an operation called hermitian conjugation, defined as $(A^\dagger)_{ij} = a_{ji}^*$, and $A^\dagger$ is referred to as “$A$-dagger.”
- **NB:** For real matrices, $A^\dagger = A^T$.
- A matrix $A$ is called **hermitian** if all matrix elements are complex numbers and $A^\dagger := (A^*)^T = A$, where $A^*$ is the matrix whose elements are complex conjugates of those in $A$.
- *The eigenvalues of a hermitian matrix are real.*
- Certain types of symmetric matrices (covariance matrices and kernel/gram matrices) are often generated from data sets, and are used extensively in ML algorithms. You’ll see a few examples as we go through the course.
Eigenvectors of real symmetric matrices

We show the following:

- Let $A$ be a real symmetric matrix. Then eigenvectors associated with distinct eigenvalues are orthogonal.
Eigenvectors of real symmetric matrices

We show the following:

- Let $A$ be a real symmetric matrix. Then eigenvectors associated with distinct eigenvalues are orthogonal.

**Proof:** Let $u$ and $v$ be two eigenvectors with distinct eigenvalues $\lambda$ and $\mu$ respectively, i.e. $Au = \lambda u$ and $Av = \mu v$. We shall prove $u^T v = 0$.

With some more effort one can show (even with repeated eigenvalues!) that for any real symmetric $n \times n$ matrix we can find $n$ orthogonal eigenvectors.
Eigenvectors of real symmetric matrices

We show the following:

- Let $A$ be a real symmetric matrix. Then eigenvectors associated with distinct eigenvalues are orthogonal.

**Proof:** Let $u$ and $v$ be two eigenvectors with distinct eigenvalues $\lambda$ and $\mu$ respectively, i.e. $Au = \lambda u$ and $Av = \mu v$. We shall prove $u^T v = 0$.

With some more effort one can show (even with repeated eigenvalues!) that for any real symmetric matrix we can find $n$ orthogonal eigenvectors.
Eigenvectors of real symmetric matrices

We show the following:

- **Let** $A$ **be a real symmetric matrix.** Then eigenvectors associated with distinct eigenvalues are orthogonal.

- **Proof:** Let $u$ and $v$ be two eigenvectors with distinct eigenvalues $\lambda$ and $\mu$ respectively, i.e. $Au = \lambda u$ and $Av = \mu v$. We shall prove $u^T v = 0$.

  Since $A$ is symmetric, $(Au)^T = u^T A$.
Eigenvectors of real symmetric matrices

We show the following:

- **Let** \( \mathbf{A} \) **be a real symmetric matrix. Then eigenvectors associated with distinct eigenvalues are orthogonal.**

- **Proof:** Let \( \mathbf{u} \) and \( \mathbf{v} \) be two eigenvectors with distinct eigenvalues \( \lambda \) and \( \mu \) respectively, i.e. \( \mathbf{A}\mathbf{u} = \lambda \mathbf{u} \) and \( \mathbf{A}\mathbf{v} = \mu \mathbf{v} \). We shall prove \( \mathbf{u}^T \mathbf{v} = 0 \).

  Since \( \mathbf{A} \) is symmetric, \( (\mathbf{A}\mathbf{u})^T = \mathbf{u}^T \mathbf{A} \). Therefore \( \mathbf{A}\mathbf{u} = \lambda \mathbf{u} \iff \mathbf{u}^T \mathbf{A}^T = \lambda \mathbf{u}^T \).
Eigenvectors of real symmetric matrices

We show the following:

- Let $A$ be a real symmetric matrix. Then eigenvectors associated with distinct eigenvalues are orthogonal.

Proof: Let $u$ and $v$ be two eigenvectors with distinct eigenvalues $\lambda$ and $\mu$ respectively, i.e. $Au = \lambda u$ and $Av = \mu v$. We shall prove $u^T v = 0$.

Since $A$ is symmetric, $(Au)^T = u^T A$. Therefore $Au = \lambda u \iff u^T A^T = \lambda u^T$. Multiply on the right with $v$:

$$\lambda u^T v = u^T Av = \mu u^T v.$$

Hence $(\lambda - \mu)u^T v = 0$ and, since $\mu \neq \lambda$, $u^T v = 0$.

- With some more effort one can show (even with repeated eigenvalues!) that for any real symmetric $n \times n$ matrix we can find $n$ orthogonal eigenvectors.
Singular Value Decomposition (SVD) of a Matrix

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis in that vector space.
Singular Value Decomposition (SVD) of a Matrix

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis in that vector space.
- SVD measures how a circle is mapped into an ellipse; how an $n$-dimensional hyper-sphere is mapped into an $n$-dimensional hyper-ellipse.
Singular Value Decomposition (SVD) of a Matrix

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis in that vector space.
- SVD measures how a circle is mapped into an ellipse; how an $n$-dimensional hyper-sphere is mapped into an $n$-dimensional hyper-ellipse.
Singular Value Decomposition (SVD) of a Matrix

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis in that vector space.
- SVD measures how a circle is mapped into an ellipse; how an $n$-dimensional hyper-sphere is mapped into an $n$-dimensional hyper-ellipse.

Action of \[
\begin{pmatrix}
1.0 & 2.0 \\
0.5 & 2.5
\end{pmatrix}
on unit vectors \[
\begin{pmatrix}
1 \\
0
\end{pmatrix}and \[
\begin{pmatrix}
0 \\
1
\end{pmatrix}.

The lengths of the semi-major axes of the hyper-ellipse are properties of the map.
Singular Value Decomposition (SVD) of a Matrix

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis in that vector space.
- SVD measures how a circle is mapped into an ellipse; how an \( n \)-dimensional hyper-sphere is mapped into an \( n \)-dimensional hyper-ellipse.

Action of \[
\begin{pmatrix}
1.0 & 2.0 \\
0.5 & 2.5
\end{pmatrix}
\] on unit vectors \[
\begin{pmatrix}
1 \\
0
\end{pmatrix}
\] and \[
\begin{pmatrix}
0 \\
1
\end{pmatrix}
\]. The lengths of the semi-major axes of the hyper-ellipse are properties of the map.
In pictures: mapping a unit circle into an ellipse

Even when the vectors in the domain and range of the map change, their locus displays the geometrical character of the transformation enacted by the matrix.
In pictures: mapping a unit circle into an ellipse

- Even when the vectors in the domain and range of the map change, their locus displays the geometrical character of the transformation enacted by the matrix.
- While the displayed pairs of vectors in the domain (red) are orthogonal by construction, the pairs they map to (blue) are usually not.
Example of SVD for recommender matrix

\[ U = \begin{pmatrix}
\frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\
\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\
\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\
0 & 1 & 0
\end{pmatrix} \]

\[ \Sigma = \begin{pmatrix}
\sqrt{3} & 0 & 0 & 0 & 0 \\
0 & \sqrt{2} & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0
\end{pmatrix} \]

\[ V = \begin{pmatrix}
0 & \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} \\
\frac{1}{\sqrt{6}} & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} \\
\frac{1}{\sqrt{6}} & 0 & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} \\
0 & \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{3}} \\
\sqrt{\frac{2}{3}} & 0 & 0 & \frac{1}{\sqrt{3}} & 0
\end{pmatrix} \]

\[ A = \begin{pmatrix}
0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 0
\end{pmatrix} \]

\[ A = U \Sigma V^T \]
The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{v_1, v_2\} \) in that vector space, here 2-dimensional. So, \( x = (v_1^T x)v_1 + (v_2^T x)v_2 \).
Example of SVD

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{v_1, v_2\} \) in that vector space, here 2-dimensional. So, \( x = (v_1^T x)v_1 + (v_2^T x)v_2 \).
- The set of vector equations \( Av_j = \sigma_j u_j \) for \( j = 1, 2 \) becomes:

\[
A x = (v_1^T x)Av_1 + (v_2^T x)Av_2
= (v_1^T x)\sigma_1 u_1 + (v_2^T x)\sigma_2 u_2
\Rightarrow A = v_1^T \sigma_1 u_1 + v_2^T \sigma_2 u_2
\]
The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{ \mathbf{v}_1, \mathbf{v}_2 \} \) in that vector space, here 2-dimensional. So, \( \mathbf{x} = (\mathbf{v}_1^T \mathbf{x}) \mathbf{v}_1 + (\mathbf{v}_2^T \mathbf{x}) \mathbf{v}_2 \).

The set of vector equations \( \mathbf{A} \mathbf{v}_j = \sigma_j \mathbf{u}_j \) for \( j = 1, 2 \) becomes:

\[
\begin{align*}
\mathbf{A} \mathbf{x} &= (\mathbf{v}_1^T \mathbf{x}) \mathbf{A} \mathbf{v}_1 + (\mathbf{v}_2^T \mathbf{x}) \mathbf{A} \mathbf{v}_2 \\
&= (\mathbf{v}_1^T \mathbf{x}) \sigma_1 \mathbf{u}_1 + (\mathbf{v}_2^T \mathbf{x}) \sigma_2 \mathbf{u}_2 \\
\Rightarrow \mathbf{A} &= \mathbf{v}_1 \sigma_1 \mathbf{u}_1 + \mathbf{v}_2 \sigma_2 \mathbf{u}_2
\end{align*}
\]

Express that as \( \mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^T \), with \( \mathbf{U} \) containing the columns of \( \mathbf{u}_i \), \( \mathbf{V} \) the columns of \( \mathbf{v}_i \), and \( \Sigma \) a diagonal matrix with \( \sigma_i \) along the diagonal.
The reduced SVD – the range may not have a basis

The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{v_1, \ldots, v_n\} \) in that vector space. The set of vector equations \( A v_j = \sigma_j u_j \) for \( j = 1, \ldots, n \) may be expressed as a matrix equation \( AV = \hat{U} \hat{\Sigma} \):

\[
\begin{pmatrix}
A \\
\end{pmatrix}
\begin{pmatrix}
v_1 & \cdots & v_n \\
\end{pmatrix}
= 
\begin{pmatrix}
u_1 & \cdots & u_n \\
\end{pmatrix}
\begin{pmatrix}
\sigma_1 & & \\
& \ddots & \\
& & \sigma_n \\
\end{pmatrix}
\]
The reduced SVD – the range may not have a basis

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{v_1, \ldots, v_n\} \) in that vector space. The set of vector equations \( Av_j = \sigma_j u_j \) for \( j = 1, \ldots, n \) may be expressed as a matrix equation \( AV = \hat{U} \hat{\Sigma} \):

\[
\begin{pmatrix}
   A \\
   \end{pmatrix} 
\begin{pmatrix}
   v_1 & \cdots & v_n \\
   \end{pmatrix} = 
\begin{pmatrix}
   u_1 & \cdots & u_n \\
   \end{pmatrix} 
\begin{pmatrix}
   \sigma_1 & \cdots \\
   \end{pmatrix} 
\]

- Since \( V \) is an orthogonal (unitary) matrix, \( V^T V = VV^T = I \) \((V^\dagger V = VV^\dagger = I)\),

\[
A = \hat{U} \hat{\Sigma} V^\dagger
\]
The reduced SVD – the range may not have a basis

- The action of an arbitrary matrix on a vector space can be pieced together from its action on an orthonormal basis \( \{v_1, \ldots, v_n\} \) in that vector space. The set of vector equations \( A v_j = \sigma_j u_j \) for \( j = 1, \ldots, n \) may be expressed as a matrix equation \( A V = \hat{U} \hat{\Sigma} \):

\[
\begin{pmatrix}
A
\end{pmatrix}
\begin{pmatrix}
v_1 & \cdots & v_n
\end{pmatrix}
= 
\begin{pmatrix}
u_1 & \cdots & u_n
\end{pmatrix}
\begin{pmatrix}
\sigma_1 & & \\
& \ddots & \\
& & \sigma_n
\end{pmatrix}
\]

- Since \( V \) is an orthogonal (unitary) matrix, \( V^T V = V V^T = I \) \( (V^\dagger V = V V^\dagger = I) \),

\[
A = \hat{U} \hat{\Sigma} V^\dagger
\]

- The columns of \( \hat{U} \) are \( n \) orthonormal vectors in \( \mathbb{C}^m \) \( (m \geq n) \).
The full SVD describes both the domain and range of a matrix by orthonormal bases

- Extend the size of the vector space in the range from $n$ to $m$ by adding columns to $\hat{U}$ to yield a $m \times m$ unitary (for complex) or orthogonal (for real) matrix $U$. 

$$A = U \Sigma V^T$$

The columns of $V$ and $U$ are the right and left singular vectors, and the diagonal entries of $\Sigma$ are the singular values of $A$. 
The full SVD describes both the domain and range of a matrix by orthonormal bases

- Extend the size of the vector space in the range from $n$ to $m$ by adding columns to $\hat{U}$ to yield a $m \times m$ unitary (for complex) or orthogonal (for real) matrix $U$.
- To maintain the same value for the product of matrices (after all, we need to recover $A$ from its factors), extend matrix $\hat{\Sigma}$ by adding zeros along the diagonal to obtain matrix $\Sigma$. 
The full SVD describes both the domain and range of a matrix by orthonormal bases

- Extend the size of the vector space in the range from \( n \) to \( m \) by adding columns to \( \hat{U} \) to yield a \( m \times m \) unitary (for complex) or orthogonal (for real) matrix \( U \).
- To maintain the same value for the product of matrices (after all, we need to recover \( A \) from its factors), extend matrix \( \hat{\Sigma} \) by adding zeros along the diagonal to obtain matrix \( \Sigma \).
- For an arbitrary matrix \( A \in \mathbb{C}^{m \times n} \) we have an \( n \times n \) matrix \( V \) and a \( m \times m \) matrix \( U \) that are both orthonormal, and a \( m \times n \) matrix \( \Sigma \) whose non-zero entries \( \sigma_i = \Sigma_{ii} \) are along the diagonal:

\[
A = U\Sigma V^\dagger
\]
The full SVD describes both the domain and range of a matrix by orthonormal bases

- Extend the size of the vector space in the range from \( n \) to \( m \) by adding columns to \( \hat{U} \) to yield a \( m \times m \text{ unitary} \) (for complex) or \( \text{orthogonal} \) (for real) matrix \( U \).
- To maintain the same value for the product of matrices (after all, we need to recover \( A \) from its factors), extend matrix \( \hat{\Sigma} \) by adding zeros along the diagonal to obtain matrix \( \Sigma \).
- For an arbitrary matrix \( A \in \mathbb{C}^{m \times n} \) we have an \( n \times n \) matrix \( V \) and a \( m \times m \) matrix \( U \) that are both orthonormal, and a \( m \times n \) matrix \( \Sigma \) whose non-zero entries \( \sigma_i = \Sigma_{ii} \) are along the diagonal:

\[
A = UV^\dagger
\]

- The columns of \( V \) and \( U \) are the right and left singular vectors, and the diagonal entries of \( \Sigma \) are the singular values of \( A \).
Geometry of SVD: choice of basis vectors lying on circle and map

Choose the pre-image of the orthogonal pair in the range of the map.
Singular vectors describe spheres and ellipsoids by semi-major axes

- There is one choice of vector pairs (basis) in the domain that gets mapped into an orthogonal pair along the major axes of the ellipse.
Singular vectors describe spheres and ellipsoids by semi-major axes

- There is one choice of vector pairs (basis) in the domain that gets mapped into an orthogonal pair along the major axes of the ellipse.
- These pairs are the singular vectors of the matrix. The lengths of the semi-major axes of the ellipse are the singular values.
- There will be left and right singular vectors.
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform

\[
y = \arg\min_{v \in \mathcal{Y}} \| x - v \|.
\]
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following

  To find a vector $\mathbf{y}$ from a set $\mathcal{Y}$ closest to $\mathbf{x}$ we perform

  $$\mathbf{y} = \text{argmin}_{\mathbf{v} \in \mathcal{Y}} \| \mathbf{x} - \mathbf{v} \|.$$ 

- For $\mathbf{z} = (z_1, \ldots, z_n)$, $\| \mathbf{z} \|$ is a norm – several choices:
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform
  \[
  y = \arg\min_{v \in \mathcal{Y}} \|x - v\|.
  \]
- For $z = (z_1, \ldots, z_n)$, $\|z\|$ is a norm — several choices:
  - $L_2$ norm: $\|z\|_2 = \sqrt{z \cdot z} = \sqrt{\sum_i z_i^2}$
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector \( y \) from a set \( \mathcal{Y} \) closest to \( x \) we perform

\[
    y = \arg\min_{v \in \mathcal{Y}} \| x - v \|.
\]

- For \( z = (z_1, \ldots, z_n) \), \( \| z \| \) is a \textbf{norm} – several choices:
  - \( L_2 \) norm: \( \| z \|_2 = \sqrt{z \cdot z} = \sqrt{\sum_i z_i^2} \)
  - \( L_1 \) norm: \( \| z \|_1 = \sum_i |z_i| \)
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform

$$y = \arg\min_{v \in \mathcal{Y}} \|x - v\|.$$  

- For $z = (z_1, \ldots, z_n)$, $\|z\|$ is a norm – several choices:
  - $L_2$ norm: $\|z\|_2 = \sqrt{\sum z_i^2}$
  - $L_1$ norm: $\|z\|_1 = \sum |z_i|$  
  - $L_p$ norm: $\|z\|_p = (\sum |z_i|^p)^{1/p}$
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector $\mathbf{y}$ from a set $\mathcal{Y}$ closest to $\mathbf{x}$ we perform
  
  $$ \mathbf{y} = \arg\min_{\mathbf{v} \in \mathcal{Y}} \| \mathbf{x} - \mathbf{v} \|_2. $$

- For $\mathbf{z} = (z_1, \ldots, z_n)$, $\| \mathbf{z} \|$ is a norm – several choices:
  - $L_2$ norm: $\| \mathbf{z} \|_2 = \sqrt{\mathbf{z} \cdot \mathbf{z}} = \sqrt{\sum_i z_i^2}$
  - $L_1$ norm: $\| \mathbf{z} \|_1 = \sum_i |z_i|$
  - $L_p$ norm: $\| \mathbf{z} \|_p = (\sum_i |z_i|^p)^{1/p}$
  - $L_0$ norm: $\| \mathbf{z} \|_0 = \#(i |z_i \neq 0)$
How is the SVD made useful in machine learning?

- **Distance minimisation:** matrix generalisation of the following
- To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform
  \[
  y = \arg \min_{v \in \mathcal{Y}} \| x - v \|.
  \]

- For $z = (z_1, \ldots, z_n)$, $\| z \|$ is a **norm** — several choices:
  - $L_2$ norm: $\| z \|_2 = \sqrt{z \cdot z} = \sqrt{\sum_i z_i^2}$
  - $L_1$ norm: $\| z \|_1 = \sum_i |z_i|$
  - $L_p$ norm: $\| z \|_p = (\sum_i |z_i|^p)^{1/p}$
  - $L_0$ norm: $\| z \|_0 = \#(i |z_i \neq 0)$
  - $L_\infty$ norm: $\| z \|_\infty = \max(|z_i|), 1 \leq i \leq n$. 

Distance minimisation: matrix generalisation of the following

To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform

$$y = \arg \min_{v \in \mathcal{Y}} \| x - v \|.$$
How is the SVD made useful in machine learning?

- Distance minimisation: matrix generalisation of the following
- To find a vector $y$ from a set $\mathcal{Y}$ closest to $x$ we perform

$$y = \arg\min_{v \in \mathcal{Y}} \|x - v\|.$$

- For $z = (z_1, \ldots, z_n)$, $\|z\|$ is a norm – several choices:
  - $L_2$ norm: $\|z\|_2 = \sqrt{z \cdot z} = \sqrt{\sum z_i^2}$
  - $L_1$ norm: $\|z\|_1 = \sum |z_i|$
  - $L_p$ norm: $\|z\|_p = \left(\sum |z_i|^p\right)^{1/p}$
  - $L_0$ norm: $\|z\|_0 = \#(i | z_i \neq 0)$
  - $L_\infty$ norm: $\|z\|_\infty = \max(|z_i|), 1 \leq i \leq n.$

- SVD helps find a matrix $\tilde{X}$ from a set $\mathcal{M}$ closest to given matrix $X$:

$$\tilde{X} = \arg\min_{Y \in \mathcal{M}} \|X - Y\|_2.$$
SVD gives low-rank approximation of matrices

- We seek $\tilde{X} = \text{argmin}_{Y \in \mathcal{M}} \|X - Y\|_2$. 

SVD gives low-rank approximation of matrices

- We seek \( \tilde{X} = \arg\min_{Y \in \mathcal{M}} \|X - Y\|_2 \).
- By partitioning the numerically ordered diagonal entries of \( \Sigma \) into the first \( k \) and the rest, we have (from the SVD)

\[
A = U\Sigma V^T = U_k \Sigma_k V_k^T + U_\perp \Sigma_\perp V_\perp^T
= (U_k \ U_\perp) \begin{pmatrix} \Sigma_k \\ \Sigma_\perp \end{pmatrix} \begin{pmatrix} V_k^T \\ V_\perp^T \end{pmatrix}
\approx U_k \Sigma_k V_k^T \equiv \tilde{A}_k
\]
SVD gives low-rank approximation of matrices

- We seek \( \tilde{X} = \operatorname{argmin}_{Y \in \mathcal{M}} \| X - Y \|_2 \).
- By partitioning the numerically ordered diagonal entries of \( \Sigma \) into the first \( k \) and the rest, we have (from the SVD)

\[
\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T = \mathbf{U}_k \Sigma_k \mathbf{V}_k^T + \mathbf{U}_\perp \Sigma_\perp \mathbf{V}_\perp^T
\]

\[
= (\mathbf{U}_k \ \mathbf{U}_\perp) \begin{pmatrix} \Sigma_k \\ \Sigma_\perp \end{pmatrix} \begin{pmatrix} \mathbf{V}_k^T \\ \mathbf{V}_\perp^T \end{pmatrix}
\]

\[
\approx \mathbf{U}_k \Sigma_k \mathbf{V}_k^T \equiv \tilde{\mathbf{A}}_k
\]

- \( \mathbf{A} \) is replaced by the rank \( k \) matrix \( \tilde{\mathbf{A}}_k \). Of all possible rank-\( k \) matrices \( \mathbf{B} \in \mathcal{M}_k \), \( \tilde{\mathbf{A}}_k \) constructed via the SVD gives the best approximation to \( \mathbf{A} \) in the sense that it minimises the \( L_2 \)-norm:

\[
\tilde{\mathbf{A}}_k = \operatorname{argmin}_{\mathbf{B} \in \mathcal{M}_k} \| \mathbf{A} - \mathbf{B} \|_2.
\]
Linear regression using SVD: find $w$ for smallest $\|A w - y\|_2$

- A vector $w$ that is closest to target vector $y$ along direction $u$ is $w = x^*v$. Proof:

\[
x^* = \text{argmin}_{x \in \mathbb{R}} \|y - xu\|_2 = \frac{y \cdot u}{u \cdot u} = y \cdot u \text{ projection.}
\]
Linear regression using SVD: find $\mathbf{w}$ for smallest $\|\mathbf{A} \mathbf{w} - \mathbf{y}\|_2$

- A vector $\mathbf{w}$ that is closest to target vector $\mathbf{y}$ along direction $\mathbf{u}$ is $\mathbf{w} = x^* \mathbf{v}$. Proof:

$$x^* = \arg\min_{x \in \mathbb{R}} \|\mathbf{y} - x \mathbf{u}\|_2 = \frac{\mathbf{y} \cdot \mathbf{u}}{\mathbf{u} \cdot \mathbf{u}} = \mathbf{y} \cdot \mathbf{u} \text{ projection.}$$

- Use SVD to find singular vectors $\mathbf{u}_i$ and find projections of $\mathbf{y}$ along each.
Linear regression using SVD: find \( \mathbf{w} \) for smallest \( \| \mathbf{A} \mathbf{w} - \mathbf{y} \|_2 \)

- A vector \( \mathbf{w} \) that is closest to target vector \( \mathbf{y} \) along direction \( \mathbf{u} \) is \( \mathbf{w} = x^* \mathbf{v} \). Proof:

\[
x^* = \arg\min_{x \in \mathbb{R}} \| \mathbf{y} - x \mathbf{u} \|_2 = \frac{\mathbf{y} \cdot \mathbf{u}}{\mathbf{u} \cdot \mathbf{u}} = \mathbf{y} \cdot \mathbf{u} \text{ projection.}
\]

- Use SVD to find singular vectors \( \mathbf{u}_i \) and find projections of \( \mathbf{y} \) along each.
- Reminder: SVD expressed as

\[
\begin{pmatrix}
\mathbf{A} \\
\end{pmatrix}
\begin{pmatrix}
\mathbf{v}_1 & \cdots & \mathbf{v}_n
\end{pmatrix}
=
\begin{pmatrix}
\mathbf{u}_1 & \cdots & \mathbf{u}_n
\end{pmatrix}
\begin{pmatrix}
\sigma_1 \\
\cdots \\
\sigma_n
\end{pmatrix}
\]
Linear regression by SVD: express weights and targets in terms of singular vectors

\[ A = U \Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T \] implies \[ Av_i = \sigma_i u_i. \]
Linear regression by SVD: express weights and targets in terms of singular vectors

- \( A = U \Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T \) implies \( Av_i = \sigma_i u_i \).
- Express \( y = \sum_k \beta_k u_k \) and \( w = \sum_i \alpha_i v_i \).
Linear regression by SVD: express weights and targets in terms of singular vectors

- \( A = U \Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T \) implies \( Av_i = \sigma_i u_i \).
- Express \( y = \sum_k \beta_k u_k \) and \( w = \sum_i \alpha_i v_i \).
- Find projection of \( y \) along \( u_k \) for closest vectors \( (u_k \cdot y) u_k \) to \( y \) along each direction \( u_k \).
Linear regression by SVD: express weights and targets in terms of singular vectors

- $\mathbf{A} = \mathbf{U} \Sigma \mathbf{V}^T = \sum_{k=1}^{r} \mathbf{u}_k \sigma_k \mathbf{v}_k^T$ implies $\mathbf{A} \mathbf{v}_i = \sigma_i \mathbf{u}_i$.
- Express $\mathbf{y} = \sum_k \beta_k \mathbf{u}_k$ and $\mathbf{w} = \sum_i \alpha_i \mathbf{v}_i$.
- Find projection of $\mathbf{y}$ along $\mathbf{u}_k$ for closest vectors $(\mathbf{u}_k \cdot \mathbf{y}) \mathbf{u}_k$ to $\mathbf{y}$ along each direction $\mathbf{u}_k$. 
Linear regression by SVD: express weights and targets in terms of singular vectors

- \( A = U\Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T \) implies \( Av_i = \sigma_i u_i \).
- Express \( y = \sum_k \beta_k u_k \) and \( w = \sum_i \alpha_i v_i \).
- Find projection of \( y \) along \( u_k \) for closest vectors \((u_k \cdot y)u_k\) to \( y \) along each direction \( u_k \). Choose \( \beta_k = (u_k^T y) \).
- The left hand side combines weighted features
  \[
  Aw = A\left(\sum_i \alpha_i v_i\right) = \sum_i \alpha_i (Av_i) = \sum_i \alpha_i \sigma_i u_i.
  \]
Linear regression by SVD: express weights and targets in terms of singular vectors

- $A = U \Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T$ implies $Av_i = \sigma_i u_i$.
- Express $y = \sum_k \beta_k u_k$ and $w = \sum_i \alpha_i v_i$.
- Find projection of $y$ along $u_k$ for closest vectors $(u_k \cdot y)u_k$ to $y$ along each direction $u_k$. Choose $\beta_k = (u_k^T y)$.
- The left hand side combines weighted features

$$Aw = A(\sum_i \alpha_i v_i) = \sum_i \alpha_i (Av_i) = \sum_i \alpha_i \sigma_i u_i.$$  

- The best fit vector to $y$ along each $u_i$ is $\beta_i u_i$. The vector in the column space of $A$ along direction $u_i$ is $\alpha_i \sigma_i u_i$. 
Linear regression by SVD: express weights and targets in terms of singular vectors

- $A = U \Sigma V^T = \sum_{k=1}^{r} u_k \sigma_k v_k^T$ implies $Av_i = \sigma_i u_i$.
- Express $y = \sum_k \beta_k u_k$ and $w = \sum_i \alpha_i v_i$.
- Find projection of $y$ along $u_k$ for closest vectors $(u_k \cdot y)u_k$ to $y$ along each direction $u_k$. Choose $\beta_k = (u_k^T y)$.
- The left hand side combines weighted features

$$Aw = A(\sum_i \alpha_i v_i) = \sum_i \alpha_i (Av_i) = \sum_i \alpha_i \sigma_i u_i.$$ 

- The best fit vector to $y$ along each $u_i$ is $\beta_i u_i$. The vector in the column space of $A$ along direction $u_i$ is $\alpha_i \sigma_i u_i$.
- The coefficients $\alpha_i$ of the optimal weight vector $w$ along each of the singular vectors $v_i$ are obtained from

$$\alpha_i \sigma_i = \beta_i = u_i^T y \implies \alpha_i = \frac{u_i^T y}{\sigma_i}.$$
Linear regression by SVD: small singular values are unwelcome

- The best fit weight vector is

\[
\mathbf{w} = \sum_i \left( \frac{\mathbf{u}_i^T \mathbf{y}}{\sigma_i} \right) \mathbf{v}_i.
\]
Linear regression by SVD: small singular values are unwelcome

- The best fit weight vector is

  $$w = \sum_i \left( \frac{u_i^T y}{\sigma_i} \right) v_i.$$  

- What is the relationship between this expression and

  $$w = \left( A^T A \right)^{-1} A^T y?$$
Linear regression by SVD: small singular values are unwelcome

- The best fit weight vector is
  \[ w = \sum_i \left( \frac{u_i^T y}{\sigma_i} \right) v_i. \]

- What is the relationship between this expression and
  \[ w = \left( A^T A \right)^{-1} A^T y? \]

- Verify
  \[ \left( A^T A \right)^{-1} A^T y = V \Sigma^{-1} U^T y. \]
Linear regression by SVD: small singular values are unwelcome

- The best fit weight vector is

\[ \mathbf{w} = \sum_i \left( \frac{\mathbf{u}_i^T \mathbf{y}}{\sigma_i} \right) \mathbf{v}_i. \]

- What is the relationship between this expression and

\[ \mathbf{w} = \left( \mathbf{A}^T \mathbf{A} \right)^{-1} \mathbf{A}^T \mathbf{y}? \]

- Verify

\[ \left( \mathbf{A}^T \mathbf{A} \right)^{-1} \mathbf{A}^T \mathbf{y} = \mathbf{V} \Sigma^{-1} \mathbf{U}^T \mathbf{y}. \]

- Very small (zero) singular values cause problems. The large (infinite) components of the weight vectors track noise in the targets, not useful signals. This leads to the subject of regularisation.
Relationship between singular vectors/values and eigen-vectors/values

Since the eigenvectors of a matrix can be used as a basis for a vector space, it will be important to show how these constructs are related.
Suppose $\mathbf{A}$ has $n$ linear independent eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. Its stacked column vectors

$$\mathbf{Q} = (\mathbf{v}_1 \mathbf{v}_2 \cdots \mathbf{v}_n)$$

give representation of the matrix $\mathbf{A}$

$$\mathbf{A} = \mathbf{Q} \Lambda \mathbf{Q}^{-1} \text{ if nonsingular } \mathbf{Q},$$

where $\Lambda$ is the diagonal matrix of eigenvalues $\text{diag}(\lambda_1, \ldots, \lambda_n)$ of $\mathbf{A}$. 
Suppose $A$ has $n$ linear independent eigenvectors $v_1, v_2, \ldots, v_n$. Its stacked column vectors

\[ Q = (v_1 v_2 \cdots v_n) \]

give representation of the matrix $A$

\[ A = Q \Lambda Q^{-1} \text{ if nonsingular } Q, \]

where $\Lambda$ is the diagonal matrix of eigenvalues $\text{diag}(\lambda_1, \ldots, \lambda_n)$ of $A$.

**Proof:** The eigenvalue equations for the $v_i$ can be written as $A Q = Q \Lambda$. Multiplying by $Q^{-1}$ from the right gives the result.
Relationship between SVD and eigen-analysis

- Representation of $\mathbf{A}$ (real symmetric) $\mathbf{A} = \mathbf{Q} \Lambda \mathbf{Q}^T$.
- For SVD of $\mathbf{X} = \mathbf{U} \Sigma \mathbf{V}^T$,

$$\mathbf{X} \mathbf{X}^T = (\mathbf{U} \Sigma \mathbf{V}^T)(\mathbf{U} \Sigma \mathbf{V}^T)^T =$$
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U\Sigma V^T$,

$$XX^T = (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)((V^T)^T\Sigma^T U^T)$$
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U\Sigma V^T$, 

$$XX^T = (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)((V^T)^T \Sigma^T U^T) = (U\Sigma(V^TV)\Sigma^T U^T) = \ldots$$
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U \Sigma V^T$, 

$$XX^T = (U \Sigma V^T)(U \Sigma V^T)^T = (U \Sigma V^T)((V^T)^T \Sigma^T U^T) = (U \Sigma (V^T V) \Sigma^T U^T) = U(\Sigma \Sigma^T)U^T$$
Relationship between SVD and eigen-analysis

- Representation of $\mathbf{A}$ (real symmetric) $\mathbf{A} = \mathbf{Q} \mathbf{Q}^T$.
- For SVD of $\mathbf{X} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$,

\[
\mathbf{X} \mathbf{X}^T = (\mathbf{U} \mathbf{\Sigma} \mathbf{V}^T)(\mathbf{U} \mathbf{\Sigma} \mathbf{V}^T)^T = (\mathbf{U} \mathbf{\Sigma} \mathbf{V}^T)((\mathbf{V}^T)^T \mathbf{\Sigma}^T \mathbf{U}^T)
= (\mathbf{U} \mathbf{\Sigma} (\mathbf{V}^T \mathbf{V}) \mathbf{\Sigma}^T \mathbf{U}^T)
= \mathbf{U} (\mathbf{\Sigma} \mathbf{\Sigma}^T) \mathbf{U}^T
\]

\[
\mathbf{X}^T \mathbf{X} = (\mathbf{U} \mathbf{\Sigma} \mathbf{V}^T)^T (\mathbf{U} \mathbf{\Sigma} \mathbf{V}^T) =
\]
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U\Sigma V^T$,

$$XX^T = (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)((V^T)^T \Sigma^T U^T) = (U\Sigma(V^T V)\Sigma^T U^T) = U(\Sigma\Sigma^T)U^T$$

$$X^TX = (U\Sigma V^T)^T(U\Sigma V^T) = ((V^T)^T \Sigma^T U^T)(U\Sigma V^T)$$
Relationship between SVD and eigen-analysis

- Representation of \( A \) (real symmetric) \( A = Q \Lambda Q^T \).
- For SVD of \( X = U\Sigma V^T \),

\[
XX^T = (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)((V^T)^T \Sigma^T U^T) = (U\Sigma(V^T V)\Sigma^T U^T) = U(\Sigma \Sigma^T)U^T
\]

\[
X^T X = (U\Sigma V^T)^T(U\Sigma V^T) = ((V^T)^T \Sigma^T U^T)(U\Sigma V^T) = (V\Sigma^T(U^T U)\Sigma V^T)
\]
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U \Sigma V^T$,
  \[
  XX^T = (U \Sigma V^T)(U \Sigma V^T)^T = (U \Sigma V^T)((V^T)^T \Sigma^T U^T)
  = (U \Sigma (V^T V) \Sigma^T U^T)
  = U(\Sigma \Sigma^T)U^T
  
  X^T X = (U \Sigma V^T)^T (U \Sigma V^T) = ((V^T)^T \Sigma^T U^T)(U \Sigma V^T)
  = (V \Sigma^T (U^T U) \Sigma V^T)
  = V(\Sigma^T \Sigma) V^T
  \]
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U \Sigma V^T$,

$$XX^T = (U \Sigma V^T)(U \Sigma V^T)^T = (U \Sigma V^T)((V^T)^T \Sigma^T U^T)$$
$$= (U \Sigma (V^T V) \Sigma^T U^T)$$
$$= U(\Sigma \Sigma^T)U^T$$

$$X^T X = (U \Sigma V^T)^T (U \Sigma V^T) = ((V^T)^T \Sigma^T U^T)(U \Sigma V^T)$$
$$= (V \Sigma^T (U^T U) \Sigma V^T)$$
$$= V(\Sigma^T \Sigma)V^T$$

- Right singular vectors of $X$ are eigenvectors of $X^T X$;
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U\Sigma V^T$,

$$XX^T = (U\Sigma V^T)(U\Sigma V^T)^T = (U\Sigma V^T)((V^T)^T\Sigma^T U^T)$$
$$= (U\Sigma(V^T V)\Sigma^T U^T)$$
$$= U(\Sigma\Sigma^T)U^T$$

$$X^TX = (U\Sigma V^T)^T(U\Sigma V^T) = ((V^T)^T\Sigma^T U^T)(U\Sigma V^T)$$
$$= (V\Sigma^T(U^T U)\Sigma V^T)$$
$$= V(\Sigma^T\Sigma)V^T$$

- Right singular vectors of $X$ are eigenvectors of $X^TX$; left singular vectors of $X$ are eigenvectors of $XX^T$. 
Relationship between SVD and eigen-analysis

- Representation of $A$ (real symmetric) $A = Q \Lambda Q^T$.
- For SVD of $X = U \Sigma V^T$,

\[
XX^T = (U \Sigma V^T)(U \Sigma V^T)^T = (U \Sigma V^T)((V^T)^T \Sigma^T U^T)
\]
\[
= (U \Sigma (V^T V) \Sigma^T U^T)
\]
\[
= U(\Sigma \Sigma^T)U^T
\]

\[
X^TX = (U \Sigma V^T)^T(U \Sigma V^T) = ((V^T)^T \Sigma^T U^T)(U \Sigma V^T)
\]
\[
= (V \Sigma^T(U^T U) \Sigma V^T)
\]
\[
= V(\Sigma^T \Sigma)V^T
\]

- Right singular vectors of $X$ are eigenvectors of $X^TX$; left singular vectors of $X$ are eigenvectors of $XX^T$. Eigenvalues are $\sigma_i^2$ where $\sigma_i = \Sigma_{ii}$.
If $A$ is a real symmetric matrix, we can construct $Q$ from the $n$ orthonormal eigenvectors $\hat{v}_i$ (i.e. the eigenvectors must also be normalized to unit length) as $Q = (\hat{v}_1 \hat{v}_2 \cdots \hat{v}_n)$. We can show that $Q$ is an orthogonal matrix i.e.

$$Q^{-1} = Q^T.$$
If $\mathbf{A}$ is a real symmetric matrix, we can construct $\mathbf{Q}$ from the $n$ orthonormal eigenvectors $\hat{\mathbf{v}}_i$ (i.e. the eigenvectors must also be normalized to unit length) as $\mathbf{Q} = (\hat{\mathbf{v}}_1 \hat{\mathbf{v}}_2 \cdots \hat{\mathbf{v}}_n)$. We can show that $\mathbf{Q}$ is an orthogonal matrix, i.e.

$$
\mathbf{Q}^{-1} = \mathbf{Q}^T.
$$

This is easily proved from the fact that $\hat{\mathbf{v}}_i \cdot \hat{\mathbf{v}}_j = 0$ for $i \neq j$ and $|\hat{\mathbf{v}}_i| = 1$, which can be written as $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$. Hence, we get

$$
\mathbf{A} = \mathbf{Q} \mathbf{Q}^T.
$$
Summary SVD/eigenvalues/vectors

- SVD for matrix $X = U \Sigma V^T$; columns of $U$, $V$ orthonormal, $\Sigma$ has only diagonal entries non-zero $\sigma_i = \Sigma_{ii}$ (singular values).
- Definition: For a square matrix $A$, find nontrivial vectors $v$ (eigenvectors) such that matrix multiplication behaves like scalar multiplication: $Av = \lambda v$ for scalars (eigenvalues) $\lambda$.
- For **real symmetric matrices** $n \times n$ matrices, eigenvalues $\lambda$ are real numbers and we can always find $n$ orthogonal eigenvectors $v_i$, for $i = 1, \ldots, n$. This means that $v_i \cdot v_j = 0$ for $j \neq i$.
- Representation of $A$ (real symmetric)
  \[ A = Q \Lambda Q^T. \]
  where $Q = (\hat{v}_1 \hat{v}_2 \cdots \hat{v}_n)$ and $\Lambda$ a diagonal matrix containing the eigenvalues.
- Right singular vectors of $X$ are eigenvectors of $X^T X$. 
Summary SVD/eigenvalues/vectors

- SVD for matrix $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$; columns of $\mathbf{U}$, $\mathbf{V}$ orthonormal, $\mathbf{\Sigma}$ has only diagonal entries non-zero $\sigma_i = \Sigma_{ii}$ (singular values).

- Definition: For a square matrix $\mathbf{A}$, find nontrivial vectors $\mathbf{v}$ (eigenvectors) such that matrix multiplication behaves like scalar multiplication: $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$ for scalars (eigenvalues) $\lambda$.

- For real symmetric matrices $n \times n$ matrices, eigenvalues $\lambda$ are real numbers and we can always find $n$ orthogonal eigenvectors $\mathbf{v}_i$, for $i = 1, \ldots, n$. This means that $\mathbf{v}_i \cdot \mathbf{v}_j = 0$ for $j \neq i$.

- Representation of $\mathbf{A}$ (real symmetric)
  \[ \mathbf{A} = \mathbf{Q} \Lambda \mathbf{Q}^T. \]
  where $\mathbf{Q} = (\hat{\mathbf{v}}_1 \hat{\mathbf{v}}_2 \cdots \hat{\mathbf{v}}_n)$ and $\Lambda$ a diagonal matrix containing the eigenvalues.

- Right singular vectors of $\mathbf{X}$ are eigenvectors of $\mathbf{X}^T\mathbf{X}$, left singular vectors of $\mathbf{X}$ are eigenvectors of $\mathbf{XX}^T$. 
Summary SVD/eigenvalues/vectors

- SVD for matrix $X = U\Sigma V^T$; columns of $U$, $V$ orthonormal, $\Sigma$ has only diagonal entries non-zero $\sigma_i = \Sigma_{ii}$ (singular values).
- Definition: For a square matrix $A$, find nontrivial vectors $v$ (eigenvectors) such that matrix multiplication behaves like scalar multiplication: $Av = \lambda v$ for scalars (eigenvalues) $\lambda$.
- For real symmetric matrices $n \times n$ matrices, eigenvalues $\lambda$ are real numbers and we can always find $n$ orthogonal eigenvectors $v_i$, for $i = 1, \ldots, n$. This means that $v_i \cdot v_j = 0$ for $j \neq i$.
- Representation of $A$ (real symmetric)
  $$A = Q \Lambda \Lambda^T.$$
  where $Q = (\hat{v}_1 \hat{v}_2 \cdots \hat{v}_n)$ and $\Lambda$ a diagonal matrix containing the eigenvalues.
- Right singular vectors of $X$ are eigenvectors of $X^T X$, left singular vectors of $X$ are eigenvectors of $XX^T$, with eigenvalues $\sigma_i^2$. 
