



Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Mathematics for Data Science

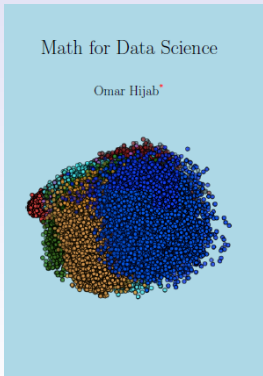
Dr. S. M. Moosavi

smohsenmoosavi2009@gmail.com

October 14, 2024



The following slides are arranged (with some modifications) based on the book "*Math for Data Science*" by "**Omar Hijab**".



You can follow me on [Linkedin](#). Also, for course materials such as slides and the related python codes, see this [Github](#) repository.



Outline

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

- 1 Data Sets
- 2 Linear Geometry



Outline

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

1 Data Sets

2 Linear Geometry



What is a dataset?

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.1

Geometrically, a dataset is a sample of N points x_1, x_2, \dots, x_N in d -dimensional space \mathbb{R}^d . Algebraically, a dataset is an $N \times d$ matrix.

Practically speaking, the following are all representations of datasets:

matrix = CSV file = spreadsheet = SQL table = array = dataframe

Definition 1.2

Each point $x = (t_1, t_2, \dots, t_d)$ in the dataset is a sample or an example, and the components t_1, t_2, \dots, t_d of a sample point x are its features or attributes. As such, d -dimensional space \mathbb{R}^d is feature space.

Definition 1.3

Sometimes one of the features is separated out as the label. In this case, the dataset is a labelled dataset.



Iris dataset

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

The *Iris dataset* contains 150 examples of four features of Iris flowers, and there are three classes of Irises, *Setosa*, *Versicolor* and *Virginica*, with 50 samples from each class.

Samples (instances, observations)						Petal	
	Sepal length	Sepal width	Petal length	Petal width	Class label		
1	5.1	3.5	1.4	0.2	Setosa		
2	4.9	3.0	1.4	0.2	Setosa		
...							
50	6.4	3.5	4.5	1.2	Versicolor		
...							
150	5.9	3.0	5.0	1.8	Virginica		
Features (attributes, measurements, dimensions)					Class labels (targets)	Sepal	



MNIST dataset

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

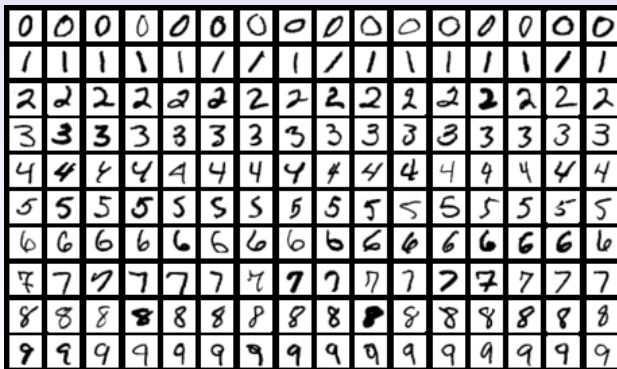
Complex Numbers

Mean and Covariance

Linear Geometry

The *MNIST dataset* consists of 60,000 images of hand-written digits. There are 10 classes of images, corresponding to each digit $0, 1, \dots, 9$. We seek to compress the images while preserving as much as possible of the images' characteristics.

Each image is a grayscale 28×28 pixel image. Since $28^2 = 784$, each image is a point in $d = 784$ dimensions. Here there are $N = 60000$ samples and $d = 784$ features.





Exercises

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

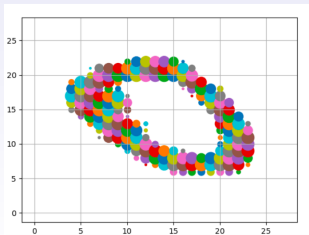
Linear Geometry

Exercise 1.1

Use sklearn to download Iris dataset.

Exercise 1.2

- *From keras read the MNIST dataset.*
- *Let $(\text{train_X}, \text{train_y}), (\text{test_X}, \text{test_y}) = \text{mnist.load_data}()$*
- *Let $\text{pixels} = \text{train_X}[1]$.*
- *Do for loops over i and j in $\text{range}(28)$ and use scatter to plot points at location (i,j) with size given by $\text{pixels}[i,j]$, then show the following image.*





Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Suppose we have a population of things (people, tables, numbers, vectors, images, etc.) and we have a sample of size N from this population:

$$\mathbf{1} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$$

The total population is the *population* or the *sample space*.

Example 1.1

The sample space consists of all real numbers and we take $N = 5$ samples from

$$\mathbf{1} = [3.95, 3.20, 3.10, 5.55, 6.93]$$

Example 1.2

The sample space consists of all integers and we take $N = 5$ samples from

$$\mathbf{1} = [35, -32, -8, 45, -8]$$



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.3

The sample space consists of all Python strings and we take $N = 5$ samples from

```
l = ['a2e?', '%T', '7y5', ' ', 'kkk>><</', '[]*+']
```

Example 1.4

The sample space consists of all HTML colors and we take $N = 5$ samples from

```
1 from random import choice
2 import matplotlib.pyplot as plt
3
4 def hexcolor():
5     return "#" + ''.join([choice('0123456789abcdef') for
6                           _ in range(6)])
7
8 for i in range(5): plt.scatter(i,0, c=hexcolor())
plt.show()
```



Mean

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Let \mathbf{l} be a list as above. The goal is to compute the sample *average* or *mean* of the list, which is

$$\text{mean} = \text{average} = \frac{x_1 + x_2 + \cdots + x_N}{N}.$$

In the Example (1.1), the average is

$$\frac{3.95 + 3.20 + 3.10 + 5.55 + 6.93}{5} = 4.546.$$

Example 1.5

```
1  import numpy as np
2
3  dataset = np.array([3.95, 3.20, 3.10, 5.55, 6.93])
4  print(np.mean(dataset))
5
6  output: 4.546
```

In the Example (1.2), the average is $\frac{32}{5}$. In the Example (1.3), while we can add strings, we can't divide them by 5, so the average is undefined. Similarly for colors: the average is undefined.



Vector space

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

A sample space or population V is called a *vector space* if, roughly speaking, one can compute means or averages in V . In this case, we call the members of the population "vectors".

Definition 1.4 (Vector space)

Let V be a set. V is a vector space (over \mathbb{R}) if for every $u, v, w \in V$ and $r, s \in \mathbb{R}$:

- 1 *vectors can be added (and the sum $v + w$ is back in V);*
- 2 *vector addition is commutative $v + w = w + v$*
- 3 *vector addition is associative $u + (v + w) = (u + v) + w$;*
- 4 *there is a zero vector $\mathbf{0}$ ($\mathbf{0} + v = v$);*
- 5 *vectors v have negatives (or opposites) $-v$ ($v + (-v) = \mathbf{0}$);*
- 6 *vectors can be multiplied by real numbers (and the product rv is back in V);*
- 7 *multiplication is distributive over addition $(r + s)v = rv + sv$ and $r(u + v) = ru + rv$;*
- 8 *$1v = v$ and $0v = \mathbf{0}$;*
- 9 *$r(sv) = (rs)v$.*



Centered dataset

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

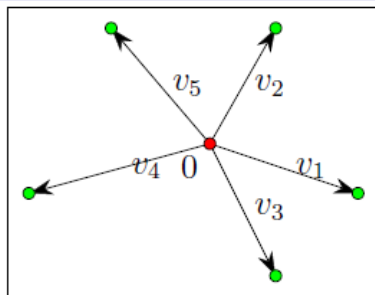
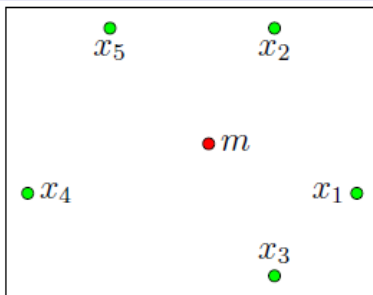
Linear Geometry

Definition 1.5 (Centered Versus Non-Centered)

If x_1, x_2, \dots, x_N is a dataset of points with mean m and

$$v_1 = x_1 - m, v_2 = x_2 - m, \dots, v_N = x_N - m,$$

then v_1, v_2, \dots, v_N is a centered dataset of vectors where its mean is zero.





Some notes

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

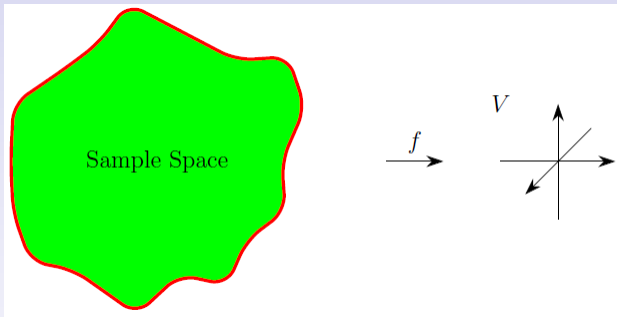
Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

- When we work with vector spaces, numbers are referred to as *scalars*.
- When we multiply a vector v by a scalar r to get the scaled vector rv , we call it *scalar multiplication*.
- The set of all real numbers \mathbb{R} is a vector space.
- The set of all integers \mathbb{Z} is not a vector space.
- The set of all rational numbers \mathbb{Q} is a vector space over \mathbb{Q} but not over \mathbb{R} .
- The set of all Python strings is not a vector space.
- Usually, we can't take sample means from a population, we instead take the sample mean of a *statistic* associated to the population. A statistic is an assignment of a number $f(\text{item})$ to each item in the population. For example, the human population on Earth is not a vector space (they can't be added), but their heights is a vector space (heights can be added). For the Python strings, a statistic might be the length of the strings. For the HTML colors, a statistic is the HTML code of the color.



In general, a statistic need not be a number. A statistic can be anything that "behaves like a number". For example, $f(\text{item})$ can be a vector or a matrix. More generally, a statistic's values may be anything that lives in a vector space V .



Cartesian plane

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

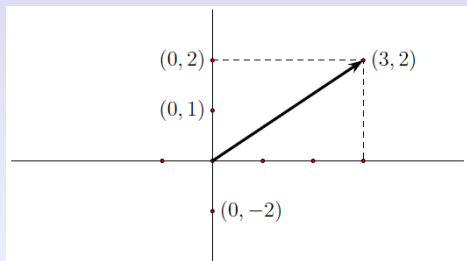
Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

The *cartesian plane* \mathbb{R}^2 , also called the 2-dimensional real space is a vector space.



For $\mathbf{v}_1 = (x_1, y_1), \mathbf{v}_2 = (x_2, y_2) \in \mathbb{R}^2$ and $t \in \mathbb{R}$ define

- $\mathbf{v}_1 + \mathbf{v}_2 = (x_1 + x_2, y_1 + y_2)$ (Addition).
- $\mathbf{0} = (0, 0)$ (Zero).
- $t\mathbf{v}_1 = (tx_1, ty_1)$ (Scaling).
- $-\mathbf{v}_1 = (-1)\mathbf{v}_1$ (Negative).
- $\mathbf{v}_1 - \mathbf{v}_2 = \mathbf{v}_1 + (-\mathbf{v}_2) = (x_1 - x_2, y_1 - y_2)$ (Subtraction).



Operations

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

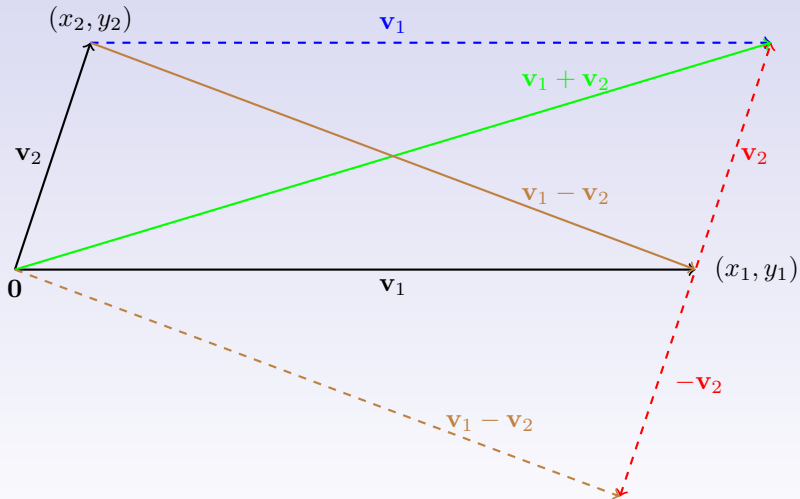
Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry





2d example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.6

```
1  import numpy as np
2
3  v1 = (1,2)
4  v2 = (3,4)
5  print(v1 + v2 == (1+3,2+4)) # returns False
6
7  v1 = [1,2]
8  v2 = [3,4]
9  print(v1 + v2 == [1+3,2+4]) # returns False
10
11 v1 = np.array([1,2])
12 v2 = np.array([3,4])
13 print(v1 + v2 == np.array([1+3,2+4]))
14 # returns [ True  True]
15 print(3*v1 == np.array([3,6]))
16 # returns [ True  True]
17 print(-v1 == np.array([-1,-2]))
18 # returns [ True  True]
19 print(v1 - v2 == np.array([1-3,2-4]))
20 # returns [ True  True]
```



2d example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

For the two-dimensional dataset

$$\mathbf{x}_1 = (1, 2), \mathbf{x}_2 = (3, 4), \mathbf{x}_3 = (-2, 11), \mathbf{x}_4 = (0, 66),$$

or, equivalently,

$$\mathbf{x} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ -2 & 11 \\ 0 & 66 \end{pmatrix},$$

the average is

$$\frac{(1, 2) + (3, 4) + (-2, 11) + (0, 66)}{4} = (0.5, 20.75).$$

Example 1.7

```
1 import numpy as np
2
3 dataset = np.array([[1,2], [3,4], [-2,11], [0,66]])
4 print(np.mean(dataset, axis=0))
5 # returns [ 0.5 , 20.75]
```



2d example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

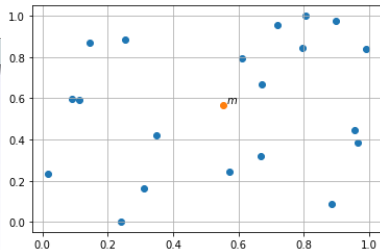
Mean and Covariance

Linear Geometry

Example 1.8

Generate a 2 dimensional dataset of random points and their mean

```
1 import numpy as np
2 from numpy.random import random as rd
3 import matplotlib.pyplot as plt
4 N = 20
5 dataset = np.array([[rd(), rd()] for _ in range(N)])
6 mean = np.mean(dataset,axis=0)
7 plt.grid()
8 X, Y = dataset[:,0], dataset[:,1]
9 plt.scatter(X,Y)
10 plt.scatter(*mean)
11 plt.annotate('$m$', xy=mean+0.01)
12 plt.show()
```





Magnitude

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.6 (Distance Formula)

If $\mathbf{v}_1 = (x_1, y_1)$ and $\mathbf{v}_2 = (x_2, y_2)$, then the distance between \mathbf{v}_1 and \mathbf{v}_2 is

$$|\mathbf{v}_1 - \mathbf{v}_2| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$

The distance of $\mathbf{v} = (x, y)$ to the origin $\mathbf{0} = (0, 0)$ is its magnitude or norm or length

$$r = |\mathbf{v}| = |\mathbf{v} - \mathbf{0}| = \sqrt{x^2 + y^2}.$$

Example 1.9

For $\mathbf{v}_1 = (1, 2)$ and $\mathbf{v}_2 = (3, 4)$

$$|\mathbf{v}_1| = \sqrt{1^2 + 2^2} = \sqrt{5} \simeq 2.236,$$

$$|\mathbf{v}_1 - \mathbf{v}_2| = \sqrt{(1 - 3)^2 + (2 - 4)^2} = \sqrt{4 + 4} = \sqrt{8} \simeq 2.828.$$

```

1  import numpy as np
2
3  v1 = np.array([1,2])
4  v2 = np.array([3,4])
5  print(np.linalg.norm(v1)) #returns 2.23606797749979
6  print(np.linalg.norm(v1-v2)) #returns 2.

```



Polar representation

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

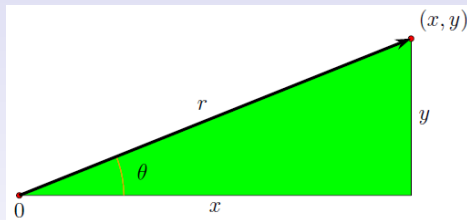
Complex Numbers

Mean and Covariance

Linear Geometry

In terms of r and θ , the *polar representation* of (x, y) is

$$x = r \cos \theta, \quad y = r \sin \theta.$$



The *unit circle* consists of the vectors which are distance 1 from the origin $\mathbf{0}$. When \mathbf{v} is on the unit circle, the magnitude of \mathbf{v} is 1, and we say \mathbf{v} is a *unit vector*. In this case, the line formed by the scalings of \mathbf{v} intersects the unit circle at $\pm \mathbf{v}$.

When \mathbf{v} is a unit vector, then $r = 1$ and $\mathbf{v} = (x, y) = (\cos \theta, \sin \theta)$.



Polar representation

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

By the distance formula, a vector $\mathbf{v} = (x, y)$ is a unit vector when

$$x^2 + y^2 = 1.$$

More generally, any circle with *center* (a, b) and radius r consists of vectors $\mathbf{v} = (x, y)$ satisfying

$$(x - a)^2 + (y - b)^2 = r^2.$$

Let R be a point on the unit circle, and let $t > 0$. The scaled point tR is on the circle with center $(0, 0)$ and radius t . Moreover, if Q is any point, $Q + tR$ is on the circle with center Q and radius t . It is easy to check that $|t\mathbf{v}| = |t||\mathbf{v}|$ for any real number t and vector \mathbf{v} .

From this, if a vector \mathbf{v} is unit and $r > 0$, then $r\mathbf{v}$ has magnitude r . If \mathbf{v} is any vector not equal to the zero vector, then $r = |\mathbf{v}|$ is positive, and

$$\left| \frac{1}{r} \mathbf{v} \right| = \frac{1}{r} |\mathbf{v}| = \frac{1}{r} r = 1$$

so \mathbf{v}/r is a unit vector.



Inner product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.7

Let $\mathbf{v}_1 = (x_1, y_1), \mathbf{v}_2 = (x_2, y_2) \in \mathbb{R}^2$. The inner product or the dot product of \mathbf{v}_1 and \mathbf{v}_2 is given algebraically as

$$\mathbf{v}_1 \cdot \mathbf{v}_2 = x_1x_2 + y_1y_2.$$

From the geometric view, we have:

Theorem 1.1 (Dot Product Identity)

$$x_1x_2 + y_1y_2 = \mathbf{v}_1 \cdot \mathbf{v}_2 = |\mathbf{v}_1||\mathbf{v}_2| \cos \theta,$$

where θ is the angle between \mathbf{v}_1 and \mathbf{v}_2 .

Exercise 1.3

Prove the "Dot Product Identity", Theorem (1.1).

Hint: Use Pythagoras' theorem for general triangles.



The angle between two vectors

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

In Python, the dot product is given by `numpy.dot` and as a consequence of the dot product identity, we have the code for the angle between two vectors:

$$\theta_{\mathbf{v}_1, \mathbf{v}_2} = \arccos \left(\frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{|\mathbf{v}_1| |\mathbf{v}_2|} \right).$$

Example 1.10

Find the angle between the vectors $\mathbf{v}_1 = (1, 2)$ and $\mathbf{v}_2 = (3, 4)$.

```
1  import numpy as np
2
3  def angle(u,v):
4      a = np.dot(u,v)
5      b = np.dot(u,u)
6      c = np.dot(v,v)
7      theta = np.arccos(a / np.sqrt(b*c))
8      return np.degrees(theta)
9
10 v1 = np.array([1,2])
11 v2 = np.array([3,4])
12 print(angle(v1,v2)) #returns 10.304846468766044 in
                        degree
```



Cauchy-Schwarz Inequality

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Recall that $-1 \leq \cos \theta \leq 1$. Using the dot product identity, we obtain the important inequality:

Theorem 1.2 (Cauchy-Schwarz Inequality)

If u and v are any two vectors, then

$$-|u||v| \leq u \cdot v \leq |u||v|.$$

Exercise 1.4

Prove the "Cauchy-Schwarz Inequality".



2d linear equations system

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Consider the homogeneous system

$$\begin{cases} ax + by = 0 \\ cx + dy = 0 \end{cases} \quad (1.1)$$

and let A be the 2×2 matrix

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}. \quad (1.2)$$

$(x, y) = (-b, a)$ is a solution of the first equation in (1.1). If we want this to be a solution of the second equation as well, we must have $cx + dy = ad - bc = 0$.

Definition 1.8 (Determinant)

The determinant of A is

$$\det(A) = \det \begin{pmatrix} a & b \\ c & d \end{pmatrix} = ad - bc.$$



2d linear equations system

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Theorem 1.3 (Homogeneous System)

*When $\det(A) = 0$, the homogeneous system (1.1) has a nonzero solution, and all solutions are scalar multiples of $(x, y) = (-b, a)$.
When $\det(A) \neq 0$, the only solution is $(x, y) = (0, 0)$.*

For the inhomogeneous case

$$\begin{cases} ax + by = e \\ cx + dy = f \end{cases} \quad (1.3)$$

we have

Theorem 1.4 (Inhomogeneous System)

When $\det(A) \neq 0$, the inhomogeneous system (1.3) has the unique solution

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{\det(A)} \begin{pmatrix} de - bf \\ af - ce \end{pmatrix}.$$

When $\det(A) = 0$, (1.3) has a solution iff $ce = af$ and $de = bf$.



2d linear equations system

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

When $a^2 + b^2 \neq 0$, a solution is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{a^2 + b^2} \begin{pmatrix} ae \\ be \end{pmatrix}.$$

When $c^2 + d^2 \neq 0$, a solution is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{c^2 + d^2} \begin{pmatrix} cf \\ df \end{pmatrix}.$$

Any other solution differs from these solutions by a scalar multiple of the homogeneous solution $(x, y) = (-b, a)$.

Exercise 1.5

Prove the Theorems (1.3) and (1.4).



Complex numbers

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Roughly speaking, the set of all *complex numbers* is the set of all points in \mathbb{R}^2 with different multiplication rule.

Definition 1.9 (Complex numbers)

The complex numbers, \mathbb{C} , is the set

$$\mathbb{C} = \{(x, y) \in \mathbb{R}^2\}$$

with operations

- *Addition:* $(x_1, y_1) + (x_2, y_2) = (x_1 + x_2, y_1 + y_2)$.
- *Scalar Multiplication:* $t(x, y) = (tx, ty)$
- *Multiplication:* $(x_1, y_1)(x_2, y_2) = (x_1x_2 - y_1y_2, x_1y_2 + x_2y_1)$.

Then, in \mathbb{C} , we have

- zero: $0 = (0, 0)$.
- opposite or additive inverse: $-(x, y) = (-x, -y)$.
- one: $1 = (1, 0)$.



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.11

- $(1, 2) + (3, 4) = (4, 6).$
- $(0, 0) + (1, 2) = (1, 2).$
- $3(1, 2) = (3, 6).$
- $(1, 0)(1, 2) = (1 - 0, 2 + 0) = (1, 2).$
- $(1, 2)(3, 4) = (3 - 8, 4 + 6) = (-5, 10).$
- $(x, 0) + (y, 0) = (x + y, 0).$
- $(x, 0)(y, 0) = (xy, 0).$

Note. By the last two examples, we see that complex numbers with 0 as their second component act like real numbers in addition and multiplication. So, from now on, we set $x = (x, 0).$

Example 1.12

- $0 = (0, 0).$
- $1 = (1, 0).$
- $-1 = (-1, 0).$



Imaginary number

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.10 (Imaginary number)

$$i = (0, 1).$$

Note. Python uses the symbol j for imaginary number.

Theorem 1.5

For each $z = (x, y) \in \mathbb{C}$, we can write

$$z = x + iy.$$

We call x as the real part of z , and y the imaginary part of z .

$$x = \operatorname{Re}(z), \quad y = \operatorname{Im}(z).$$

Proof. $x + iy = (x, 0) + (0, 1)(y, 0) = (x, 0) + (0 - 0, 0 + y) = (x, y).$

Theorem 1.6

$$i^2 = -1.$$

Proof. $i^2 = (0, 1)(0, 1) = (0 - 1, 0 + 0) = (-1, 0) = -1.$



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.13

In complex numbers:

- $\sqrt{-1} = i.$
- $\sqrt{-4} = 2i.$
- $(1, 2)(3, 4) = (1 + 2i)(3 + 4i)$
$$= 3 + 4i + 6i + 8i^2$$
$$= 3 + 10i - 8$$
$$= -5 + 10i$$
$$= (-5, 10).$$
- $(1, 2)^3 = (1 + 2i)^3$
$$= (1)^3 + 3(1)^2(2i) + 3(1)(2i)^2 + (2i)^3$$
$$= 1 + 6i + 12i^2 + 8i^3$$
$$= 1 + 6i - 12 - 8i$$
$$= -11 - 2i$$
$$= -(11, 2).$$



Conjugate

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.11 (Conjugate)

For $z = (x, y) \in \mathbb{C}$, the conjugate is

$$\bar{z} = (x, -y) = x - iy \in \mathbb{C}.$$

Some properties.

- $z + \bar{z} = 2\text{Re}(z)$, $z - \bar{z} = 2i\text{Im}(z)$.
- $z\bar{z} = \text{Re}(z)^2 + \text{Im}(z)^2$,

$$\Rightarrow |z| = \sqrt{\text{Re}(z)^2 + \text{Im}(z)^2} = \sqrt{z\bar{z}}$$

$$\Rightarrow |z|^2 = z\bar{z}.$$

Example 1.14

For $z = (4, -3) \in \mathbb{C}$:

- $\bar{z} = (4, 3) = 4 + 3i$,
- $z + \bar{z} = 2 \times 4 = 8$, $z - \bar{z} = 2i \times (-3) = -6i$.
- $z\bar{z} = (4)^2 + (-3)^2 = 16 + 9 = 25 \Rightarrow |z| = \sqrt{25} = 5$.
- $z^2 = (4 - 3i)^2 = 7 - 24i$.
- $|z|^2 = 25$.



Theorem 1.7

For a non-zero $z \in \mathbb{C}$, the inverse of z is

$$z^{-1} = \frac{1}{z} = \frac{\bar{z}}{z\bar{z}} = \frac{\bar{z}}{|z|^2}.$$

Proof. Firstly, if $z = (x, y)$ then $\frac{1}{z} \in \mathbb{C}$, because,

$$\frac{1}{z} = \frac{x - iy}{x^2 + y^2} = \left(\frac{x}{x^2 + y^2}, \frac{-y}{x^2 + y^2} \right) \in \mathbb{C}.$$

Secondly,

$$zz^{-1} = (x + iy) \left(\frac{x - iy}{x^2 + y^2} \right) = \frac{x^2 + y^2}{x^2 + y^2} = 1.$$

Corollary 1.1 (Division)

For $z_1 \in \mathbb{C}$ and $0 \neq z_2 \in \mathbb{C}$

$$\frac{z_1}{z_2} = z_1 z_2^{-1}.$$



Definitions

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.12 (Mean-squared distance)

Let $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ be a dataset, say D , in \mathbb{R}^d , and let $\mathbf{x} \in \mathbb{R}^d$. The mean-squared distance of \mathbf{x} to D is

$$MSD(\mathbf{x}) = \frac{1}{N} \sum_{k=1}^N |\mathbf{x}_k - \mathbf{x}|^2.$$

Definition 1.13 (Mean)

Let $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ be a dataset in \mathbb{R}^d . The mean or sample mean is

$$\mathbf{m} = \bar{\mathbf{x}}_N = \frac{1}{N} \sum_{k=1}^N \mathbf{x}_k = \frac{\mathbf{x}_1 + \mathbf{x}_2 + \dots + \mathbf{x}_N}{N}.$$

Theorem 1.8 (Point of Best-fit)

The mean is the point of best-fit: The mean minimizes the mean-squared distance to the dataset.

Exercise 1.6

Prove the Theorem (1.8).



Point of Best-fit

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.15

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 np.random.seed(1)
5 N = 20
6 rnd = np.random.random
7 dataset = np.array([ [rnd(), rnd()] for _ in range(N) ])
8 # Mean
9 m = np.mean(dataset, axis=0)
10 #Random point
11 p = np.array([rnd(), rnd()])
12
13 plt.grid()
14 X, Y = dataset[:,0], dataset[:,1]
15 plt.scatter(X,Y)
16 for v in dataset:
17     plt.plot([m[0], v[0]], [m[1], v[1]], c='green')
18     plt.plot([p[0], v[0]], [p[1], v[1]], c='red')
19 plt.show()
20
21 # Comparison of MSD of the mean and a random point
22 MSD_m = np.sum(np.abs(dataset-m)**2)/N
23 MSD_p = np.sum(np.abs(dataset-p)**2)/N
24 print(MSD_m, MSD_p) # 0.160478187272121 0.5984208474157081
```



Point of Best-fit

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

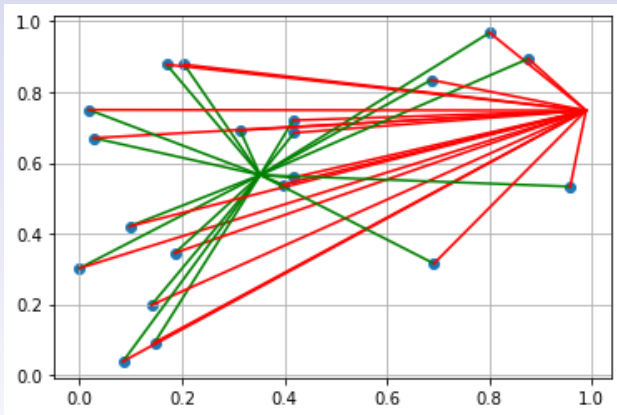


Figure 1.1: MSD for the mean (green) versus MSD for a random point (red).



Tensor product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

For simplicity, let $\mathbf{u} = (a, b)$ and $\mathbf{v} = (c, d, e)$ be two vectors.

Definition 1.14 (Tensor product)

The tensor product of \mathbf{u} and \mathbf{v} is the matrix

$$\mathbf{u} \otimes \mathbf{v} = \begin{pmatrix} ac & ad & ae \\ bc & bd & be \end{pmatrix} = \begin{pmatrix} c\mathbf{u} & d\mathbf{u} & e\mathbf{u} \end{pmatrix} = \begin{pmatrix} a\mathbf{v} \\ b\mathbf{v} \end{pmatrix}$$

Definition 1.15 (Trace of a matrix)

The trace of a squared matrix A is the sum of the diagonal entries.

Note. For any vectors \mathbf{u} , \mathbf{v} and \mathbf{w} :

- $\mathbf{v} \otimes \mathbf{u} = (\mathbf{u} \otimes \mathbf{v})^t.$

In square case:

- $\det(\mathbf{u} \otimes \mathbf{v}) = 0.$

- $\text{trace}(\mathbf{u} \otimes \mathbf{v}) = \mathbf{u} \cdot \mathbf{v}.$

- $\text{trace}(\mathbf{u} \otimes \mathbf{u}) = |\mathbf{u}|^2.$

- $(\mathbf{u} \otimes \mathbf{v})\mathbf{w} = (\mathbf{v} \cdot \mathbf{w})\mathbf{u}.$



Covariance

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Let $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ be a dataset in \mathbb{R}^d with \mathbf{m} as its mean.

Definition 1.16 (1d Covariance)

When $d = 1$, the covariance q is a scalar

$$q = \frac{1}{N} \sum_{k=1}^N (x_k - m)^2 = MSD(m).$$

In the scalar case, the covariance is called the variance of the scalar dataset.

In general, the covariance is a symmetric $d \times d$ matrix Q . We can center the dataset as

$$\mathbf{v}_1 = \mathbf{x}_1 - \mathbf{m}, \mathbf{v}_2 = \mathbf{x}_2 - \mathbf{m}, \dots, \mathbf{v}_N = \mathbf{x}_N - \mathbf{m}.$$

Then the *covariance matrix* is the $d \times d$ matrix Q as

$$Q = \frac{\mathbf{v}_1 \otimes \mathbf{v}_1 + \mathbf{v}_2 \otimes \mathbf{v}_2 + \dots + \mathbf{v}_N \otimes \mathbf{v}_N}{N}. \quad (1.4)$$



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.16

Suppose $N = 5$ and

$$\mathbf{x}_1 = (1, 2), \quad \mathbf{x}_2 = (3, 4), \quad \mathbf{x}_3 = (5, 6), \quad \mathbf{x}_4 = (7, 8), \quad \mathbf{x}_5 = (9, 10).$$

Then $\mathbf{m} = (5, 6)$ and

$$\mathbf{v}_1 = \mathbf{x}_1 - \mathbf{m} = (-4, -4), \quad \mathbf{v}_2 = \mathbf{x}_2 - \mathbf{m} = (-2, -2),$$

$$\mathbf{v}_3 = \mathbf{x}_3 - \mathbf{m} = (0, 0), \quad \mathbf{v}_4 = \mathbf{x}_4 - \mathbf{m} = (2, 2), \quad \mathbf{v}_5 = \mathbf{x}_5 - \mathbf{m} = (4, 4).$$

Since

$$(\pm 4, \pm 4) \otimes (\pm 4, \pm 4) = \begin{pmatrix} 16 & 16 \\ 16 & 16 \end{pmatrix},$$

$$(\pm 2, \pm 2) \otimes (\pm 2, \pm 2) = \begin{pmatrix} 4 & 4 \\ 4 & 4 \end{pmatrix},$$

$$(0, 0) \otimes (0, 0) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix},$$

then

$$Q = \begin{pmatrix} 8 & 8 \\ 8 & 8 \end{pmatrix}.$$



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Example 1.17

```
1  import numpy as np
2
3  def tensor(u,v):
4      return np.array([ [ a*b for b in v] for a in u ])
5
6  np.random.seed(1)
7  N = 20
8  rnd = np.random.random
9  dataset = np.array([[rnd(), rnd()] for _ in range(N)])
10 # mean
11 m = np.mean(dataset,axis=0)
12 # center dataset
13 vectors = dataset - m
14 # covariance
15 Q = np.mean([ tensor(v,v) for v in vectors ],axis=0)
16 print(Q)
```



Note. The covariance matrix as written in (1.4) is the *biased covariance matrix*. If the denominator is instead $N - 1$, the matrix is the *unbiased covariance matrix*.

For datasets with large N , it doesn't matter, since N and $N - 1$ are almost equal.

In numpy, the Python covariance constructor is

Example 1.18

```
1  import numpy as np
2
3  np.random.seed(1)
4  N = 20
5  rnd = np.random.random
6  dataset = np.array([[rnd(), rnd()] for _ in range(N)])
7  # covariance
8  Q = np.cov(dataset, bias=True, rowvar=False)
9  print(Q)
```



Total variance

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Definition 1.17 (Total variance)

From $\text{trace}(\mathbf{u} \otimes \mathbf{u}) = |\mathbf{u}|^2$, if Q is the covariance matrix then

$$\text{trace}(Q) = \frac{1}{N} \sum_{k=1}^N |\mathbf{x}_k - \mathbf{m}|^2. \quad (1.5)$$

We call (1.5) the total variance of the dataset. Thus the total variance equals $\text{MSD}(\mathbf{m})$.

Example 1.19

```
1  import numpy as np
2
3  np.random.seed(1)
4  N = 20
5  rnd = np.random.random
6  dataset = np.array([[rnd(), rnd()] for _ in range(N)])
7  # covariance
8  Q = np.cov(dataset.T, bias=True)
9  print(Q.trace()) # returns 0.16047818727212101
```



Projections

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

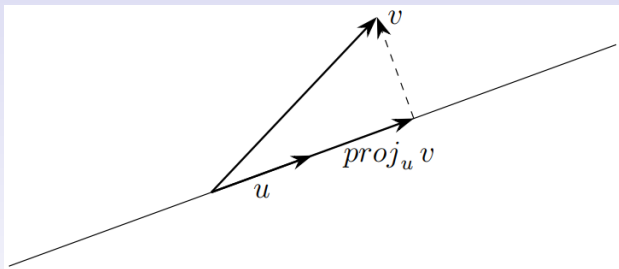
Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

We would like to project a $2d$ dataset onto a line. Let \mathbf{u} be a unit vector (a vector of length one, $|\mathbf{u}| = 1$), and let $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ be a $2d$ dataset, assumed for simplicity to be centered. We wish to project this dataset onto the line through \mathbf{u} . This will result in a $1d$ dataset.



When a vector \mathbf{v} is projected onto the line through \mathbf{u} , the length of the projected vector reads

$$|proj_{\mathbf{u}} \mathbf{v}| = |\mathbf{v}| \cos \theta,$$

where θ is the angle between the vectors \mathbf{v} and \mathbf{u} . Since $|\mathbf{u}| = 1$, this length equals the dot product $\mathbf{v} \cdot \mathbf{u}$. Hence the projected vector is

$$proj_{\mathbf{u}} \mathbf{v} = (\mathbf{v} \cdot \mathbf{u}) \mathbf{u}.$$



Projections

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Hence,

Definition 1.18 (Reduced dataset)

The projected dataset of $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ onto the line through \mathbf{u} is the dataset

$$(\mathbf{v}_1 \cdot \mathbf{u})\mathbf{u}, (\mathbf{v}_2 \cdot \mathbf{u})\mathbf{u}, \dots, (\mathbf{v}_N \cdot \mathbf{u})\mathbf{u}.$$

The projected dataset is in \mathbb{R}^2 . The reduced dataset is

$$(\mathbf{v}_1 \cdot \mathbf{u}), (\mathbf{v}_2 \cdot \mathbf{u}), \dots, (\mathbf{v}_N \cdot \mathbf{u}),$$

which is in \mathbb{R} .

Exercise 1.7

Show that when a $2d$ dataset is centered then the mean of the reduced dataset is 0.

Exercise 1.8

Prove that if Q is the covariance matrix of a $2d$ dataset, then the variance of the projected dataset onto the line through the vector \mathbf{u} equals the quadratic function $\mathbf{u} \cdot Q\mathbf{u}$:

$$q = \frac{1}{N} \sum_{k=1}^N \mathbf{u} \cdot (\mathbf{v}_k \otimes \mathbf{v}_k) \mathbf{u} = \mathbf{u} \cdot Q\mathbf{u}.$$



Covariance ellipse

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Hence,

Definition 1.19 (Covariance ellipse)

The contour of all points \mathbf{x} satisfying $\mathbf{x} \cdot Q\mathbf{x} = 1$ is the covariance ellipsoid. In two dimensions $d = 2$, this is the covariance ellipse. The contour of all points \mathbf{x} satisfying $\mathbf{x} \cdot Q^{-1}\mathbf{x} = 1$ is the inverse covariance ellipsoid. In two dimensions $d = 2$, this is the inverse covariance ellipse.

In two dimensions $d = 2$, a covariance matrix has the form

$$Q = \begin{pmatrix} a & b \\ b & c \end{pmatrix}.$$

If we write $\mathbf{u} = (x, y)$ for a vector in the plane, the covariance ellipse is

$$\mathbf{u} \cdot Q\mathbf{u} = (x, y) \cdot \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = ax^2 + 2bxy + cy^2 = 1.$$

The covariance ellipse and inverse covariance ellipses described above are centered at the origin $(0, 0)$. When a dataset has mean \mathbf{m} and covariance Q , the ellipses are drawn centered at \mathbf{m} .

In particular, when $a = c$ and $b = 0$, then $Q = aI$ is a multiple of the identity, the inverse covariance ellipse is the circle of radius \sqrt{a} , and the covariance ellipse is the circle of radius $\frac{1}{\sqrt{a}}$.



Covariance ellipse I

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

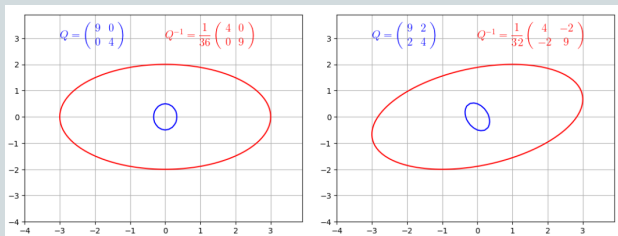
Mean and Covariance

Linear Geometry

Example 1.20

Plot the contour ellipses for

$$Q_1 = \begin{pmatrix} 9 & 0 \\ 0 & 4 \end{pmatrix}, \quad Q_2 = \begin{pmatrix} 9 & 2 \\ 2 & 4 \end{pmatrix}.$$





Covariance ellipse II

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

```
1  import matplotlib.pyplot as plt
2  import numpy as np
3
4  def ellipse(a, b, c, levels, color):
5      L, delta = 4, .1
6      x = np.arange(-L,L,delta)
7      y = np.arange(-L,L,delta)
8      X,Y = np.meshgrid(x, y)
9      plt.contour(X, Y, a*X**2 + 2*b*X*Y + c*Y**2, levels,
                  colors=color)
10
11  # Q1 Covariance entities
12  a, b, c = 9, 0, 4
13
14  # Inverse Covariance entities
15  det = a*c - b**2
16  A, B, C = c/det, -b/det, a/det
17
18  plt.grid()
19  ellipse(a, b, c, [20], 'blue')
20  ellipse(A, B, C, [1], 'red')
21  plt.show()
```



Covariance ellipse III

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

```
22
23 # Q2 Covariance entities
24 a, b, c = 9, 2, 4
25
26 # Inverse Covariance entities
27 det = a*c - b**2
28 A, B, C = c/det, -b/det, a/det
29
30 plt.grid()
31 ellipse(a, b, c, [1], 'blue')
32 ellipse(A, B, C, [1], 'red')
33 plt.show()
```



Standardization

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

Here, we describe how to standardize datasets in \mathbb{R}^2 . *Standardizing* the dataset means to center the dataset and to place the x and y features on the same scale.

Consider the dataset

$\mathbf{x}_1 = (x_1, y_1), \mathbf{x}_2 = (x_2, y_2), \dots, \mathbf{x}_N = (x_N, y_N)$ with mean $\mathbf{m} = (m_x, m_y)$. Then the covariance matrix is

$$Q = \begin{pmatrix} a & b \\ b & c \end{pmatrix},$$

where

$$a = \frac{1}{N} \sum_{k=1}^N (x_k - m_x)^2, \quad b = \frac{1}{N} \sum_{k=1}^N (x_k - m_x)(y_k - m_y),$$

$$c = \frac{1}{N} \sum_{k=1}^N (y_k - m_y)^2.$$



Standardization

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

If a and c differ, the different scales of x 's and y 's distorts the relation between them, and b may not accurately reflect the correlation. To correct for this, we center and re-scale

$$x_1, x_2, \dots, x_N \rightarrow x'_1 = \frac{x_1 - m_x}{\sqrt{a}}, x'_2 = \frac{x_2 - m_x}{\sqrt{a}}, \dots, x'_N = \frac{x_N - m_x}{\sqrt{a}}$$

and

$$y_1, y_2, \dots, y_N \rightarrow y'_1 = \frac{y_1 - m_y}{\sqrt{c}}, y'_2 = \frac{y_2 - m_y}{\sqrt{c}}, \dots, y'_N = \frac{y_N - m_y}{\sqrt{c}}$$

This results in a new dataset

$\mathbf{v}_1 = (x'_1, y'_1), \mathbf{v}_2 = (x'_2, y'_2), \dots, \mathbf{v}_N = (x'_N, y'_N)$ that is centered:

$$\frac{\mathbf{v}_1 + \mathbf{v}_2 + \dots + \mathbf{v}_N}{N} = 0,$$

with each feature standardized to have unit variance,

$$\frac{1}{N} \sum_{k=1}^N x'_k = 1, \quad \frac{1}{N} \sum_{k=1}^N y'_k = 1.$$

This is the *standardized dataset*.



Standardization

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

The covariance matrix of the standardized dataset has the form

$$Q' = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix},$$

where

$$\rho = \frac{1}{N} \sum_{k=1}^N x'_k y'_k = \frac{b}{\sqrt{ac}} = \frac{\sum_{k=1}^N (x_k - m_x)(y_k - m_y)}{\sqrt{\left(\sum_{k=1}^N (x_k - m_x)^2\right) \left(\sum_{k=1}^N (y_k - m_y)^2\right)}}$$

is the *Pearson correlation coefficient* of the dataset. The matrix Q' is the *correlation matrix*, or the *standardized covariance matrix*.

Example 1.21

$$Q = \begin{pmatrix} 9 & 2 \\ 2 & 4 \end{pmatrix} \Rightarrow \rho = \frac{b}{\sqrt{ac}} = \frac{1}{3} \Rightarrow Q' = \begin{pmatrix} 1 & 1/3 \\ 1/3 & 1 \end{pmatrix}.$$



Standardization

Math for Data

Dr. S. M.
Moosavi

Data Sets

Introduction

Averages and Vector
Spaces

Two Dimensions

Complex Numbers

Mean and Covariance

Linear Geometry

From the Cauchy-Schwarz inequality, the correlation coefficient ρ is always between -1 and 1 . When $\rho = \pm 1$, the dataset samples are perfectly correlated and lie on a line passing through the mean.

When $\rho = 1$, the line has slope 1 , and when $\rho = -1$, the line has slope -1 . When $\rho = 0$, the dataset samples are completely uncorrelated and are considered two independent one-dimensional datasets (In standardized case).

In Python numpy, the correlation matrix is returned by

```
1 import numpy as np
2 np.corrcoef(dataset.T)
```

Here again, we input the transpose of the dataset if our default is vectors as rows.

Notice the $1/N$ cancels in the definition of ρ . Because of this, `corrcoef` is the same whether we deal with biased or unbiased covariance matrices.



Outline

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

1 Data Sets

2 Linear Geometry



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.1

A matrix is a listing arranged in a rectangle of rows and columns. Specifically, an $N \times d$ matrix A has N rows and d columns,

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1d} \\ a_{21} & a_{22} & \dots & a_{2d} \\ \vdots & \vdots & \dots & \vdots \\ a_{N1} & a_{N2} & \dots & a_{Nd} \end{pmatrix}$$

The transpose of A is

$$A^t = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{N1} \\ a_{12} & a_{22} & \dots & a_{N2} \\ \vdots & \vdots & \dots & \vdots \\ a_{1d} & a_{2d} & \dots & a_{Nd} \end{pmatrix}$$



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.1

$$A = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix} \Rightarrow A^t = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{pmatrix}.$$

```
1 import numpy as np
2
3 A = np.array([[1,6,11],[2,7,12],[3,8,13],[4,9,14],[5,10,15]])
4 print(A)
5 print(A.shape)
6 print(len(A))
7 print(A[1])
8 print(A[1,2])
9 print(A[1:3])
10
11 # transpose
12 A_t = np.transpose(A)
13 print(A_t)
14 print(A_t.shape)
15 print(len(A_t))
16 print(A_t[1])
17 print(A_t[1,2])
18 print(A_t[1:3])
```



Definition 2.2

A d -dimensional vector \mathbf{v} may be written as a $1 \times d$ matrix

$$\mathbf{v} = (t_1 \quad t_2 \quad \cdots \quad t_d).$$

In this case, we call \mathbf{v} a row vector.

Definition 2.3

An N -dimensional vector \mathbf{v} may be written as an $N \times 1$ matrix

$$\mathbf{v} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}.$$

In this case, we call \mathbf{v} a column vector.



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ with the same dimension may be stacked as columns (`np.column_stack` in Python) of a matrix,

$$A = \begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_d \end{pmatrix}.$$

Similarly, vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ with the same dimension may be stacked as rows (`np.row_stack` in Python) of a matrix,

$$A = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_N \end{pmatrix}.$$



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.2

The row stack of $\mathbf{v}_1 = (1, 6, 11)$, $\mathbf{v}_2 = (2, 7, 12)$, $\mathbf{v}_3 = (3, 8, 13)$, $\mathbf{v}_4 = (4, 9, 14)$ and $\mathbf{v}_5 = (5, 10, 15)$ reads:

$$A = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix},$$

and the column stack of them is:

$$A^t = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{pmatrix}.$$

```
1 import numpy as np
2
3 v1 = [1, 6, 11]
4 v2 = [2, 7, 12]
5 v3 = [3, 8, 13]
6 v4 = [4, 9, 14]
7 v5 = [5, 10, 15]
8 A = np.row_stack((v1, v2, v3, v4, v5))
9 print(A)
10 A_t = np.column_stack((v1, v2, v3, v4, v5))
11 print(A_t)
```



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.4

A matrix is square if the number of rows equals the number of columns.

Definition 2.5

A matrix is diagonal if the off-diagonal entities are zero.

Example 2.3

The matrix

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{pmatrix},$$

is square and diagonal.

The following matrices are not square but they are diagonal:

$$\begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \end{pmatrix}, \quad \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \\ 0 & 0 & 0 \end{pmatrix}.$$



Definition 2.6

A dataset is a collection of points $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ in \mathbb{R}^d . After centering the mean to the origin, the dataset becomes a collection of vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$. Usually the vectors are presented as the rows of an $N \times d$ matrix A .

Corresponding to this, datasets are often provided as a CSV file. The matrix A is the dataset matrix. In excel, this is called a spreadsheet. In SQL, this is called a table. In `numpy`, it's an array. In `pandas`, it's a dataframe. So, effectively,

matrix = dataset = CSV file = spreadsheet = table = array =
dataframe



Example 2.4

For the Iris dataset:

```
1 import numpy as np
2 import pandas as pd
3 from sklearn import datasets
4
5 iris = datasets.load_iris()
6
7 # The dataset
8 dataset = iris["data"]
9
10 # To center the dataset
11 m = np.mean(dataset,axis=0)
12 vectors = dataset - m
13
14 # To make a data frame
15 centered_df = pd.DataFrame(data=vectors)
```



Addition & scalar multiplication

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections

Matrices consisting of numbers are added and multiplied by scalars as follows. With t as an scalar and the matrices

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1d} \\ a_{21} & a_{22} & \dots & a_{2d} \\ \vdots & \vdots & \dots & \vdots \\ a_{N1} & a_{N2} & \dots & a_{Nd} \end{pmatrix} \quad \text{and} \quad A' = \begin{pmatrix} a'_{11} & a'_{12} & \dots & a'_{1d} \\ a'_{21} & a'_{22} & \dots & a'_{2d} \\ \vdots & \vdots & \dots & \vdots \\ a'_{N1} & a'_{N2} & \dots & a'_{Nd} \end{pmatrix}$$

we have

$$A + A' = \begin{pmatrix} a_{11} + a'_{11} & a_{12} + a'_{12} & \dots & a_{1d} + a'_{1d} \\ a_{21} + a'_{21} & a_{22} + a'_{22} & \dots & a_{2d} + a'_{2d} \\ \vdots & \vdots & \dots & \vdots \\ a_{N1} + a'_{N1} & a_{N2} + a'_{N2} & \dots & a_{Nd} + a'_{Nd} \end{pmatrix},$$

and

$$tA = \begin{pmatrix} ta_{11} & ta_{12} & \dots & ta_{1d} \\ ta_{21} & ta_{22} & \dots & ta_{2d} \\ \vdots & \vdots & \dots & \vdots \\ ta_{N1} & ta_{N2} & \dots & ta_{Nd} \end{pmatrix}.$$

Matrices may be added only if they have the same shape.



Addition & scalar multiplication

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.5

```
1  import numpy as np
2
3  A = np.zeros((4,3))
4  print(A)
5  B = np.eye(3)
6  print(B)
7  C = np.eye(4,3)
8  print(C)
9  D = np.array([[1,2,3],[4,5,6],[7,8,9],[10,11,12]])
10 print(D)
11 E = np.diag([1,2,3,4])
12 print(E)
13
14 print(A+C)
15 print(C+D)
16 print(4*D)
17 print(-D)
18 print(-2*D)
```



Introduction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Let t be a scalar, $\mathbf{u}, \mathbf{v}, \mathbf{w}$ be vectors, and let A, B be matrices. We already know how to compute $t\mathbf{u}$, $t\mathbf{v}$, and tA , tB . In this section, we compute the *dot product* $\mathbf{u} \cdot \mathbf{v}$, the *matrix-vector product* $A\mathbf{v}$, and the *matrix-matrix product* AB .



Dot product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

In the first chapter, we defined the dot product in two dimensions. We now generalize it to any dimension d . Suppose \mathbf{u}, \mathbf{v} are vectors in \mathbb{R}^d . Then their dot product $\mathbf{u} \cdot \mathbf{v}$ is the scalar obtained by multiplying corresponding features and then summing the products. **This only works if the dimensions of \mathbf{u} and \mathbf{v} agree.**

In other words, if $\mathbf{u} = (u_1, u_2, \dots, u_d)$ and $\mathbf{v} = (v_1, v_2, \dots, v_d)$, then

$$\mathbf{u} \cdot \mathbf{v} = u_1 v_1 + u_2 v_2 + \dots + u_d v_d.$$

It's best to think of this as "row-times-column" multiplication,

$$\mathbf{u} \cdot \mathbf{v} = \begin{pmatrix} u_1 & u_2 & \cdots & u_d \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_d \end{pmatrix} = u_1 v_1 + u_2 v_2 + \dots + u_d v_d.$$



Dot product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.6

In Python, calculate the dot product of $\mathbf{u} = (1, 2, 3)$ and $\mathbf{v} = (4, 5, 6)$.

```
1  import numpy as np
2
3  u = np.array([1,2,3])
4  v = np.array([4, 5, 6])
5
6  u_dot_v = np.dot(u,v)
7  print(u_dot_v)
8
9  u_dot_v_ = u[0]*v[0] + u[1]*v[1] + u[2]*v[2]
10 print(u_dot_v_)
11
12 print(u_dot_v == u_dot_v_)
```



Dot product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

As we mentioned in 2 dimensions, we have the following generalizations in d dimension:

Definition 2.7

The length or norm or magnitude of a vector \mathbf{v} is the square root of the dot product $\mathbf{v} \cdot \mathbf{v}$,

$$|\mathbf{v}| = \sqrt{\mathbf{v} \cdot \mathbf{v}}$$

Theorem 2.1 (Dot Product)

The dot product $\mathbf{u} \cdot \mathbf{v}$ satisfies

$$\mathbf{u} \cdot \mathbf{v} = |\mathbf{u}||\mathbf{v}| \cos \theta,$$

where θ is the angle between \mathbf{u} and \mathbf{v} .

Corollary 2.1

To calculate the angle θ between \mathbf{u} and \mathbf{v} we have:

$$\cos \theta = \frac{\mathbf{u} \cdot \mathbf{v}}{\sqrt{|\mathbf{u}||\mathbf{v}|}} = \frac{\mathbf{u} \cdot \mathbf{v}}{\sqrt{(\mathbf{u} \cdot \mathbf{u})(\mathbf{v} \cdot \mathbf{v})}}.$$



Dot product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections

Corollary 2.2 (Cauchy-Schwarz Inequality)

The dot product of two vectors is absolutely less or equal to the product of their lengths,

$$|\mathbf{u} \cdot \mathbf{v}| \leq \|\mathbf{u}\| \|\mathbf{v}\| \quad \text{or} \quad |\mathbf{u} \cdot \mathbf{v}| \leq (\mathbf{u} \cdot \mathbf{u})(\mathbf{v} \cdot \mathbf{v}).$$

Definition 2.8

Vectors \mathbf{u} and \mathbf{v} are said to be perpendicular or orthogonal if $|\mathbf{u} \cdot \mathbf{v}| = 0$. A collection of vectors is orthogonal if any pair of vectors in the collection are orthogonal.

Vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are said to be orthonormal if they are both unit vectors and orthogonal.

Exercise 2.1

The zero vector is orthogonal to every vector. The converse is true as well: if a vector is orthogonal to every vector then it is the zero vector.



Matrix-vector product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections

Definition 2.9

Suppose \mathbf{v} is a vector and A is a matrix. If the rows of A have the same dimension as that of \mathbf{v} , we can take the dot product of each row of A with \mathbf{v} , obtaining the matrix-vector product $A\mathbf{v}$: $A\mathbf{v}$ is the vector whose features are the dot products of the rows of A with \mathbf{v} .

Note:

- In Python we use again `np.dot(A, v)` for matrix-vector product.
- If \mathbf{u} and \mathbf{v} are vectors, we can think of \mathbf{u} as a row vector, or a matrix consisting of a single row. With this interpretation, the matrix-vector product $\mathbf{u}\mathbf{v}$ equals the dot product $\mathbf{u} \cdot \mathbf{v}$.
- If \mathbf{u} and \mathbf{v} are vectors, we can think of \mathbf{u} as a column vector, or a matrix consisting of a single column. With this interpretation, \mathbf{u}^t is a single row, and the matrix-vector product $\mathbf{u}^t\mathbf{v}$ equals the dot product $\mathbf{u} \cdot \mathbf{v}$.
- $(A\mathbf{v})^t = \mathbf{v}^t A^t$.
- $(A\mathbf{u}) \cdot \mathbf{v} = \mathbf{u} \cdot (A^t\mathbf{v})$.



Matrix-vector product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.7

Calculate $A\mathbf{v}$, when

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix} \quad \text{and} \quad \mathbf{v} = (1, 2, 3, 4).$$

Answer:

$$\begin{aligned} A\mathbf{v} &= \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \\ &= \begin{pmatrix} (1 \times 1) + (2 \times 2) + (3 \times 3) + (4 \times 4) \\ (5 \times 1) + (6 \times 2) + (7 \times 3) + (8 \times 4) \\ (9 \times 1) + (10 \times 2) + (11 \times 3) + (12 \times 4) \end{pmatrix} = \begin{pmatrix} 30 \\ 70 \\ 110 \end{pmatrix} \end{aligned}$$

```
1 import numpy as np
2
3 A = np.arange(1,13).reshape(3,4)
4 v = np.array([1,2,3,4])
5
6 Av = np.dot(A, v)
7 print(Av)
```




Matrix-matrix product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections

Definition 2.10

Let A and B be two matrices. If the row dimension of A equals the column dimension of B , the matrix-matrix product AB is defined.

When this condition holds, the entries in the matrix AB are the dot products of the rows of A with the columns of B .

Note:

- In Python we use again `np.dot(A,B)` for matrix-vector product.
- $(AB)^t = B^t A^t$.



Matrix-vector product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.8

Calculate AB , when

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 13 & 14 \\ 15 & 16 \\ 17 & 18 \\ 19 & 20 \end{pmatrix}.$$

Answer:

$$\begin{aligned} AB &= \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix} \begin{pmatrix} 13 & 14 \\ 15 & 16 \\ 17 & 18 \\ 19 & 20 \end{pmatrix} \\ &= \begin{pmatrix} (1 \times 13) + (2 \times 15) + (3 \times 17) + (4 \times 19) & (1 \times 14) + (2 \times 16) + (3 \times 18) + (4 \times 20) \\ (5 \times 13) + (6 \times 15) + (7 \times 17) + (8 \times 19) & (5 \times 14) + (6 \times 16) + (7 \times 18) + (8 \times 20) \\ (9 \times 13) + (10 \times 15) + (11 \times 17) + (12 \times 19) & (9 \times 14) + (10 \times 16) + (11 \times 18) + (12 \times 20) \end{pmatrix} \\ &= \begin{pmatrix} 170 & 180 \\ 426 & 452 \\ 682 & 724 \end{pmatrix} \end{aligned}$$

```
1 import numpy as np
2
3 A = np.arange(1,13).reshape(3,4)
4 B = np.arange(13,21).reshape(4,2)
5
6 AB = np.dot(A, B)
7 print(AB)
```



Orthonormal Rows and Columns

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Assume the rows of a matrix A are $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$. Since matrix-matrix multiplication is *row* \times *column*, we have

$$AA^t = \begin{pmatrix} \mathbf{v}_1 \cdot \mathbf{v}_1 & \mathbf{v}_1 \cdot \mathbf{v}_2 & \cdots & \mathbf{v}_1 \cdot \mathbf{v}_N \\ \mathbf{v}_2 \cdot \mathbf{v}_1 & \mathbf{v}_2 \cdot \mathbf{v}_2 & \cdots & \mathbf{v}_2 \cdot \mathbf{v}_N \\ \vdots & \vdots & & \vdots \\ \mathbf{v}_N \cdot \mathbf{v}_1 & \mathbf{v}_N \cdot \mathbf{v}_2 & \cdots & \mathbf{v}_N \cdot \mathbf{v}_N \end{pmatrix}.$$

Corollary 2.3

Let U be a matrix.

- U has orthonormal rows iff $UU^t = I$.
- U has orthonormal columns iff $U^tU = I$.



Tensor product

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.11

If \mathbf{u} and \mathbf{v} are vectors, the tensor product $\mathbf{u} \otimes \mathbf{v}$ is the matrix-matrix product $\mathbf{u}^t \mathbf{v}$, with \mathbf{u} and \mathbf{v} row vectors. If \mathbf{u} is N -dimensional and \mathbf{v} is d -dimensional, then $\mathbf{u} \otimes \mathbf{v}$ is an $N \times d$ matrix.

Example 2.9

if $\mathbf{u} = (a, b, c)$ and $\mathbf{v} = (\alpha, \beta)$, then

$$\mathbf{u} \otimes \mathbf{v} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \begin{pmatrix} \alpha & \beta \end{pmatrix} = \begin{pmatrix} a\alpha & a\beta \\ b\alpha & b\beta \\ c\alpha & c\beta \end{pmatrix}.$$

Using the tensor product, we have

Theorem 2.2 (Tensor Identity)

Let A be a matrix with rows $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$. Then

$$A^t A = \mathbf{v}_1 \otimes \mathbf{v}_1 + \mathbf{v}_2 \otimes \mathbf{v}_2 + \dots + \mathbf{v}_N \otimes \mathbf{v}_N.$$

Exercise 2.2

Prove the tensor identity.



Some definitions

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.12

A matrix Q is symmetric if $Q = Q^t$.

For any matrix A , $Q = AA^t$ and $Q = A^t A$ are symmetric.

A symmetric matrix Q satisfying $\mathbf{v} \cdot Q\mathbf{v} \geq 0$ for every vector \mathbf{v} is nonnegative.

A symmetric matrix Q satisfying $\mathbf{v} \cdot Q\mathbf{v} > 0$ for every nonzero vector \mathbf{v} is positive.

Definition 2.13

The trace of a square matrix is the sum of its diagonal elements.

Even though in general $AB \neq BA$, it is always true that

Exercise 2.3

$$\text{trace}(AB) = \text{trace}(BA).$$

Exercise 2.4

$$\mathbf{u} \cdot Q\mathbf{v} = \text{trace}(Q(\mathbf{v} \otimes \mathbf{u})).$$



Norm squared

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.14

If $A = (a_{ij})$ is any matrix, then the norm squared of A is

$$\| A \|^2 = \sum_{i,j} a_{ij}^2.$$

Theorem 2.3 (Norm Squared of Matrix)

Let A be a matrix with rows $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$. Then

$$\| A \|^2 = |\mathbf{v}_1|^2 + |\mathbf{v}_2|^2 + \dots + |\mathbf{v}_N|^2,$$

and

$$\| A \|^2 = \text{trace}(A^t A).$$

Exercise 2.5

Prove Theorem (2.3).



If $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ is a dataset of points in \mathbb{R}^d with mean \mathbf{m} , and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ is the corresponding centered dataset, then we saw that the covariance matrix Q is the average of tensor products

$$Q = \frac{\mathbf{v}_1 \otimes \mathbf{v}_1 + \mathbf{v}_2 \otimes \mathbf{v}_2 + \dots + \mathbf{v}_N \otimes \mathbf{v}_N}{N}.$$

Let A be the matrix with rows $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$. By Theorem (2.2), the last equation is the same as

$$Q = \frac{1}{N} A^t A.$$



Example 2.10

Calculate the mean, covariance and total variance of the Iris dataset.

```
1  import numpy as np
2  from sklearn import datasets
3
4  iris = datasets.load_iris()
5
6  # The dataset
7  dataset = iris["data"]
8
9  # Mean
10 m = np.mean(dataset, axis=0)
11
12 # Centered dataset
13 vectors = dataset - m
14
15 # Covariance
16 N = len(vectors)
17 # Biased
18 Q = np.dot(vectors.T, vectors)/N
19 Q = np.cov(dataset, rowvar=False, ddof=0) # ddof = delta degrees of freedom
20 Q = np.cov(dataset.T, ddof=0)
21
22 # Unbiased
23 Q = np.dot(vectors.T, vectors)/(N-1)
24 Q = np.cov(dataset, rowvar=False)
25 Q = np.cov(dataset.T)
26
27 # Total Variance
28 TV = np.trace(Q)
```




Standardized dataset

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Let $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ is a dataset of points in \mathbb{R}^d . Each sample point \mathbf{x} has d features (t_1, t_2, \dots, t_d) . We compute the variance of each feature separately.

Let $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_d$ be the standard basis in \mathbb{R}^d , and, for each $j = 1, 2, \dots, d$, project the dataset onto \mathbf{e}_j , obtaining the scalar dataset $\mathbf{x}_1 \cdot \mathbf{e}_j, \mathbf{x}_2 \cdot \mathbf{e}_j, \dots, \mathbf{x}_N \cdot \mathbf{e}_j$, consisting of the j -th feature of the samples. If q_{jj} is the variance of this scalar dataset, then $q_{11}, q_{22}, \dots, q_{dd}$ are the diagonal entries of the covariance matrix.

To standardize the dataset, we center it, and rescale the features to have variance one, as follows. Let $\mathbf{m} = (m_1, m_2, \dots, m_d)$ be the dataset mean. For each sample point $\mathbf{x} = (t_1, t_2, \dots, t_d)$, the standardized vector is

$$\mathbf{v} = \left(\frac{t_1 - m_1}{\sqrt{q_{11}}}, \frac{t_2 - m_2}{\sqrt{q_{22}}}, \dots, \frac{t_d - m_d}{\sqrt{q_{dd}}} \right).$$

Then the standardized dataset is $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$.



Definition 2.15

If $Q = (q_{ij})$ is the covariance matrix, then the correlation matrix is the $d \times d$ matrix $Q' = (q'_{ij})$ with entries

$$q'_{ij} = \frac{q_{ij}}{\sqrt{q_{ii}q_{jj}}}, \quad i, j = 1, 2, \dots, d.$$

Theorem 2.4 (Standardized Covariance Equals Correlation)

The covariance matrix of the standardized dataset equals the correlation matrix of the original dataset.

Exercise 2.6

Prove Theorem (2.4).



Example 2.11

For the Iris dataset check Theorem (2.4).

```
1  import numpy as np
2  from sklearn import datasets
3  from sklearn.preprocessing import StandardScaler
4
5  iris = datasets.load_iris()
6
7  # The dataset
8  dataset = iris["data"]
9
10 # standardize dataset
11 vectors = StandardScaler().fit_transform(dataset)
12 Qcorr = np.corrcoef(dataset.T)
13 Qcov = np.cov(vectors.T, bias=True)
14 np.allclose(Qcov, Qcorr)
```



Matrix Invers

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.16

Given a square matrix A , the inverse matrix is the matrix B satisfying

$$AB = I = BA.$$

When A has an inverse, we say A is invertible. If a matrix is $d \times d$, then the inverse is also $d \times d$. We write $B = A^{-1}$ for the inverse matrix of A .

Here I is the identity matrix. **Not every square matrix has an inverse.**
For example, the zero matrix does not have an inverse.

Example 2.12

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \Rightarrow A^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.$$

Since we can't divide by zero, a 2×2 matrix is invertible only if $ad - bc \neq 0$.



Notes

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Exercise 2.7

Prove that $(AB)^{-1} = B^{-1}A^{-1}$.

Exercise 2.8

Prove that for a linear system $A\mathbf{x} = \mathbf{b}$, if A is invertible then $\mathbf{x} = A^{-1}\mathbf{b}$.

Example 2.13

Solve the following linear system

$$\begin{cases} x + 2y + 3z = 1 \\ -3x + 6y = 2 \\ 10x - 5y + 23z = 3 \end{cases}$$

```

1  import numpy as np
2
3  A = np.array([[1,2,3],[-3,6,0],[10,-5,23]])
4  b = np.array([1,2,3])
5  # Determinant of A
6  np.linalg.det(A)
7  # Inverse of A
8  np.linalg.inv(A)
9  # Solution of Ax=b
10 x = np.dot(np.linalg.inv(A),b)

```



Linear combination

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections

Definition 2.17 (Linear combination)

A linear combination of vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ is

$$t_1\mathbf{v}_1 + t_2\mathbf{v}_2 + \dots + t_d\mathbf{v}_d,$$

where t_1, t_2, \dots, t_d are scalars.

Example 2.14

Let $\mathbf{u}, \mathbf{v}, \mathbf{w}$ be three vectors. Then

$$3\mathbf{u} - \frac{1}{6}\mathbf{v} + 9\mathbf{w}, \quad 5\mathbf{u} + 0\mathbf{v} - \mathbf{w}, \quad 0\mathbf{u} + 0\mathbf{v} + 0\mathbf{w},$$

are linear combinations of $\mathbf{u}, \mathbf{v}, \mathbf{w}$.

Example 2.15

Let A be a matrix with columns $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$, and let $\mathbf{x} = (t_1, t_2, \dots, t_d)$. Then $A\mathbf{x}$ is a linear combination of the columns of A as:

$$A\mathbf{x} = t_1\mathbf{v}_1 + t_2\mathbf{v}_2 + \dots + t_d\mathbf{v}_d.$$



Definition 2.18 (Span)

The span of $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ is the set S of all linear combinations of $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$, and we write

$$S = \text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d).$$

Exercise 2.9

Let A be the matrix with columns $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$. Then $S = \text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d)$ is the set S of all vectors of the form $A\mathbf{x}$.

Exercise 2.10

If each vector \mathbf{v}_k of $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ is a linear combination of vectors $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N$, then

$$\text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d) \subseteq \text{span}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N).$$



Column space

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance

Directions

Pseudo-Inverse

Projections

Definition 2.19

Let A be a matrix. The column space of A is the span of its columns.

Example 2.16

```

1  import sympy as sp
2  import scipy as sc
3  import numpy as np
4
5  A = sp.Matrix([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
6  A
7  # column vectors
8  u = sp.Matrix([1,2,3,4,5])
9  v = sp.Matrix([6,7,8,9,10])
10 w = sp.Matrix([11,12,13,14,15])
11 A = sp.Matrix.hstack(u,v,w)
12 A
13 # returns minimal spanning set for column space of A
14 A.columnspace()
15 # returns minimal spanning orthonormal set for column space of A
16 A = np.array([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
17 sc.linalg.orth(A)

```

`A.columnspace()` returns a minimal set of vectors spanning the column space of A . The **column rank** of A is the number of vectors returned: for A in the above example, the column rank is 2. `sc.linalg.orth(A)` returns a minimal orthonormal set of vectors spanning the column space of A .



Notes

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Exercise 2.11

As in example 2.16, show that if

$$\mathbf{v}_1 = (1, 2, 3, 4, 5), \quad \mathbf{v}_2 = (6, 7, 8, 9, 10), \quad \mathbf{v}_3 = (11, 12, 13, 14, 15)$$

then $\text{span}(\mathbf{v}_1, \mathbf{v}_2) = \text{span}(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$.

Exercise 2.12

Show that: the column space of a matrix A consists of all vectors of the form $A\mathbf{x}$. A vector \mathbf{b} is in the column space of A when $A\mathbf{x} = \mathbf{b}$ has a solution.

The augmented matrix $\bar{A} = (A, \mathbf{b})$ is obtained by adding \mathbf{b} as an extra column next to the columns of A .

Exercise 2.13

Let \bar{A} be the matrix A augmented by a vector \mathbf{b} . Then \mathbf{b} is in the column space of A iff

$$\text{column rank}(A) = \text{column rank}(\bar{A}).$$



Exercise 2.14

Show that the vectors

$$\mathbf{e}_1 = (1, 0, 0, \dots, 0, 0)$$

$$\mathbf{e}_2 = (0, 1, 0, \dots, 0, 0)$$

$$\mathbf{e}_3 = (0, 0, 1, \dots, 0, 0)$$

$$\vdots$$

$$\mathbf{e}_d = (0, 0, 0, \dots, 0, 1)$$

span \mathbb{R}^d .

The set $\{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_d\}$ is the *standard basis* for \mathbb{R}^d .



Row space

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.20

The row space of a matrix is the span of its rows.

Example 2.17

```
1  import sympy as sp
2  import scipy as sc
3  import numpy as np
4
5  A = sp.Matrix([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
6  A
7
8  # returns minimal spanning set for row space of A
9  A.rowspace()
10
11 # returns minimal spanning orthonormal set for column space of A
12 A = np.array([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
13 sc.linalg.orth(A.T)
```

The **row rank** of a matrix is the number of vectors returned by `A.rowspace()`. This is the minimal number of vectors spanning the row space of A which for the above example is 2. `sc.linalg.orth(A.T)` returns a minimal orthonormal set of vectors spanning the row space of A .



Linearly independence

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.21

A linear combination $t_1\mathbf{v}_1 + t_2\mathbf{v}_2 + \dots + t_d\mathbf{v}_d$ is trivial if all the coefficients are zero: $t_1 = t_2 = \dots = t_d = 0$. Otherwise it is non-trivial: if at least one coefficient is not zero.

A linear combination $t_1\mathbf{v}_1 + t_2\mathbf{v}_2 + \dots + t_d\mathbf{v}_d$ vanishes if it equals the zero vector:

$$t_1\mathbf{v}_1 + t_2\mathbf{v}_2 + \dots + t_d\mathbf{v}_d = \mathbf{0}.$$

We say $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ are linearly dependent if there is a non-trivial vanishing linear combination of $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$. Otherwise, we say $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ are linearly independent.

Example 2.18

The vectors $\mathbf{v}_1 = (1, 2, 3, 4, 5)$, $\mathbf{v}_2 = (6, 7, 8, 9, 10)$, $\mathbf{v}_3 = (11, 12, 13, 14, 15)$ are linearly dependent, because

$$\mathbf{v}_3 + \mathbf{v}_1 - 2\mathbf{v}_2 = \mathbf{0}.$$

We can see $\mathbf{v}_3 = 2\mathbf{v}_2 - \mathbf{v}_1$.



Exercise 2.15

Show that if $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ are linearly dependent then at least one of the vectors is a linear combination of the remaining vectors.

Exercise 2.16 (Homogeneous Linear Systems)

Let A be the matrix with columns $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$. Then

$\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$

- are linearly dependent when $A\mathbf{x} = \mathbf{0}$ has a nonzero solution \mathbf{x} , and*
- are linearly independent when $A\mathbf{x} = \mathbf{0}$ has only the zero solution $\mathbf{x} = \mathbf{0}$.*

Exercise 2.17

Show that if $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$ are orthonormal then they are linearly independent.



Null space

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.22

The set of vectors \mathbf{x} satisfying $A\mathbf{x} = \mathbf{0}$, or the set of solutions \mathbf{x} of $A\mathbf{x} = \mathbf{0}$, is the null space of the matrix A .

The cardinality of a minimal set of vectors spanning the null space of A is called the nullity of A .

Example 2.19

Show that the nullity of the following matrix is 1.

$$A = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix}.$$

```
1 import sympy as sp
2 import scipy as sc
3 import numpy as np
4
5 # using sympy
6 A = sp.Matrix([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
7 A.nullspace()
8
9 # using numpy and scipy
10 A = np.array([[1, 6, 11], [2, 7, 12], [3, 8, 13], [4, 9, 14], [5, 10, 15]])
11 sc.linalg.null_space(A)
```



Exercise 2.18

Let A be any matrix. Show that the null space, row space and column space of A equals the null space, row space and column space of $A^t A$, respectively.

Definition 2.23 (Orthogonal complements)

Let S and T be spans. We say S and T are orthogonal complements if every vector in S is orthogonal to every vector in T . In symbols, we write $S = T^\perp$ and $T = S^\perp$ (pronounced "T-perp" and "S-perp").

Exercise 2.19

Show that, if $S = \text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N)$, and A is the matrix with rows $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$, then S^\perp equals the null space of A .

Exercise 2.20

For a matrix A , show that $(\text{nullspace}^\perp = \text{rowspace})$ and $(\text{rowspace}^\perp = \text{nullspace})$



Subspace

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance

Directions

Pseudo-Inverse

Projections

Definition 2.24 (Subspace)

A subspace is a set of vectors closed under addition and scalar multiplication. precisely: A subset $S \subseteq V$ is a subspace of the vector space V whenever for every $\mathbf{x}_1, \mathbf{x}_2 \in S$ and every scalar t we have

- $\mathbf{x}_1 + \mathbf{x}_2 \in S$ and
- $t\mathbf{x}_1 \in S$.

or equivalently: $t\mathbf{x}_1 + \mathbf{x}_2 \in S$.

Exercise 2.21

If V is a vector space then \emptyset and V are the trivial subspaces of V .

Exercise 2.22

Show that

- *the null space: all \mathbf{x} 's satisfying $A\mathbf{x} = \mathbf{0}$,*
- *the row space: the span of the rows of A , and*
- *the column space: the span of the columns of A*

are subspaces, but

- *the solution space: the solutions \mathbf{x} of $A\mathbf{x} = \mathbf{b}$*

is not a subspace.



Projected dataset

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Let $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ be the centered dataset of the dataset $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ in \mathbb{R}^d with mean \mathbf{m} . Then the covariance is

$$Q = \frac{\mathbf{v}_1 \otimes \mathbf{v}_1 + \mathbf{v}_2 \otimes \mathbf{v}_2 + \dots + \mathbf{v}_N \otimes \mathbf{v}_N}{N} = \frac{1}{N} A^t A,$$

where A is the matrix with rows $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$.

If \mathbf{b} is a vector, the projection of the centered dataset onto the line through \mathbf{b} results in the reduced dataset

$$\mathbf{v}_1 \cdot \mathbf{b}, \mathbf{v}_2 \cdot \mathbf{b}, \dots, \mathbf{v}_N \cdot \mathbf{b}.$$

The mean of this projected dataset is zero, and its variance is

$$\frac{(\mathbf{v}_1 \cdot \mathbf{b})^2 + (\mathbf{v}_2 \cdot \mathbf{b})^2 + \dots + (\mathbf{v}_N \cdot \mathbf{b})^2}{N} = \frac{1}{N} \mathbf{b}^t A^t A \mathbf{b} = \mathbf{b} \cdot Q \mathbf{b}.$$



Zero variance direction

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear

Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.25

Let \mathbf{m} be a point in \mathbb{R}^d and \mathbf{b} a vector in \mathbb{R}^d . The hyperplane passing through \mathbf{m} and orthogonal to \mathbf{b} is the set of points \mathbf{x} satisfying the equation

$$\mathbf{b} \cdot (\mathbf{x} - \mathbf{m}) = 0.$$

Example 2.20

In \mathbb{R}^3 , a hyperplane is a plane, and in \mathbb{R}^2 , a hyperplane is a line. In general, in \mathbb{R}^d , a hyperplane is $(d - 1)$ -dimensional, always one less than the ambient dimension.



Definition 2.26

A vector \mathbf{b} is a zero variance direction of Q if the projected variance is zero:

$$\mathbf{b} \cdot Q\mathbf{b} = 0.$$

Theorem 2.5

Let \mathbf{m} and Q be the mean and covariance of a dataset in \mathbb{R}^d . Then $\mathbf{b} \cdot Q\mathbf{b} = 0$ is the same as saying every point in the dataset lies in the hyperplane passing through \mathbf{m} and orthogonal to \mathbf{b} : $\mathbf{b} \cdot (\mathbf{x} - \mathbf{m}) = 0$.

Theorem 2.6

Let Q be a covariance matrix. Then the null space of Q equals the zero variance directions of Q .

Corollary 2.4

Let Q be a covariance matrix of a centered dataset A . Then the null space of A equals the zero variance directions of Q .



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.21

Suppose the dataset

	t_1	t_2	t_3	t_4	t_5
\mathbf{x}_1	1	2	3	4	5
\mathbf{x}_2	6	7	8	9	10
\mathbf{x}_3	11	12	13	14	15
\mathbf{x}_4	16	17	18	19	20

Here we have 5 features. By the following code the null space of the covariance matrix, say Q , has 4 vectors which means it is 4-dimensional (or the nullity of Q is 4). Hence the dataset is a 1-dimensional dataset ($5 - 4 = 1$). It means that there is a hyperplane (here a line) in \mathbb{R}^5 which we can project the dataset on it without losing any information.

```
1 import numpy as np
2 import scipy as sc
3
4 dataset = np.array([ [1,2,3,4,5], [6,7,8,9,10], [11,12,13,14,15], [16,17,18,19,20] ])
5 Q = np.cov(dataset.T)
6 N = sc.linalg.null_space(Q)
7 nullity = N.shape[1]
```



Concept

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

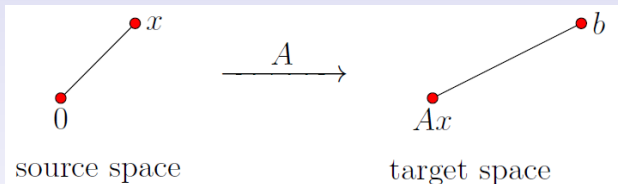
Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Think of \mathbf{b} and $A\mathbf{x}$ as points, and measure the distance between them, and think of \mathbf{x} and the origin $\mathbf{0}$ as points, and measure the distance between them.



If $A\mathbf{x} = \mathbf{b}$ is solvable, then, among all solutions \mathbf{x}^* , select the solution \mathbf{x}^+ closest to $\mathbf{0}$. More generally, if $A\mathbf{x} = \mathbf{b}$ is not solvable, select the points \mathbf{x}^* so that $A\mathbf{x}^*$ is closest to \mathbf{b} , then, among all such \mathbf{x}^* , select the point \mathbf{x}^+ closest to the origin.



Concept

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

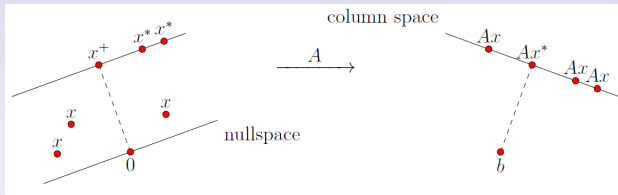
Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections



Even though the point x^+ may not solve $Ax = b$, this procedure results in a uniquely determined x^+ : While there may be several points x^* , there is only one x^+ .



Concept

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

The results in this section are as follows. Let A be any matrix. There is a unique matrix A^+ — the *pseudo-inverse* of A — with the following properties:

- the linear system $A\mathbf{x} = \mathbf{b}$ is solvable, when $\mathbf{b} = AA^+\mathbf{b}$.
- $\mathbf{x}^+ = A^+\mathbf{b}$ is a solution of
 - 1 the linear system $A\mathbf{x} = \mathbf{b}$, if $A\mathbf{x} = \mathbf{b}$ is solvable.
 - 2 the *regression equation* $A^t A\mathbf{x} = A^t \mathbf{b}$, always.
- In either case,
 - 1 there is exactly one solution with minimum norm.
 - 2 Among all solutions, \mathbf{x}^+ has minimum norm.
 - 3 Every other solution is $\mathbf{x} = \mathbf{x}^+ + \mathbf{v}$ for \mathbf{v} in the null space of A .

Key concepts in this section are the *residual*

$$|A\mathbf{x} - \mathbf{b}|^2$$

and the *regression equation*

$$A^t A\mathbf{x} = A^t \mathbf{b}.$$

Exercise 2.23

\mathbf{x} is a solution of $A\mathbf{x} = \mathbf{b}$ iff the residual is zero.



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.22

For A and \mathbf{b} as below

$$A = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} -9 \\ -3 \\ 3 \\ 9 \\ 10 \end{pmatrix},$$

the linear system $A\mathbf{x} = \mathbf{b}$ and the regression equation $A^t A\mathbf{x} = A^t \mathbf{b}$ are

$$\begin{cases} x + 6y + 11z = -9 \\ 2x + 7y + 12z = -3 \\ 3x + 8y + 13z = 3 \\ 4x + 9y + 14z = 9 \\ 5x + 10y + 15z = 10 \end{cases}, \quad \begin{cases} 11x + 26y + 41z = 16 \\ 13x + 33y + 53z = 13 \\ 41x + 106y + 171z = 36 \end{cases},$$

respectively.



Residual minimizer

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Let \mathbf{b} be any vector, not necessarily in the column space of A . To see how close we can get to solving $A\mathbf{x} = \mathbf{b}$, we minimize the residual $|A\mathbf{x} - \mathbf{b}|^2$.

Definition 2.27 (Residual minimizer)

We say \mathbf{x}^ is a residual minimizer if*

$$|A\mathbf{x}^* - \mathbf{b}|^2 = \min_x |A\mathbf{x} - \mathbf{b}|^2.$$

Theorem 2.7 (Existence of Residual Minimizer)

There is a residual minimizer \mathbf{x}^ in the row space of A .*

Exercise 2.24

Prove Theorem 2.7.



Residual minimizer

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Theorem 2.8

\mathbf{x}^* is a residual minimizer iff \mathbf{x}^* solves the regression equation.

Proof: let \mathbf{v} be any vector, and t a scalar. Insert $\mathbf{x} = \mathbf{x}^* + t\mathbf{v}$ into the residual:

$$\begin{aligned} |A\mathbf{x} - \mathbf{b}|^2 &= |A(\mathbf{x}^* + t\mathbf{v}) - \mathbf{b}|^2 \\ &= |(A\mathbf{x}^* - \mathbf{b}) + At\mathbf{v}|^2 \\ &= |A\mathbf{x}^* - \mathbf{b}|^2 + 2t(A\mathbf{x}^* - \mathbf{b}) \cdot A\mathbf{v} + t^2|A\mathbf{v}|^2 \\ &:= f(t). \end{aligned}$$

If \mathbf{x}^* is a residual minimizer, then $f(t)$ is minimized when $t = 0$. But a parabola $f(t) = a + 2bt + ct^2$ is minimized at $t = 0$ only when $b = 0$. Thus the linear coefficient vanishes, $(A\mathbf{x}^* - \mathbf{b}) \cdot A\mathbf{v} = 0$. This implies

$$A^t(A\mathbf{x}^* - \mathbf{b}) \cdot \mathbf{v} = (A\mathbf{x}^* - \mathbf{b}) \cdot A\mathbf{v} = 0.$$

Since \mathbf{v} is any vector, this implies

$$A^t(A\mathbf{x}^* - \mathbf{b}) = \mathbf{0},$$

which is the regression equation. Conversely, if the regression equation holds, then the linear coefficient in the parabola $f(t)$ vanishes, so $t = 0$ is a minimum, establishing that \mathbf{x}^* is a residual minimizer. □



Exercise 2.25

Any two residual minimizers differ by a vector in the nullspace of A .

Definition 2.28

We say \mathbf{x}^+ is a minimum norm residual minimizer if \mathbf{x}^+ is a residual minimizer and

$$|\mathbf{x}^+|^2 \leq |\mathbf{x}^*|^2$$

for any residual minimizer \mathbf{x}^ .*

Theorem 2.9

Let \mathbf{x}^ be a residual minimizer. Then \mathbf{x}^* is a minimum norm residual minimizer iff \mathbf{x}^* is in the row space of A .*

Exercise 2.26

Prove Theorem 2.9.



Residual minimizer

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Theorem 2.10 (Uniqueness of Residual Minimizer)

There is exactly one minimum norm residual minimizer \mathbf{x}^+ .

Proof: If \mathbf{x}_1^+ and \mathbf{x}_2^+ are minimum norm residual minimizers, then $\mathbf{v} = \mathbf{x}_1^+ - \mathbf{x}_2^+$ is both in the row space and in the null space of A , $\mathbf{x}_1^+ - \mathbf{x}_2^+ = \mathbf{0}$. Hence $\mathbf{x}_1^+ = \mathbf{x}_2^+$. □

Putting the above all together, each vector \mathbf{b} leads to a unique \mathbf{x}^+ .
Defining A^+ by setting

$$\mathbf{x}^+ = A^+ \mathbf{b},$$

we obtain A^+ , the pseudo-inverse of A .

Notice if A is, for example, 5×4 , then $A\mathbf{x} = \mathbf{b}$ implies \mathbf{x} is a 4-vector and \mathbf{b} is a 5-vector. Then from $\mathbf{x}^+ = A^+ \mathbf{b}$, it follows A^+ is 4×5 . Thus the shape of A^+ equals the shape of A^t .

Theorem 2.11 (Regression Equation is Always Solvable)

The regression equation $A^t A \mathbf{x} = A^t \mathbf{b}$ is always solvable. The solution of minimum norm is $\mathbf{x}^+ = A^+ \mathbf{b}$. Any other solution differs by a vector in the null space of A .



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.23

For A and \mathbf{b} as below

$$A = \begin{pmatrix} 1 & 6 & 11 \\ 2 & 7 & 12 \\ 3 & 8 & 13 \\ 4 & 9 & 14 \\ 5 & 10 & 15 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} -9 \\ -3 \\ 3 \\ 9 \\ 10 \end{pmatrix},$$

the minimum norm solution of the regression equation $A^t A \mathbf{x} = A^t \mathbf{b}$ is

$$\mathbf{x}^+ = A^+ \mathbf{b} = \frac{1}{150} \begin{pmatrix} -37 & -20 & -3 & 14 & 31 \\ -10 & -5 & 0 & 5 & 10 \\ 17 & 10 & 3 & -4 & -11 \end{pmatrix} \begin{pmatrix} -9 \\ -3 \\ 3 \\ 9 \\ 10 \end{pmatrix} = \frac{1}{15} \begin{pmatrix} 82 \\ 25 \\ -32 \end{pmatrix}.$$

```

1  import sympy as sm
2
3  u = sm.Matrix([1,2,3,4,5])
4  v = sm.Matrix([6,7,8,9,10])
5  w = sm.Matrix([11,12,13,14,15])
6  A = sm.Matrix.hstack(u,v,w)
7
8  A.pinv()
9
10 b = sm.Matrix([-9,-3,3,9,10])
11 A.pinv()*b

```



Linear systems

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Returning to the linear system, we have

Theorem 2.12

If the linear system is solvable, then every solution of the regression equation is a solution of the linear system, and vice-versa.

Corollary 2.5

The linear system $A\mathbf{x} = \mathbf{b}$ is solvable iff $\mathbf{b} = AA^+\mathbf{b}$. When this happens, $\mathbf{x}^+ = A^+\mathbf{b}$ is the solution of minimum norm.

Example 2.24

For A and \mathbf{b} as in Example 2.23, since

$$AA^+\mathbf{b} = \begin{pmatrix} -8 \\ -3 \\ 2 \\ 7 \\ 12 \end{pmatrix}$$

is not equal to \mathbf{b} , the linear system $A\mathbf{x} = \mathbf{b}$ is not solvable.



Corollary 2.6

If A is invertible, then $A^+ = A^{-1}$.

Theorem 2.13 (Properties of Pseudo-Inverse)

- 1 $AA^+A = A$.
- 2 $A^+AA^+ = A^+$.
- 3 AA^+ and A^+A are symmetric.
- 4 *If A has orthonormal columns or orthonormal rows, then $A^+ = A^t$.*

Exercise 2.27

Prove Theorem 2.12, Corollary 2.5, Corollary 2.6 and Theorem 2.13.



Projection onto a line

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

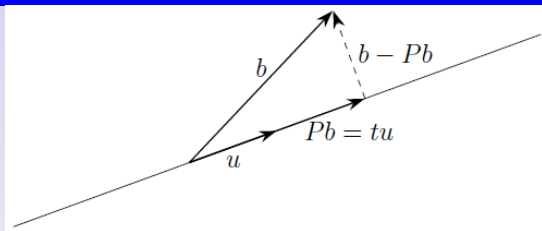
Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections



Let \mathbf{u} be a unit vector, and let \mathbf{b} be any vector. Let $\text{span}(\mathbf{u})$ be the line through \mathbf{u} . The *projection* of \mathbf{b} onto $\text{span}(\mathbf{u})$ is the vector \mathbf{v} in $\text{span}(\mathbf{u})$ that is closest to \mathbf{b} (**Exercise**). It turns out this closest vector \mathbf{v} equals $P\mathbf{b}$ for some matrix P , the *projection matrix*. Since $\text{span}(\mathbf{u})$ is a line, the projected vector $P\mathbf{b}$ is a multiple $t\mathbf{u}$ of \mathbf{u} . We have $\mathbf{b} - P\mathbf{b}$ is orthogonal to \mathbf{u} , so

$$0 = (\mathbf{b} - P\mathbf{b}) \cdot \mathbf{u} = \mathbf{b} \cdot \mathbf{u} - P\mathbf{b} \cdot \mathbf{u} = \mathbf{b} \cdot \mathbf{u} - t\mathbf{u} \cdot \mathbf{u} = \mathbf{b} \cdot \mathbf{u} - t.$$

Solving for t , this implies $t = \mathbf{b} \cdot \mathbf{u}$. If U is the matrix with column \mathbf{u}

$$P\mathbf{b} = (\mathbf{b} \cdot \mathbf{u})\mathbf{u} = (\mathbf{u} \otimes \mathbf{u})\mathbf{b} = UU^t\mathbf{b}.$$

We call $\mathbf{b} \cdot \mathbf{u} = U^t\mathbf{b}$ the *reduced vector*.



Projecting onto a plane

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

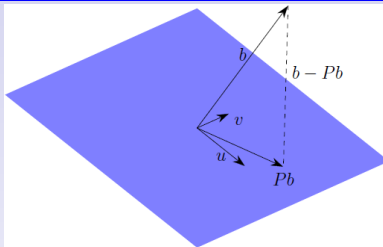
Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse
Projections



Let \mathbf{u}, \mathbf{v} be an orthonormal pair of vectors, so $\mathbf{u} \cdot \mathbf{v} = 0$, $\mathbf{u} \cdot \mathbf{u} = 1 = \mathbf{v} \cdot \mathbf{v}$. We project a vector \mathbf{b} onto $\text{span}(\mathbf{u}, \mathbf{v})$. As before, there is a matrix P , the *projection matrix*, such that the projection of \mathbf{b} onto the plane equals $P\mathbf{b}$. Then $\mathbf{b} - P\mathbf{b}$ is orthogonal to the plane:

$$(\mathbf{b} - P\mathbf{b}) \cdot \mathbf{u} = 0 \quad \text{and} \quad (\mathbf{b} - P\mathbf{b}) \cdot \mathbf{v} = 0.$$

Since $P\mathbf{b}$ lies in the plane, $P\mathbf{b} = r\mathbf{u} + s\mathbf{v}$ is a linear combination of \mathbf{u} and \mathbf{v} . So:

$$r = \mathbf{b} \cdot \mathbf{u}, \quad s = \mathbf{b} \cdot \mathbf{v}.$$

If U is the matrix with columns \mathbf{u}, \mathbf{v} , then

$$P\mathbf{b} = (\mathbf{b} \cdot \mathbf{u})\mathbf{u} + (\mathbf{b} \cdot \mathbf{v})\mathbf{v} = (\mathbf{u} \otimes \mathbf{u} + \mathbf{v} \otimes \mathbf{v})\mathbf{b} = UU^t\mathbf{b}.$$

We call $(\mathbf{b} \cdot \mathbf{u}, \mathbf{b} \cdot \mathbf{v}) = U^t\mathbf{b}$ the *reduced vector*.



Projection matrices in general

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Definition 2.29

Let S be a span. A matrix P is a projection matrix onto S if

- 1 $P\mathbf{b}$ is in S for any vector \mathbf{b} ,
- 2 $P\mathbf{b} = \mathbf{b}$ if \mathbf{b} is in S ,
- 3 $\mathbf{b} - P\mathbf{b}$ is orthogonal to S for any vector \mathbf{b} .

Exercise 2.28

Show that, the projection of a vector onto a span equals the vector itself when the vector is already in the span.

Theorem 2.14 (Projection Onto a Column Space)

Let A be a matrix and \mathbf{v} a vector. Then the projected vector onto the column space of A is $P\mathbf{v} = AA^+\mathbf{v}$ and the reduced vector is $\mathbf{x} = A^+\mathbf{v}$.

Theorem 2.15 (Projection Onto a Row Space)

Let A be a matrix and \mathbf{v} a vector. Then the projected vector onto the row space of A is $P\mathbf{v} = A^+A\mathbf{v}$.

Exercise 2.29

Prove Theorems 2.14 and 2.15.



Example

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Example 2.25

```
1  import numpy as np
2
3  # projection of column vector b onto column space of A
4  def project_col(A,b):
5      Aplus = np.linalg.pinv(A)
6      x = np.dot(Aplus,b) # reduced
7      return np.dot(A,x) # projected
8
9  # projection of column vector b onto row space of A
10 def project_row(A,b):
11     Aplus = np.linalg.pinv(A)
12     AplusA = np.dot(Aplus,A)
13     return np.dot(AplusA,b) # projected
14
15 A = np.array([[1,6,11],[2,7,12],[3,8,13],[4,9,14],[5,10
16               ,15]])
17
18 b = np.array([-9,-3,3,9,10])
19 project_col(A, b.T)
20
21 b = np.array([-9,-3,3])
22 project_row(A, b.T)
```



Projecting onto Orthonormal Vectors

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Theorem 2.16 (Projection Onto Orthonormal Vectors)

If the columns of U are orthonormal and \mathbf{v} is a vector. Then the projected vector onto the column space of U is $P\mathbf{v} = UU^t\mathbf{v}$ and the reduced vector is $\mathbf{x} = U^t\mathbf{v}$.

Example 2.26

```
1  import numpy as np
2
3  # projection of column vector b onto column space of U
4  # with orthonormal columns
5  def project_col_ortho(U,b):
6      x = np.dot(U.T,b) # reduced
7      return np.dot(U,x) # projected
8
9  # Matrices with orthonormal columns
10 U1 = np.array([[1,0,0],[0,1,0],[0,0,1]])
11 U2 = np.array([[1,1,1]/np.sqrt(3),[1,-1,0]/np.sqrt(2),[1,1,-2]/np.sqrt(6)])
12 U3 = np.array([[1,0,0],[0,1,0],[0,0,1],[0,0,0],[0,0,0]])
13
14 b = np.array([1,2,3]).reshape(3,1)
15
16 project_col_ortho(U1, b)
17 project_col_ortho(U2, b)
18
19 b = np.array([1,2,3,4,5]).reshape(5,1)
20
21 project_col_ortho(U3, b)
```



Direct sum

Math for Data

Dr. S. M.
Moosavi

Data Sets

Linear Geometry

Matrices

Products

Matrix Inverse

Span and Linear
Independence

Zero Variance
Directions

Pseudo-Inverse

Projections

Let S and T be spans. Let $S + T$ consist of all sums of vectors $\mathbf{u} + \mathbf{v}$ with \mathbf{u} in S and \mathbf{u} in T . Then a moment's thought shows $S + T$ is itself a span. When the intersection of S and T is the zero vector, we write $S \oplus T$, and we say $S \oplus T$ is the *direct sum* of S and T .

Theorem 2.17

If S is a span in \mathbb{R}^d , then

$$\mathbb{R}^d = S \oplus S^\perp.$$

Theorem 2.18

If A is an $N \times d$ matrix,

$$\text{nullspace} \oplus \text{row space} = \mathbb{R}^d,$$

and the null space and row space are orthogonal to each other.



Corollary 2.7

From Theorem 2.18, the projection matrix onto the null space of A is $P = I - A^+A$.

Theorem 2.19 (Projection is the Nearest Point in the Span)

Let $P\mathbf{b} = AA^+\mathbf{b}$ be the projection of \mathbf{b} onto the column space of A , and let $\mathbf{x}^+ = A^+\mathbf{b}$ be the reduced vector. Then

$$|\mathbf{Ax}^+ - \mathbf{b}|^2 = \min_{\mathbf{x}} |\mathbf{Ax} - \mathbf{b}|^2.$$

Exercise 2.30

Prove Theorems 2.17, 2.18, 2.19 and Corollary 2.7.