Advanced Machine Learning

Vector Spaces

$$\mathbf{Mx=b}$$

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 16 \\ 1 & 3 & 9 & 27 & 81 \\ 1 & 4 & 16 & 64 & 256 \\ 1 & 5 & 25 & 125 & 625 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 2 \\ -3 \\ 1 \end{pmatrix}$$

$$\mathbf{b=M^{-1}x}$$

$$\nabla_X \mathrm{Tr}(X^{-1}A) = -X^{-1}AX^{-1}$$

$$\mathbf{Mv}_i = \lambda_i \mathbf{v}_i$$

Vectors, metric spaces, norms

Outline

1. **Vector Spaces**
2. Metrics (distances)
3. Norms

$\mathbf{Mx=b}$

$\mathbf{Mv}_i=\lambda_i\mathbf{v}_i$

$\mathbf{b=M}$

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know

Matrices, Vectors and All That

- The language of machine learning is mathematics
- Sometimes we draw pretty pictures to explain the mathematics
- Much of the mathematics we will use involves vectors, matrices and functions
- You need to master the language of mathematics, otherwise you won't understand the algorithms
- I'm going to spend this lecture and the next revising the mathematics you need to know (but I'm going use a slightly posher language than you are probably used to)

Scalars (Fields)

- Vector spaces involve **fields** (numbers)

- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**

- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a+b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication and they are distributive

$$a \times (b + c) = a \times b + a \times c$$

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication and they are distributive

$$a \times (b + c) = a \times b + a \times c$$

- Although this sounds rather daunting don't panic

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a+b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication and they are distributive

$$a \times (b+c) = a \times b + a \times c$$

- Although this sounds rather daunting don't panic. They behave like numbers.

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication and they are distributive

$$a \times (b + c) = a \times b + a \times c$$

- Although this sounds rather daunting don't panic. They behave like numbers. The field might be integers, rational numbers, reals, complex numbers or something a bit more exotic

Scalars (Fields)

- Vector spaces involve **fields** (numbers)—aka **scalars**
- These are quantities we can add together $(a + b)$ and multiply together $(a \times b)$
- Formally they form an Abelian group under addition with an identity 0 and excluding 0 an Abeilian group under multiplication and they are distributive

$$a \times (b + c) = a \times b + a \times c$$

- Although this sounds rather daunting don't panic. They behave like numbers.The field might be integers, rational numbers, reals, complex numbers or something a bit more exotic—but we will almost always consider reals

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Vectors

- We often work with objects with many components (features)
- To help handle this we will use vector notation

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

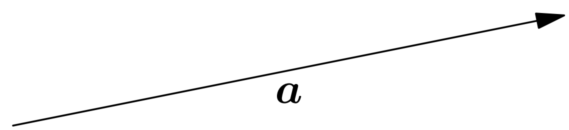
- We represent vectors by bold symbols
- All our vectors are column vectors by default
- We treat them as $n \times 1$ matrix

- We write row vectors as transposes of column vectors

$$\boldsymbol{y}^{\mathsf{T}} = (y_1, y_2, \ldots, y_n)$$

Basic Vector Operations

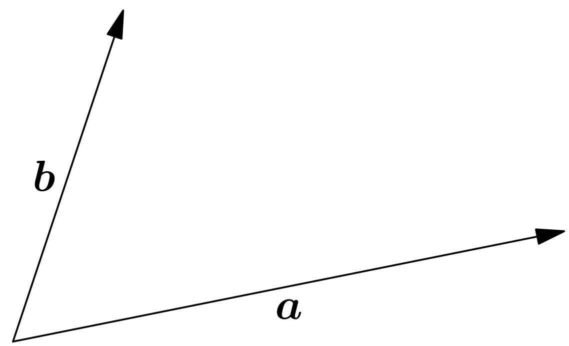
- The basic vector operations are adding



- multiplying by a scalar (a number)

Basic Vector Operations

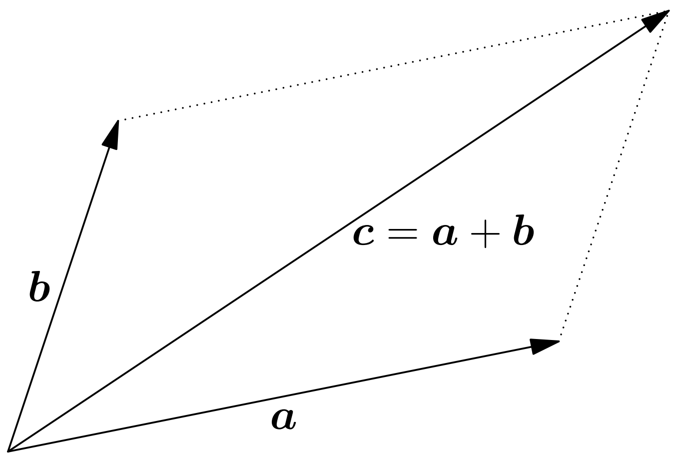
- The basic vector operations are adding



- multiplying by a scalar (a number)

Basic Vector Operations

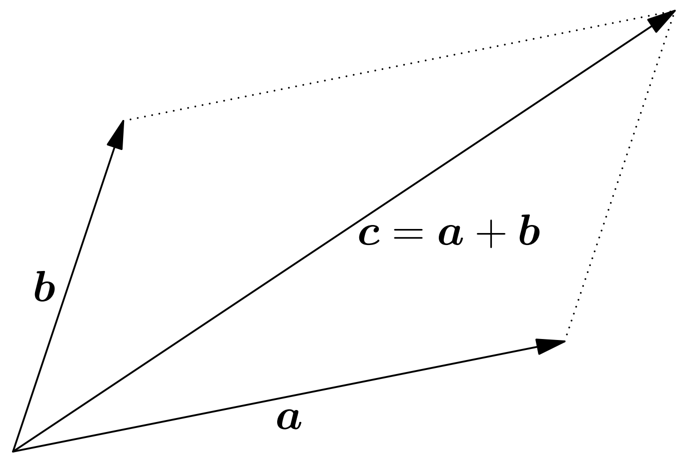
- The basic vector operations are adding



- multiplying by a scalar (a number)

Basic Vector Operations

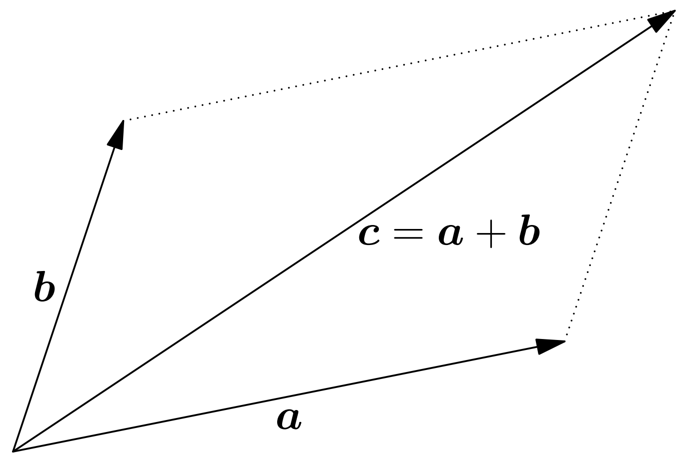
- The basic vector operations are adding



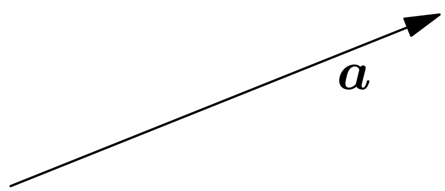
- multiplying by a scalar (a number)

Basic Vector Operations

- The basic vector operations are adding

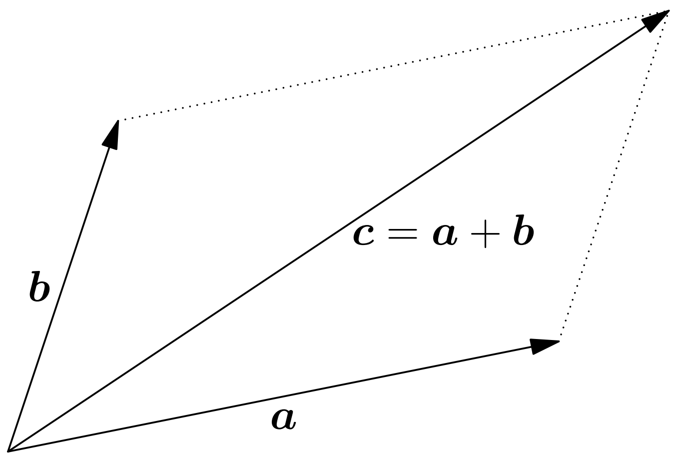


- multiplying by a scalar (a number)

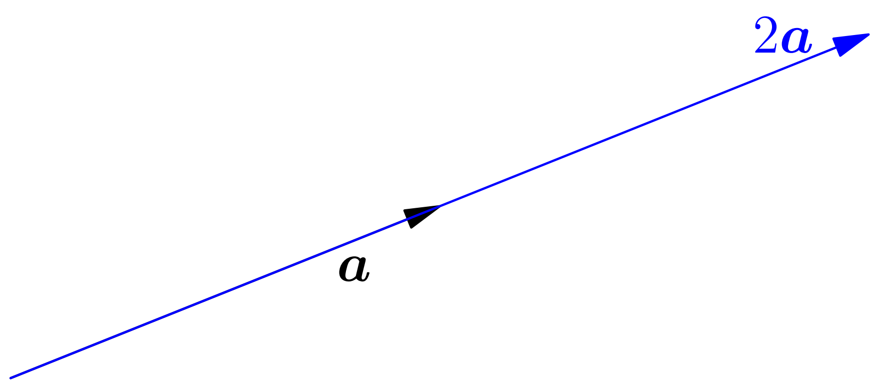


Basic Vector Operations

- The basic vector operations are adding



- multiplying by a scalar (a number)



Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

  1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
  2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
  3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
  4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
  5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
  6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
  7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
  8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

  (You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

(You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

(You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

  1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
  2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
  3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
  4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
  5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
  6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
  7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
  8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

  (You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

  1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
  2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
  3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
  4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
  5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
  6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
  7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
  8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

  (You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
4. $\boldsymbol{v} + \boldsymbol{0} = \boldsymbol{v}$ (existence of additive identity $\boldsymbol{0}$)
5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

(You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

  1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
  2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
  3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
  4. $\boldsymbol{v} + \mathbf{0} = \boldsymbol{v}$ (existence of additive identity $\mathbf{0}$)
  5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
  6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
  7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
  8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

  (You don't need to remember these)

- Just from these properties we can deduce other properties

Vector Space

- A vector space, $\mathcal{V}$, is a set of vectors which satisfies

  1. if $\boldsymbol{v}, \boldsymbol{w} \in \mathcal{V}$ then $a\boldsymbol{v} \in \mathcal{V}$ and $\boldsymbol{v} + \boldsymbol{w} \in \mathcal{V}$ (closure)
  2. $\boldsymbol{v} + \boldsymbol{w} = \boldsymbol{w} + \boldsymbol{v}$ (commutativity of addition)
  3. $(\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w} = \boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w})$ (associativity of addition)
  4. $\boldsymbol{v} + \mathbf{0} = \boldsymbol{v}$ (existence of additive identity $\mathbf{0}$)
  5. $1\boldsymbol{v} = \boldsymbol{v}$ (existence of multiplicative identity 1)
  6. $a(b\boldsymbol{v}) = (ab)\boldsymbol{v}$ (distributive properties)
  7. $a(\boldsymbol{v} + \boldsymbol{w}) = a\boldsymbol{v} + a\boldsymbol{w}$
  8. $(a + b)\boldsymbol{v} = a\boldsymbol{v} + b\boldsymbol{v}$

  (You don't need to remember these)

- Just from these properties we can deduce other properties

$\mathbb{R}^n$

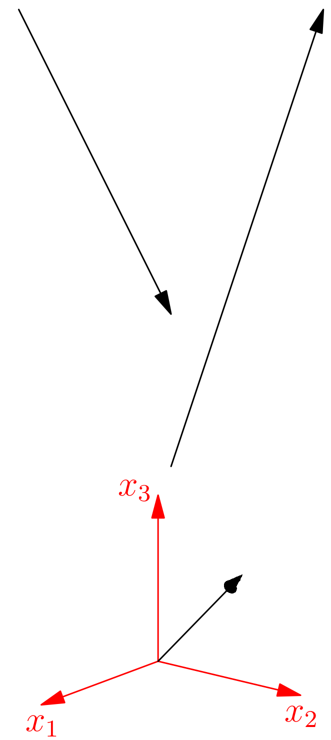
- When we first learn about vectors we think of them arrows in 3-D space

- If we centre them all at the origin then there is a one-to-one correspondence between vectors and points in space

- We call this vector space $\mathbb{R}^3$

- Any set of quantities $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^\mathsf{T}$ which satisfy the axioms above form a vector space $\mathbb{R}^n$

- Of course, we can't so easily draw pictures of high-dimensional vectors



$\mathbb{R}^n$

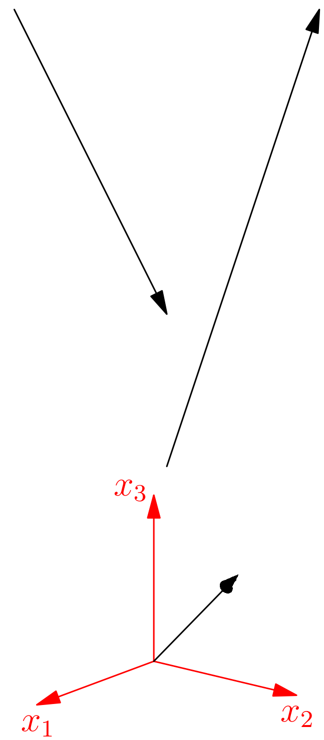
- When we first learn about vectors we think of them arrows in 3-D space

- If we centre them all at the origin then there is a one-to-one correspondence between vectors and points in space

- We call this vector space $\mathbb{R}^3$

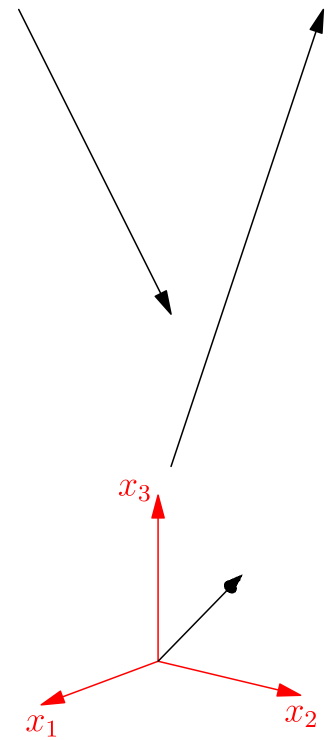
- Any set of quantities $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^\mathsf{T}$ which satisfy the axioms above form a vector space $\mathbb{R}^n$

- Of course, we can't so easily draw pictures of high-dimensional vectors



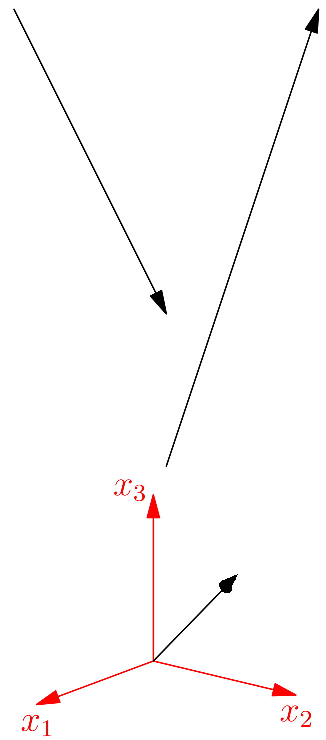
$\mathbb{R}^n$

- When we first learn about vectors we think of them arrows in 3-D space
- If we centre them all at the origin then there is a one-to-one correspondence between vectors and points in space
- We call this vector space $\mathbb{R}^3$
- Any set of quantities $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^\mathsf{T}$ which satisfy the axioms above form a vector space $\mathbb{R}^n$
- Of course, we can't so easily draw pictures of high-dimensional vectors



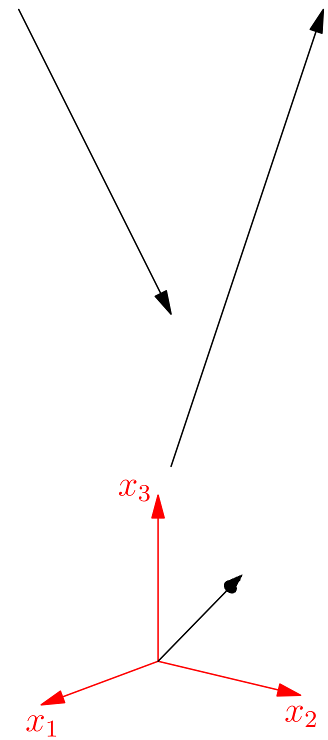
$\mathbb{R}^n$

- When we first learn about vectors we think of them arrows in 3-D space
- If we centre them all at the origin then there is a one-to-one correspondence between vectors and points in space
- We call this vector space $\mathbb{R}^3$
- Any set of quantities $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^\mathsf{T}$ which satisfy the axioms above form a vector space $\mathbb{R}^n$
- Of course, we can't so easily draw pictures of high-dimensional vectors



$\mathbb{R}^n$

- When we first learn about vectors we think of them arrows in 3-D space
- If we centre them all at the origin then there is a one-to-one correspondence between vectors and points in space
- We call this vector space $\mathbb{R}^3$
- Any set of quantities $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^\mathsf{T}$ which satisfy the axioms above form a vector space $\mathbb{R}^n$
- Of course, we can't so easily draw pictures of high-dimensional vectors



Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\,f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\,f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\,f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\,f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\,f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Other Vector Spaces

- Vectors (i.e. $\mathbb{R}^n$) are not the only object that form vector spaces
- Matrices satisfy all the conditions of a vector space
- Infinite sequences form a vector space
- Functions form a vector space
  - ⋆ Let $C(a,b)$ be the set of functions defined on the interval $[a,b]$
  - ⋆ Note that if $f(x), g(x) \in C(a,b)$ then $a\, f(x) \in C(a,b)$ and $f(x) + g(x) \in C(a,b)$
- Bounded vectors **don't** form a vector space

Outline

1. Vector Spaces
2. **Metrics (distances)**
3. Norms

$\mathbf{Mx}=\mathbf{b}$

$\mathbf{Mv}_i=\lambda_i\mathbf{v}_i$

$\mathbf{b}=\mathbf{M}$

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance
- We say $d(\boldsymbol{x}, \boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x}, \boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x}, \boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x}, \boldsymbol{y}) \leq d(\boldsymbol{x}, \boldsymbol{z}) + d(\boldsymbol{z}, \boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)
- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Metrics

- Vector spaces become more interesting if we have a notion of distance

- We say $d(\boldsymbol{x},\boldsymbol{y})$ is a **proper distance** or **metric** if

  1. $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$ (non-negativity)
  2. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ iff $\boldsymbol{x} = \boldsymbol{y}$ (identity of indiscernibles)
  3. $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$ (symmetry)
  4. $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$ (triangular inequality)

- There are typically many possible distances (e.g. Euclidean distance, Manhattan distance, etc.)

- Often one or more condition isn't satisfied then we have a **pseudo-metric**

Mappings and Functions

- A function defines a mapping from one vector space to another (although the spaces might be the same),

Mappings and Functions

- A function defines a mapping from one vector space to another (although the spaces might be the same),e.g.

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

Mappings and Functions

- A function defines a mapping from one vector space to another (although the spaces might be the same),e.g.

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

(f maps the reals onto reals, i.e. $f(x)$ takes a real x and gives you a new real number $y = f(x)$)

Mappings and Functions

- A function defines a mapping from one vector space to another (although the spaces might be the same),e.g.

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

  (f maps the reals onto reals, i.e. $f(x)$ takes a real x and gives you a new real number $y = f(x)$)

- We are often interested in functions that behave nicely

Mappings and Functions

- A function defines a mapping from one vector space to another (although the spaces might be the same),e.g.

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

  (f maps the reals onto reals, i.e. $f(x)$ takes a real x and gives you a new real number $y = f(x)$)

- We are often interested in functions that behave nicely

- E.g. They are continuous

Lipschitz Function

- One way to characterise well behaved function, $f(x)$ is if there exists a number $K < \infty$ such that for all x and y

$$d(f(x), f(y)) \leq K \, d(x, y)$$

- This is known a **Lipschitz condition** and the function is said to be K-Lipschitz
- Note that such functions cannot have any jumps (i.e. they are continuous)
- The size of K measures the limit on the amplifying effect of the function

Lipschitz Function

- One way to characterise well behaved function, $f(x)$ is if there exists a number $K < \infty$ such that for all x and y

$$d(f(x), f(y)) \leq K\, d(x, y)$$

- This is known a **Lipschitz condition** and the function is said to be K-Lipschitz

- Note that such functions cannot have any jumps (i.e. they are continuous)

- The size of K measures the limit on the amplifying effect of the function

Lipschitz Function

- One way to characterise well behaved function, $f(x)$ is if there exists a number $K < \infty$ such that for all x and y

$$d(f(x), f(y)) \leq K\, d(x, y)$$

- This is known a **Lipschitz condition** and the function is said to be K-Lipschitz

- Note that such functions cannot have any jumps (i.e. they are continuous)

- The size of K measures the limit on the amplifying effect of the function

Lipschitz Function

- One way to characterise well behaved function, $f(x)$ is if there exists a number $K < \infty$ such that for all x and y

$$d(f(x), f(y)) \le K\, d(x, y)$$

- This is known a **Lipschitz condition** and the function is said to be K-Lipschitz
- Note that such functions cannot have any jumps (i.e. they are continuous)
- The size of K measures the limit on the amplifying effect of the function

Contractive Mappings

- An interesting class of function are those for which $K < 1$
- These are said to be contractive mappings
- A famous theorem that applies to contractive mappings is the Banach fixed-point theorem which says there exists a unique fixed point such that $f(x) = x$
- This is used for example in showing that various algorithms will converge

Contractive Mappings

- An interesting class of function are those for which $K < 1$
- These are said to be contractive mappings
- A famous theorem that applies to contractive mappings is the Banach fixed-point theorem which says there exists a unique fixed point such that $f(x) = x$
- This is used for example in showing that various algorithms will converge

Contractive Mappings

- An interesting class of function are those for which $K < 1$
- These are said to be contractive mappings
- A famous theorem that applies to contractive mappings is the Banach fixed-point theorem which says there exists a unique fixed point such that $f(x) = x$
- This is used for example in showing that various algorithms will converge

Contractive Mappings

- An interesting class of function are those for which $K < 1$
- These are said to be contractive mappings
- A famous theorem that applies to contractive mappings is the Banach fixed-point theorem which says there exists a unique fixed point such that $f(x) = x$
- This is used for example in showing that various algorithms will converge

Outline

$\mathbf{Mx=b}$

1. Vector Spaces
2. Metrics (distances)
3. **Norms**

$\mathbf{b=M}$

$\mathbf{Mv}_i = \lambda_i \mathbf{v}_i$

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \mathbf{0}$ (non-negativity)
2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
3. $\|\boldsymbol{u}+\boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

  1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \boldsymbol{0}$ (non-negativity)
  2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
  3. $\|\boldsymbol{u}+\boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

  1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \mathbf{0}$ (non-negativity)
  2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
  3. $\|\boldsymbol{u}+\boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x}-\boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \mathbf{0}$ (non-negativity)
2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
3. $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \boldsymbol{0}$ (non-negativity)
2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
3. $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \boldsymbol{0}$ (non-negativity)
2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
3. $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$ (they are metric spaces)

Norms

- Vector spaces are even more interesting with a notion of length
- **Norms** provide some measure of the size of a vector
- To formalise this we define the **norm** of an object $\boldsymbol{v}$ as $\|\boldsymbol{v}\|$ satisfying

  1. $\|\boldsymbol{v}\| > 0$ if $\boldsymbol{v} \neq \mathbf{0}$ (non-negativity)
  2. $\|a\boldsymbol{v}\| = a\|\boldsymbol{v}\|$ (linearity)
  3. $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (triangular inequality)

- When some criteria aren't satisfied we have a **pseudo-norms**
- Norms provide a metric $d(\boldsymbol{x},\boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$ (they are metric spaces)

Vector Norms

- The familiar vector norm is the (Euclidean) two norm

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- Other norms exist, such as the p-norm $(p \geq 1)$

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

- Special cases include the 1-norm and the infinite norm

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} |v_i| \qquad\qquad \|\boldsymbol{v}\|_\infty = \max_i |v_i|$$

- The 0-norm is a pseudo-norm as it does not satisfy condition 2

$$\|\boldsymbol{v}\|_0 = \text{number of non-zero components}$$

Vector Norms

- The familiar vector norm is the (Euclidean) two norm

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- Other norms exist, such as the p-norm $(p \geq 1)$

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

- Special cases include the 1-norm and the infinite norm

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} |v_i| \qquad\qquad \|\boldsymbol{v}\|_\infty = \max_i |v_i|$$

- The 0-norm is a pseudo-norm as it does not satisfy condition 2

$$\|\boldsymbol{v}\|_0 = \text{number of non-zero components}$$

Vector Norms

- The familiar vector norm is the (Euclidean) two norm

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- Other norms exist, such as the p-norm $(p \geq 1)$

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

- Special cases include the 1-norm and the infinite norm

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} |v_i| \qquad\qquad \|\boldsymbol{v}\|_\infty = \max_i |v_i|$$

- The 0-norm is a pseudo-norm as it does not satisfy condition 2

$$\|\boldsymbol{v}\|_0 = \text{number of non-zero components}$$

Vector Norms

- The familiar vector norm is the (Euclidean) two norm

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- Other norms exist, such as the p-norm $(p \geq 1)$

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

- Special cases include the 1-norm and the infinite norm

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} |v_i| \qquad\qquad \|\boldsymbol{v}\|_\infty = \max_i |v_i|$$

- The 0-norm is a pseudo-norm as it does not satisfy condition 2

$$\|\boldsymbol{v}\|_0 = \text{number of non-zero components}$$

Vector Norms

- The familiar vector norm is the (Euclidean) two norm

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- Other norms exist, such as the p-norm $(p \geq 1)$

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

- Special cases include the 1-norm and the infinite norm

$$\|\boldsymbol{v}\|_1 = \sum_{i=1}^{n} |v_i| \qquad\qquad \|\boldsymbol{v}\|_\infty = \max_i |v_i|$$

- The 0-norm is a pseudo-norm as it does not satisfy condition 2

$$\|\boldsymbol{v}\|_0 = \text{number of non-zero components}$$

Matrix Norms

- We can define norms for other objects
- The norm of a matrix encodes how large the mapping is
- The Frobenius norm is defined by

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |A_{ij}|^2}$$

- Many other norms exist including 1-norm, max-norm, etc.
- For square matrices, some, but not all, norms satisfy the inequality

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \times \|\mathbf{B}\|$$

Matrix Norms

- We can define norms for other objects
- The norm of a matrix encodes how large the mapping is
- The Frobenius norm is defined by

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |A_{ij}|^2}$$

- Many other norms exist including 1-norm, max-norm, etc.
- For square matrices, some, but not all, norms satisfy the inequality

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \times \|\mathbf{B}\|$$

Matrix Norms

- We can define norms for other objects
- The norm of a matrix encodes how large the mapping is
- The Frobenius norm is defined by

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}|^2}$$

- Many other norms exist including 1-norm, max-norm, etc.
- For square matrices, some, but not all, norms satisfy the inequality

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \times \|\mathbf{B}\|$$

Matrix Norms

- We can define norms for other objects
- The norm of a matrix encodes how large the mapping is
- The Frobenius norm is defined by

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}|^2}$$

- Many other norms exist including 1-norm, max-norm, etc.
- For square matrices, some, but not all, norms satisfy the inequality

$$\|\mathbf{A}\mathbf{B}\| \leq \|\mathbf{A}\| \times \|\mathbf{B}\|$$

Matrix Norms

- We can define norms for other objects
- The norm of a matrix encodes how large the mapping is
- The Frobenius norm is defined by

$$\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}|^2}$$

- Many other norms exist including 1-norm, max-norm, etc.
- For square matrices, some, but not all, norms satisfy the inequality

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\| \times \|\mathbf{B}\|$$

Compatible Norms

- A vector and matrix norm are said to be compatible if

$$\|\boldsymbol{M}\boldsymbol{v}\|_b \leq \|\boldsymbol{M}\|_a \times \|\boldsymbol{v}\|_b$$

  (Spectral and Euclidean norms are compatible)

- Norms provide quick ways to bound the maximum growth of a vector under a mapping induced by the matrix

- We will see that a measure of the sensitivity of a mapping is in terms of the ratio of its maximum effect to its minimum effect on a vector

- This is known as the **conditioning**, given by $\|\mathbf{M}\| \times \|\mathbf{M}^{-1}\|$

Compatible Norms

- A vector and matrix norm are said to be compatible if

$$\|\mathbf{M}\boldsymbol{v}\|_b \leq \|\mathbf{M}\|_a \times \|\boldsymbol{v}\|_b$$

  (Spectral and Euclidean norms are compatible)

- Norms provide quick ways to bound the maximum growth of a vector under a mapping induced by the matrix

- We will see that a measure of the sensitivity of a mapping is in terms of the ratio of its maximum effect to its minimum effect on a vector

- This is known as the **conditioning**, given by $\|\mathbf{M}\| \times \|\mathbf{M}^{-1}\|$

Compatible Norms

- A vector and matrix norm are said to be compatible if

$$\|\boldsymbol{M}\boldsymbol{v}\|_b \leq \|\boldsymbol{M}\|_a \times \|\boldsymbol{v}\|_b$$

  (Spectral and Euclidean norms are compatible)

- Norms provide quick ways to bound the maximum growth of a vector under a mapping induced by the matrix

- We will see that a measure of the sensitivity of a mapping is in terms of the ratio of its maximum effect to its minimum effect on a vector

- This is known as the **conditioning**, given by $\|\mathbf{M}\| \times \|\mathbf{M}^{-1}\|$

Compatible Norms

- A vector and matrix norm are said to be compatible if

$$\|\boldsymbol{M}\boldsymbol{v}\|_b \leq \|\boldsymbol{M}\|_a \times \|\boldsymbol{v}\|_b$$

  (Spectral and Euclidean norms are compatible)

- Norms provide quick ways to bound the maximum growth of a vector under a mapping induced by the matrix

- We will see that a measure of the sensitivity of a mapping is in terms of the ratio of its maximum effect to its minimum effect on a vector

- This is known as the **conditioning**, given by $\|\mathbf{M}\| \times \|\mathbf{M}^{-1}\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers
- In CNNs we have convolutional layers and dense layers
- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$
- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation
- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish
- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers
- In CNNs we have convolutional layers and dense layers
- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$
- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation
- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish
- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers
- In CNNs we have convolutional layers and dense layers
- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$
- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation
- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish
- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers

- In CNNs we have convolutional layers and dense layers

- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$

- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation

- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish

- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers

- In CNNs we have convolutional layers and dense layers

- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$

- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation

- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish

- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Why Should You Care?

- Deep learning involves multiply the input (which we can think of as a vector $\boldsymbol{x}$) by many layers

- In CNNs we have convolutional layers and dense layers

- The effect of applying these layers can be represented by a matrix multiplication $\boldsymbol{x}_n = \mathbf{L}_n \boldsymbol{x}_{n-1}$

- We also do other things like applying ReLU's or pooling that changes the magnitude, $\boldsymbol{x}_n$, of our representation

- If you are developing new architectures you want $\|\boldsymbol{x}_n\|$ neither to blow up or vanish

- This can be controlled by carefully choosing $\|\mathbf{L}_n\|$

Function Norms

- Functions can also have norms, for example, if $f(x)$ is defined in some interval $\mathcal{I}$

$$\|f\|_{L_2} = \sqrt{\int_{x \in \mathcal{I}} f^2(x) \, \mathrm{d}x}$$

- The L_2 vector space is the set of function where $\|f\|_{L_2} < \infty$
- The L_1-norm is given by $\|f\|_{L_1} = \int_{x \in \mathcal{I}} |f(x)| \, \mathrm{d}x$
- The infinite-norm is given by $\|f\|_\infty = \max_{x \in \mathcal{I}} |f(x)|$

Function Norms

- Functions can also have norms, for example, if $f(x)$ is defined in some interval $\mathcal{I}$

$$\|f\|_{L_2} = \sqrt{\int_{x\in\mathcal{I}} f^2(x)\,\mathrm{d}x}$$

- The L_2 vector space is the set of function where $\|f\|_{L_2} < \infty$

- The L_1-norm is given by $\|f\|_{L_1} = \int_{x\in\mathcal{I}} |f(x)|\,\mathrm{d}x$

- The infinite-norm is given by $\|f\|_{\infty} = \max_{x\in\mathcal{I}} |f(x)|$

Function Norms

- Functions can also have norms, for example, if $f(x)$ is defined in some interval $\mathcal{I}$

$$\|f\|_{L_2} = \sqrt{\int_{x\in\mathcal{I}} f^2(x)\,\mathrm{d}x}$$

- The L_2 vector space is the set of function where $\|f\|_{L_2} < \infty$

- The L_1-norm is given by $\|f\|_{L_1} = \int_{x\in\mathcal{I}} |f(x)|\,\mathrm{d}x$

- The infinite-norm is given by $\|f\|_{\infty} = \max_{x\in\mathcal{I}} |f(x)|$

Function Norms

- Functions can also have norms, for example, if $f(x)$ is defined in some interval $\mathcal{I}$

$$\|f\|_{L_2} = \sqrt{\int_{x\in\mathcal{I}} f^2(x)\,\mathrm{d}x}$$

- The L_2 vector space is the set of function where $\|f\|_{L_2} < \infty$

- The L_1-norm is given by $\|f\|_{L_1} = \int_{x\in\mathcal{I}} |f(x)|\,\mathrm{d}x$

- The infinite-norm is given by $\|f\|_\infty = \max_{x\in\mathcal{I}} |f(x)|$

Summary

- Vector spaces with a distance (metric spaces) and vector spaces with a norm (normed vector spaces) are interesting objects
- They allow you to define a topology (open/closed sets, etc.)
- You can build up ideas about connectedness, continuity, contractive maps, fixed-point theorems, . . .
- For the most part we are going to consider an even more powerful vector space that has an inner-product defined

Summary

- Vector spaces with a distance (metric spaces) and vector spaces with a norm (normed vector spaces) are interesting objects
- They allow you to define a topology (open/closed sets, etc.)
- You can build up ideas about connectedness, continuity, contractive maps, fixed-point theorems, . . .
- For the most part we are going to consider an even more powerful vector space that has an inner-product defined

Summary

- Vector spaces with a distance (metric spaces) and vector spaces with a norm (normed vector spaces) are interesting objects
- They allow you to define a topology (open/closed sets, etc.)
- You can build up ideas about connectedness, continuity, contractive maps, fixed-point theorems, . . .
- For the most part we are going to consider an even more powerful vector space that has an inner-product defined

Summary

- Vector spaces with a distance (metric spaces) and vector spaces with a norm (normed vector spaces) are interesting objects
- They allow you to define a topology (open/closed sets, etc.)
- You can build up ideas about connectedness, continuity, contractive maps, fixed-point theorems, . . .
- For the most part we are going to consider an even more powerful vector space that has an inner-product defined