Unit 2: Linear Algebra and Multivariable Calculus

Charles Rambo

UCLA Anderson

2024

Table of Contents I

Table of Contents II

Linear Algebra

Vectors

Definition

- A **vector** is quantity that has both direction and magnitude.
- For this class, a **scalar** is simply an element of $\mathbb{R}$.

In introductory texts, a vector is usually written with a bold letter or an arrow over the the letter, e.g. $\boldsymbol{v}$ or $\vec{v}$, and no special notation is used for a scalar. However, beyond introductory texts, typically no special notation is used for a vector either. Whether a quantity is a vector or scalar is implied by the context. We will *mostly* follow the convention of introductory texts here.

Vector Spaces

Definition

A **real vector space** V is a set such that, for $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ in V and α and β in $\mathbb{R}$, the following hold.

VS.1 $(\boldsymbol{u}+\boldsymbol{v})+\boldsymbol{w}=\boldsymbol{u}+(\boldsymbol{v}+\boldsymbol{w})$.

VS.2 There is an element $\mathbf{0}$ in V such that $\mathbf{0}+\boldsymbol{u}=\boldsymbol{u}+\mathbf{0}=\boldsymbol{u}$.

VS.3 There exists $-\boldsymbol{u}$ in V such that $\boldsymbol{u}+(-\boldsymbol{u})=(-\boldsymbol{u})+\boldsymbol{u}=\mathbf{0}$.

VS.4 $\boldsymbol{u}+\boldsymbol{v}=\boldsymbol{v}+\boldsymbol{u}$.

VS.5 $\alpha(\boldsymbol{u}+\boldsymbol{v})=\alpha\boldsymbol{u}+\alpha\boldsymbol{v}$.

VS.6 $(\alpha+\beta)\boldsymbol{u}=\alpha\boldsymbol{u}+\beta\boldsymbol{u}$.

VS.7 $(\alpha\beta)\boldsymbol{u}=\alpha(\beta\boldsymbol{u})$.

VS.8 $1\boldsymbol{u}=\boldsymbol{u}$.

Subspaces

Definition

Let V be a vector space and let W be a subset of V. Then W is a **subspace** of V if the following properties hold.

(i) $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ in W implies $\boldsymbol{w}_1 + \boldsymbol{w}_2$ is in W

(ii) α in $\mathbb{R}$ and $\boldsymbol{w}$ in W implies $\alpha \boldsymbol{w}$ is in W

(iii) The element $\mathbf{0}$ is in W

Vector Spaces and Subspaces

Example

The set $V = \{f : \mathbb{R} \to \mathbb{R} | f \text{ continuous}\}$ is a real vector space and $W = \{f : \mathbb{R} \to \mathbb{R} | f \text{ differentiable}\}$ is a subspace, if for f and g in V, we define $f + g$ to be the function which satisfies

$$(f + g)(x) = f(x) + g(x)$$

and α in $\mathbb{R}$ we define αf to be the function which satisfies

$$(\alpha f)(x) = \alpha f(x).$$

Linear Independence

Definition

Let $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$ to be a set of elements of V. Then the set is **linearly independent** if for $\alpha_1, \alpha_2, \ldots, \alpha_n$ in $\mathbb{R}$,

$$\alpha_1 \boldsymbol{v}_1 + \alpha_2 \boldsymbol{v}_2 + \ldots + \alpha_n \boldsymbol{v}_n = \mathbf{0} \qquad \text{implies} \qquad \alpha_i = 0 \text{ for all } i.$$

If the set is not linearly independent, then it is **linearly dependent**.

Span

Definition

The **span** of $\{v_1, v_2, \ldots, v_m\}$ is

$$\text{span}\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m\} = \{\alpha_1 \boldsymbol{v}_1 + \alpha_2 \boldsymbol{v}_2 + \ldots + \alpha_m \boldsymbol{v}_m : \alpha_i \in \mathbb{R}\}.$$

Basis

Definition

The set $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$ is a **basis** of V if

(i) $\text{span}\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\} = V$ and

(ii) The set $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$ is linearly independent.

Standard Basis

Consider $\mathbb{R}^n$ as a real vector space. If we define

$$\begin{aligned} \boldsymbol{e}_1 &= (1, 0, \ldots, 0, 0) \\ \boldsymbol{e}_2 &= (0, 1, \ldots, 0, 0) \\ &\vdots \\ \boldsymbol{e}_{n-1} &= (0, 0, \ldots, 1, 0) \\ \boldsymbol{e}_n &= (0, 0, \ldots, 0, 1), \end{aligned}$$

then $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_{n-1}, \boldsymbol{e}_n\}$ is a basis for $\mathbb{R}^n$.

Basis Elements and Independence

Theorem

Let V be a vector space, and suppose $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m\}$ is a basis of V. If $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots \boldsymbol{w}_n$ are elements fo V and $n > m$, then $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_n$ are linearly dependent.

Dimension

Theorem

Suppose V is a vector space. If one basis has m elements, and another has n elements, then $m = n$.

This means that the number of elements in a basis is unique.

Definition

The **dimension** of a vector space V is the number of elements in any basis of V.

Matrix Arithmetic

Example

Suppose

$$A = \begin{pmatrix} 1 & 2 & 3 \\ -1 & 0 & 2 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -1 & 5 & -2 \\ 2 & 2 & -1 \end{pmatrix}.$$

Compute (a) $2A + B$ and (b) AB^T.

Python Matrix Arithmetic

Example

Suppose

$$C = \begin{pmatrix} 2 & 1 & 3 & 4 \\ -3 & 1 & 5 & 1 \\ 5 & -1 & 11 & 7 \\ -1 & 10 & 2 & 4 \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} 100 & -5 & 2 & 1 \\ 7 & -2 & 1 & 1 \\ -5 & 1 & 2 & 3 \\ 20 & 1 & 4 & 50 \end{pmatrix}.$$

Use Python to compute (a) $2C + D$ and (b) CD^T.

Python Matrix Arithmetic Solution

```
# Import module
import numpy as np

# Define matrices
C = np.array([[2, 1, 3, 4],
        [-3, 1, 5, 1],
        [5, -1, 11, 7],
        [-1, 10, 2, 4]])

D = np.array([[100, -5, 2, 1],
        [7, -2, 1, 1],
        [-5, 1, 2, 3],
        [20, 1, 4, 50]])

# Perform arithmetic
result_a = 2 * C + D
result_b = C @ D.T
```

Python Matrix Arithmetic Result

The results are

$$\texttt{result\_a} = \begin{pmatrix} 104 & -3 & 8 & 9 \\ 1 & 0 & 11 & 3 \\ 5 & -1 & 24 & 17 \\ 18 & 21 & 8 & 58 \end{pmatrix}$$

and

$$\texttt{result\_b} = \begin{pmatrix} 205 & 19 & 9 & 253 \\ -294 & -17 & 29 & 11 \\ 534 & 55 & 17 & 493 \\ -142 & -21 & 31 & 198 \end{pmatrix}.$$

`np.matmul` and `@`

The operator `@` was introduced in Python 3.5, and is equivalent to `np.matmul`. See `https://numpy.org/doc/stable/reference/generated/numpy.matmul.html` for more details.

Special Matrices

- The $n \times n$ identity matrix I, such that $AI = A$ for A an $m \times n$ matrix and $IB = B$ for B an $n \times m$ matrix. This matrix has ones on the main diagonal and zeros elsewhere. In Python, the command for the $n \times n$ identity matrix is `np.eye(n)`.
- If A is an $n \times n$ **invertible** or **non-singular** matrix, there is an $n \times n$ matrix B such that $AB = BA = I$. We typically call B the **inverse** of A and write it as A^{-1}. In Python, the command is `np.linalg.inv(A)`.

Useful Inverse Matrix Formula

Suppose

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

Then

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

whenever the formula makes sense.

Linear Transformations

Definition

Let U and V be real vector spaces, and suppose α and β are in $\mathbb{R}$. A **linear transformation** $T : U \to V$ satisfies

$$T(\alpha \boldsymbol{u}_1 + \beta \boldsymbol{u}_2) = \alpha T(\boldsymbol{u}_1) + \beta T(\boldsymbol{u}_2)$$

for all $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ in U. The vector space U is the **domain** of T and V is the **codomain** of T. The set $\text{Im}(T) = \{T(\boldsymbol{u}) : \boldsymbol{u} \in U\}$ is the **image** or **range** of T.

Example

Example

Prove $F : \mathbb{R}^3 \to \mathbb{R}^2$ defined by $F : (x, y, z) \mapsto (x, y)$ is a linear transformation.

Kernel

Definition

The **kernel** of a linear transformation $T : U \to V$ is

$$\text{Ker}(T) = \{\boldsymbol{u} : T(\boldsymbol{u}) = \boldsymbol{0}\}.$$

Kernel Example

Example

Define $T : \mathbb{R}^3 \to \mathbb{R}^2$ via

$$T\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 1 \\ 5 & 1 & -5 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

Find $\mathrm{Ker}(T)$.

Solution. Typically this is done by row reducing A. You can also use the function `scipy.linalg.null_space`. In either case, the kernel is

$$\mathrm{Ker}(A) = \mathrm{span}\left\{\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}\right\}.$$

Kernel Theorems

Theorem

Suppose $T : U \to V$. The set $Ker(T)$ is a subspace of U.

Theorem (Rank-Nullity Theorem)

Let U be a vector space. Let $T : U \to V$ be a linear transformation of U into another vector space V. Then

$$dim(U) = \overbrace{dim\ Im(T)}^{rank} + \underbrace{dim\ Ker(T)}_{nullity}$$

Coordinate Representation of a Vector

For V a vectors space with basis $\mathcal{B} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n)$. We can represent an arbitrary $\boldsymbol{w}$ in V using the unique linear combination of the elements of $\mathcal{B}$. Specifically,

$$\boldsymbol{w} = \alpha_1 \boldsymbol{v}_1 + \alpha_2 \boldsymbol{v}_2 + \ldots \alpha_n \boldsymbol{v}_n.$$

Using this, we can write the coordinate representation of $\boldsymbol{w}$ with respect to the basis $\mathcal{B}$:

$$(\boldsymbol{w})_{\mathcal{B}} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}.$$

Linear Transformations and Matrices

There is a matrix representation of any linear transformations between finite dimensional vector spaces. Consider vector space U with ordered basis $\mathcal{B} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_m)$, and vector space V with ordered basis $\mathcal{C} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n)$. Suppose $T : U \to V$ such that

$$T(u_j) = \alpha_{1j}\boldsymbol{v}_1 + \alpha_{2j}\boldsymbol{v}_2 + \ldots + \alpha_{nj}\boldsymbol{v}_n.$$

Then the matrix representation in terms of these two bases is

$$\mathcal{M}(T)_{\mathcal{B}}^{\mathcal{C}} = \begin{pmatrix} \alpha_{11} & \ldots & \alpha_{1m} \\ \vdots & \ddots & \vdots \\ \alpha_{n1} & \ldots & \alpha_{nm} \end{pmatrix}.$$

Linear Transformations and Matrices Example

Example

Consider $F : \mathbb{R}^3 \to \mathbb{R}^2$ defined by $F(x, y, z) = (x, y)$. Write the matrix representation of F in terms of the standard bases.

Change of Basis

Suppose we have bases $\mathcal{B} = (\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n)$ and $\mathcal{C} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_n)$ for a vector space V. Let $T : V \to V$. Then

$$\mathcal{M}(T)_{\mathcal{C}}^{\mathcal{C}} = N^{-1}\mathcal{M}(T)_{\mathcal{B}}^{\mathcal{B}}N,$$

where N is the $n \times n$ matrix whose columns are the basis elements of $\mathcal{C}$ written in terms of the basis $\mathcal{B}$. That is,

$$N = \Big((\boldsymbol{w}_1)_{\mathcal{B}} \ (\boldsymbol{w}_2)_{\mathcal{B}} \ \ldots (\boldsymbol{w}_n)_{\mathcal{B}} \Big).$$

Change of Basis Example

Example

Consider

$$\mathcal{M}(T) = \begin{pmatrix} 2 & -4 \\ 6 & 2 \end{pmatrix}.$$

Write the matrix representation of T in terms of the basis

$$\mathcal{B} = \left(\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right).$$

Inner Product

Definition

Let V be a real vector space, and suppose $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{w}$ are arbitrary elements of V. An **inner product** on V is a function $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ with the following properties.

IP.1 The inner product $\langle \boldsymbol{v}, \boldsymbol{v} \rangle \geq 0$ and $\langle \boldsymbol{v}, \boldsymbol{v} \rangle = 0$ if and only if $\boldsymbol{v} = \mathbf{0}$.

IP.2 $\langle \boldsymbol{v}, \boldsymbol{w} \rangle = \langle \boldsymbol{w}, \boldsymbol{v} \rangle$.

IP.3 For all α and β in $\mathbb{R}$, $\langle \alpha \boldsymbol{u} + \beta \boldsymbol{v}, \boldsymbol{w} \rangle = \alpha \langle \boldsymbol{u}, \boldsymbol{w} \rangle + \beta \langle \boldsymbol{v}, \boldsymbol{w} \rangle$.

The Dot Product

The most common example is the “dot product” in $\mathbb{R}^n$. Suppose

$$\boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \quad \text{and} \quad \boldsymbol{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}.$$

Then this inner product is

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \boldsymbol{u} \bullet \boldsymbol{v} = \boldsymbol{u}^T \boldsymbol{v} = u_1 v_1 + u_2 v_2 + \ldots + u_n v_n.$$

In Python, the command is `np.dot`, though you can also do `u.T @ v` provided that the dimensions are properly defined.

Another Inner Product Example

Consider the vector space of continuous functions on the interval $[0, 1]$.

Then an inner product is

$$\langle f, g \rangle = \int_0^1 f(x)g(x)\ dx.$$

Norm

Definition

Suppose V is a a real vector space, $\boldsymbol{u}$ and $\boldsymbol{v}$ are in V, and α is in $\mathbb{R}$. A **norm** is a real valued function $\|\cdot\| : V \to R$ with the following properties.

N.1 $\|\boldsymbol{v}\| \geq 0$ and $\|\boldsymbol{v}\| = 0$ if and only if $\boldsymbol{v} = \mathbf{0}$.

N.2 $\|\alpha \boldsymbol{v}\| = |\alpha| \|\boldsymbol{v}\|$.

N.3 $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$.

Norm Examples

For the real vector space $\mathbb{R}^n$, the Euclidean norm is most common. For $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, it is defined to be

$$\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x} \bullet \boldsymbol{x}} = \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2}.$$

In Python, the command is `np.linalg.norm`.

Other Norm Examples

- For the real vector space $\mathbb{R}^n$, one example is the ℓ_p-norm where $p \geq 1$. It is defined as

$$\|\boldsymbol{x}\|_p = \left(|x_1|^p + |x_2|^p + \ldots + |x_n|^p\right)^{1/p}.$$

- The limiting case of the ℓ_p-norm is the ℓ_∞-norm. It is defined as

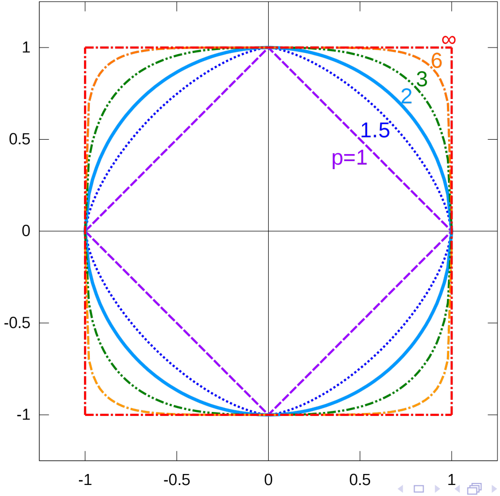
$$\|\boldsymbol{x}\|_\infty = \max_i |x_i|.$$

- For the vector space of continuous real-valued functions on $[0, 1]$, we can define the L^p norm of f to be

$$\|f\|_p = \left(\int_0^1 |f(x)|^p \, dx\right)^{1/p}.$$

ℓ_p-Norms

Unit circle in $\mathbb{R}^2$ for various ℓ_p-norms.



Inner Products Induce Norms

If V is an inner product space, the norm induced by the inner product is

$$\|\boldsymbol{v}\| = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle}.$$

Law of Cosines and Inner Products

People like to think of inner products defining the angle θ between vectors

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \|\boldsymbol{u}\| \|\boldsymbol{v}\| \cos \theta$$

In particular, we say $\boldsymbol{u}$ and $\boldsymbol{v}$ are **orthogonal** if

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0.$$

Distance

Definition

A bivariate function d on a set V is a **distance metric** if for all $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ in V the following hold.

D.1 $d(\boldsymbol{u}, \boldsymbol{v}) \geq 0$ and $d(\boldsymbol{u}, \boldsymbol{v}) = 0$ if and only if $\boldsymbol{u} = \boldsymbol{v}$

D.2 $d(\boldsymbol{u}, \boldsymbol{v}) = d(\boldsymbol{v}, \boldsymbol{u})$

D.3 $d(\boldsymbol{u}, \boldsymbol{v}) + d(\boldsymbol{v}, \boldsymbol{w}) \geq d(\boldsymbol{u}, \boldsymbol{w})$

Distance Metric Examples

- For the real vector space $\mathbb{R}^n$, the Euclidean distance is most common. Given $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$, it is defined to be

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2}.$$

- The discrete metric on any set:

$$d(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} 1 & \boldsymbol{x} \neq \boldsymbol{y} \\ 0 & \boldsymbol{x} = \boldsymbol{y}. \end{cases}$$

- On the set of continuous real-valued functions on the interval [0, 1], we can define

$$d(f, g) = \int_0^1 |f(x) - g(x)| \; dx.$$

Norms Induce Distances

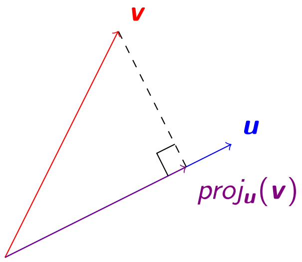
The distance metric induced by a norm is

$$d(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|.$$

Projection

The projection of $\boldsymbol{v}$ onto $\boldsymbol{u}$ is

$$proj_{\boldsymbol{u}}(\boldsymbol{v}) = \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\|^2}\boldsymbol{u}.$$



Projection Example

Example

Suppose

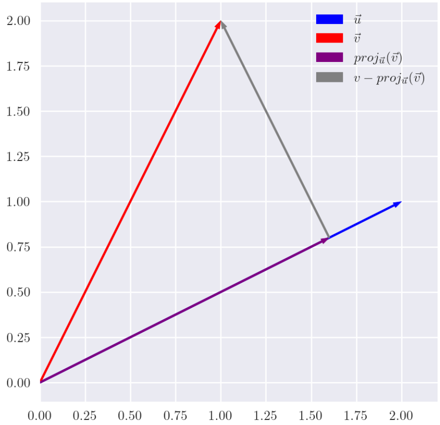
$$\boldsymbol{u} = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{v} = \begin{pmatrix} 1 \\ 3 \end{pmatrix}.$$

Compute $proj_{\boldsymbol{u}}(\boldsymbol{v})$.

Projection Solution Python Code

```python
# Import modules
import numpy as np, matplotlib.pyplot as
    plt

# Use Seaborn style
plt.style.use('seaborn')

# Use latex
plt.rcParams['text.usetex'] = True

# Define u and v
u, v = np.array([2, 1]), np.array([1, 2])

# Calculate project; key step
proj = np.dot(u, v)/np.linalg.norm(u)**2 *
    u

# Calc the part of v perpendicular to u
proj_perp = v - proj

# Define origin
origin = np.array([0, 0])

# Plot figure
fig, ax = plt.subplots(1, 1, dpi = 200)

# Draw arrow for u
ax.arrow(*origin, *u, label = r'$\vec{u}$',
         color = 'blue', width = 0.01,
         length_includes_head = True)

# Draw arrow for v
ax.arrow(*origin, *v, label = r'$\vec{v}$',
         color = 'red', width = 0.01,
         length_includes_head = True)

# Draw arrow for projection
ax.arrow(*origin, *proj, label = r'$proj_{\
    vec{u}}(\vec{v})$',
         color = 'purple', width = 0.01,
         length_includes_head = True)

# Draw arrow for v - proj; initial side at
    terminal side of proj
ax.arrow(*proj, *proj_perp, label = r'$v -
    proj_{\vec{u}}(\vec{v})$',
         color = 'gray', width = 0.01,
         length_includes_head = True)

# Add a little horizontal space for legend
ax.set_xlim([0, np.max([u[1], v[1], proj
    [1], proj_perp[1]]) + 0.2])

# Make aspect ratio equal
fig.gca().set_aspect('equal')

# Place legend at upper right
ax.legend(loc = 'upper right')

# Save the figure
plt.savefig(path + r'ex2-1.png')

# Show graph
plt.show()
```

Projection Result



Gram-Schmidt Orthogonalization Process

Theorem

If $(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n)$ *is a sequence of linearly independent vectors in an inner product space* V, *then the sequence* $(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n)$, *defined by*

$$\boldsymbol{u}_1 = \boldsymbol{v}_1 \qquad \text{and} \qquad \boldsymbol{u}_k = \boldsymbol{v}_k - \sum_{i=1}^{k-1} \frac{\langle \boldsymbol{v}_k, \boldsymbol{u}_i \rangle}{\|\boldsymbol{u}_i\|^2} \boldsymbol{u}_i$$

for $k = 2, 3, \ldots, n$, *is an orthogonal sequence in* V *with the property that*

$$\mathit{span}\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n\} = \mathit{span}\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}.$$

Gram-Schmidt Orthogonalization Process

Example

Orthogonalize the first two vectors of the basis $(1, x, x^2, \ldots)$ for the set of polynomials over $\mathbb{R}$ with inner product

$$\langle p, q \rangle = \int_0^1 p(x)q(x)\ dx.$$

Orthogonal Basis

If $(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n)$ is an orthogonal basis of V, then for $\boldsymbol{v}$ in V we have

$$\boldsymbol{v} = \alpha_1 \boldsymbol{u}_1 + \alpha_2 \boldsymbol{u}_2 + \ldots + \alpha_n \boldsymbol{u}_n$$

implies

$$\alpha_i = \frac{\langle \boldsymbol{u}_i, \boldsymbol{v} \rangle}{\|\boldsymbol{u}_i\|^2}.$$

Cauchy-Schwarz Inequality

Theorem (Cauchy-Schwarz Inequality)

Suppose $\boldsymbol{u}$ *and* $\boldsymbol{v}$ *are in the inner product space* V*. Then*

$$|\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \leq \|\boldsymbol{u}\| \|\boldsymbol{v}\|.$$

The Projection Theorem

Definition

The **orthogonal complement** of a set $X \subseteq V$ is the set

$$X^{\perp} = \{\boldsymbol{v} \in V : \langle \boldsymbol{x}, \boldsymbol{v} \rangle = 0 \text{ for all } \boldsymbol{x} \in X\}.$$

Theorem (Projection Theorem)

If U is a finite-dimensional subspace of an inner product space V, then for each element $\boldsymbol{v}$ in V, there exists unique elements $\boldsymbol{u}$ in U and $\boldsymbol{w}$ in $U^{\perp}$ such that $\boldsymbol{v} = \boldsymbol{u} + \boldsymbol{w}$.

Best Approximation

The Projection Theorem tells us that for $\boldsymbol{u}'$ in U, we will alway have

$$\|\boldsymbol{v} - \boldsymbol{u}\| \leq \|\boldsymbol{v} - \boldsymbol{u}'\|,$$

where

$$proj_U(\boldsymbol{v}) = \boldsymbol{u}.$$

Easy Projection Example

Example

Consider the subspace U of $\mathbb{R}^3$ spanned by the orthogonal vectors

$$\boldsymbol{u}_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{u}_2 = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}.$$

Compute the best approximation of $\boldsymbol{v} = (1, 2, 3)^T$ contained in U.

Hard Projection Example

Example

Consider the inner product space of continuous functions on $[0, 1]$, where the inner product is

$$\langle p, q \rangle = \int_0^1 p(x)q(x)\ dx.$$

Project e^x onto the subspace spanned by the orthogonal vectors 1 and $x - 1/2$. Compare the projection with the tangent line approximation at $x = 1/2$.

Hard Projection Example Solution

Solution. The projection is

$$proj(x) = a_{proj} + b_{proj}\left(x - \frac{1}{2}\right),$$

where

$$a_{proj} = \frac{\langle 1, e^x \rangle}{\|1\|^2} = \frac{\int_0^1 e^x \; dx}{\int_0^1 1^2 \; dx} = e - 1$$

$$b_{proj} = \frac{\langle x - 1/2, e^x \rangle}{\|x - 1/2\|^2} = \frac{\int_0^1 (x - 1/2)e^x \; dx}{\int_0^1 (x - 1/2)^2 \; dx} = 6(3 - e).$$

The tangent line approximation is

$$tan\_line(x) = a_{tl} + b_{tl}\left(x - \frac{1}{2}\right),$$

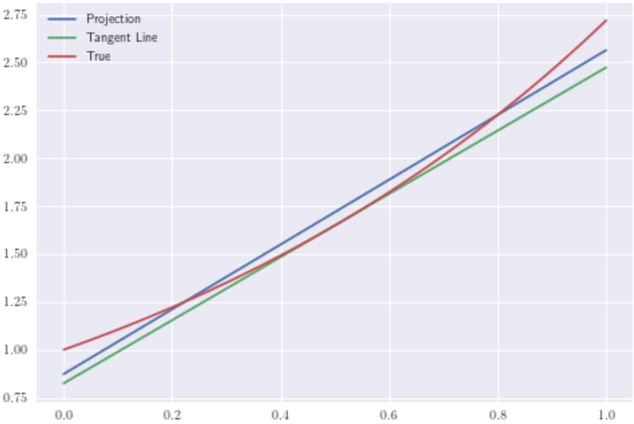
where

$$a_{tl} = e^{1/2} \qquad b_{tl} = e^{1/2}.$$

Hard Projection Example Solution Python Code

```
# Import numerical integrator
from scipy.integrate import quad

# Define inner product
inner = lambda f, g: quad(lambda x: f(x) * g(x), 0, 1)[0]

# Define basis elements
u1, u2 = lambda x: 1, lambda x: x - 1/2

# Calculate inner products
a_proj, b_proj = inner(u1, np.exp)/inner(u1, u1), inner(u2, np.exp)/inner(u2, u2)

# Define small value
h = 1e-5

# Calculate tangent line coeffs
a_tl, b_tl = np.exp(1/2), (np.exp(0.5 + h) - np.exp(0.5 - h))/(2 * h)

# Define functions
proj = lambda x: a_proj * u1(x) + b_proj * u2(x)
tan_line = lambda x: a_tl * u1(x) + b_tl * u2(x)

# Get the x-values for plot
x_vals = np.linspace(0, 1, 100)

# Plot results
plt.plot(x_vals, [proj(x) for x in x_vals], label = 'Projection')
plt.plot(x_vals, [tan_line(x) for x in x_vals], label = 'Tangent Line')
plt.plot(x_vals, [np.exp(x) for x in x_vals], label = 'True')

# Create a legend
plt.legend()
# Save the figure
plt.savefig(path + r'ex2-2.png')
# Show plot
plt.show()
```

Hard Projection Example Image



Projection Example Solution Result

```
# Define norm
norm = lambda f: np.sqrt(inner(f, f))

# Let's calculate the norms
norm_proj, norm_tl = norm(lambda x: np.exp(x) - proj(x)), norm(lambda x: np.exp(x) -
    tan_line(x))

print(f'Using the projection approximation the norm is {norm_proj:.3f}.')
print(f'Using the tangent line approximation the norm is {norm_tl:.3f}.')
```

The output reads:

Using the projection approximation the norm is 0.063.

Using the tangent line approximation the norm is 0.094.

Determinant of a 2×2 Matrix

Suppose

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

Then

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

Determinant Example

Example

$$\begin{vmatrix} -1 & -5 \\ 2 & 1 \end{vmatrix} =$$

Determinant of an $n \times n$ Matrix

Suppose A is the $n \times n$ matrix

$$A = (a_{ij}) .$$

Let A_{ij} denote A with the i-th row and j-th column removed. Then

$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j} a_{ij} \det(A_{ij})$$

for any choice of i in $\{1, 2, \ldots, n\}$.

Determinant Example

Example

$$\begin{vmatrix} 2 & 1 & 2 \\ 0 & 3 & -1 \\ 4 & 1 & 1 \end{vmatrix} =$$

Properties of Determinants

Let A and B be $n \times n$ matrices, $\boldsymbol{a}_j$, $\boldsymbol{b}$, and $\boldsymbol{c}$ be $n \times 1$ vectors, and α and β real numbers.

(a) If the columns of A are linearly dependent, $\det(A) = 0$.

(b) If A^{-1} exists, then $\det(A^{-1}) = \frac{1}{\det(A)}$.

(c) $\det(\alpha A) = \alpha^n \det(A)$.

(d) $\det(A^T) = \det(A)$

(e) $\det(AB) = \det(A)\det(B)$

(f) $\det(I) = 1$

(g) $\left|\boldsymbol{a}_1 \ \boldsymbol{a}_2 \ \ldots \boldsymbol{a}_{j-1} \ \alpha\boldsymbol{b} + \beta\boldsymbol{c} \ \boldsymbol{a}_{j+1} \ldots \ \boldsymbol{a}_n\right| = \alpha \left|\boldsymbol{a}_1 \ \boldsymbol{a}_2 \ \ldots \boldsymbol{a}_{j-1} \ \boldsymbol{b} \ \boldsymbol{a}_{j+1} \ldots \boldsymbol{a}_n\right| + \beta \left|\boldsymbol{a}_1 \ \boldsymbol{a}_2 \ \ldots \boldsymbol{a}_{j-1} \ \boldsymbol{c} \ \boldsymbol{a}_{j+1} \ldots \boldsymbol{a}_n\right|$.

(h) $\left|\boldsymbol{a}_1 \ \boldsymbol{a}_2 \ \ldots \boldsymbol{a}_j \ \boldsymbol{a}_{j+1} \ldots \ \boldsymbol{a}_n\right| = -\left|\boldsymbol{a}_1 \ \boldsymbol{a}_2 \ \ldots \boldsymbol{a}_{j+1} \ \boldsymbol{a}_j \ldots \ \boldsymbol{a}_n\right|$

Property (a) is very important. In `numpy`, there's `np.linalg.det` which computes the determinant. Since computers will be available to you in most circumstances the other properties are less important.

Cramer's Rule

Consider a system of n linear equations with n unknowns

$$A\boldsymbol{x} = \boldsymbol{b}$$

where A is an $n \times n$ matrix with nonzero determinant and

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

Then

$$x_i = \frac{\det(A_i)}{\det(A)},$$

where A_i is the matrix formed by replacing the i-th column of A by the column vector $\boldsymbol{b}$.

Cramer's Rule Example

Example

Solve the system

$$\begin{aligned} 3x + 2y + 4z &= 1 \\ 2x - y + z &= 0 \\ x + 2y + 3z &= 1. \end{aligned}$$

Eigenvectors and Eigenvalues

Definition

Let V be a vector space and consider a linear transformation $T : V \to V$ with matrix representation A. An element $\boldsymbol{v}$ in V is an **eigenvector** of A if there exists a number λ such that $A\boldsymbol{v} = \lambda\boldsymbol{v}$. If $\boldsymbol{v} \neq \mathbf{0}$, then λ is called an **eigenvalue** of A.

In `numpy`, we have `np.linalg.eig`. The function computes the eigenvalues and eigenvectors, respectively.

Eigenvector and Eigenvalues

Example

Let $A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$. Show that $\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ and $\boldsymbol{v}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ are eigenvectors. What are their eigenvalues?

Eigenvectors and Kernels

If $\boldsymbol{v}$ is an eigenvector of A with eigenvalue λ, then

$$A\boldsymbol{v} = \lambda\boldsymbol{v} \qquad \text{implies} \qquad (A - \lambda I)\boldsymbol{v} = \mathbf{0}.$$

So, if $\boldsymbol{v} \neq \mathbf{0}$, then $\text{Ker}(A - \lambda I) \neq \{\mathbf{0}\}$. This, implies $A - \lambda I$ has linearly dependent columns. Hence, $\det(A - \lambda I) = 0$.

Characteristic Polynomial

Definition

For an $n \times n$ matrix A, the **characteristic polynomial** of A is

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

We can find the eigenvalues by finding the zeros of p_A. We can then plug the eigenvalues into $(A - \lambda I)\boldsymbol{x} = \mathbf{0}$ to find the corresponding eigenvectors.

Using the Characteristic Polynomial

Example

Find the eigenvalues and eigenvectors of $A = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$.

Diagonalizable Matrices

Definition

We say that a matrix A is **diagonalizable** if V has a basis of eigenvectors of A.

Diagonalizable Matrices Example

In our previous example, we saw the eigenvectors of

$$A = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix},$$

are

$$\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{v}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

These eigenvectors are linearly independent, so they form a basis for $\mathbb{R}^2$. As a result, A is diagonalizable. In particular, we can use the basis $(\boldsymbol{v}_1, \boldsymbol{v}_2)$ for $\mathbb{R}^2$ and the change of basis formula to diagonalize A. Under the basis, $(\boldsymbol{v}_1, \boldsymbol{v}_2)$ the matrix representation of the linear transformation corresponding to A is

$$N^{-1}AN = \begin{pmatrix} 0.5 & 0 \\ 0 & 1.5 \end{pmatrix},$$

where N is the matrix which contains the basis elements $(\boldsymbol{v}_1, \boldsymbol{v}_2)$ as columns, i.e.

$$N = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$$

Recall: $\boldsymbol{v}_1$ has eigenvalue 0.5 and $\boldsymbol{v}_2$ has eigenvalue 1.5.

Eigenvectors and Eigenvalues Example

Example

Suppose $A = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix}$. Find a basis of eigenvectors as well as their eigenvalues.

Eigenvectors and Eigenvalues Python Example

Example

Suppose $A = \begin{pmatrix} 4 & 0 & 1 \\ -2 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix}$. Find a basis of eigenvectors as well as their eigenvalues.

Eigenvectors and Eigenvalues Python Code and Result

```python
import numpy as np

# Define matrix
A = np.array([[4, 0, 1], [-2, 1, 0], [-2, 0, 1]])

# Get the eigenvalues and eigenvectors
evals, evecs = np.linalg.eig(A)

# Loop through results
for i in range(len(evals)):

    print(f'eigenvalue: {evals[i]:.2f}; eigenvector: {evecs[:, i]}\n')
```

The output is shown below

```
eigenvalue: 1.00; eigenvector: [0. 1. 0.]

eigenvalue: 3.00; eigenvector: [ 0.57735027 -0.57735027 -0.57735027]

eigenvalue: 2.00; eigenvector: [-0.33333333  0.66666667  0.66666667]
```

Symmetric Matrices

Definition

Suppose V is an inner product space, and $T : V \to V$. Then T is **symmetric** if we have the relation

$$\langle T\boldsymbol{v}, \boldsymbol{w} \rangle = \langle \boldsymbol{v}, T\boldsymbol{w} \rangle$$

for all $\boldsymbol{v}$ and $\boldsymbol{w}$ in V.

For the dot product on $\mathbb{R}^n$, if T has matrix representation A then symmetric means $A = A^T$.

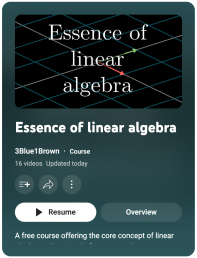
Spectral Theorem

Theorem (Spectral Theorem)

Let V be a finite dimensional non-trivial inner product space over the real numbers, and suppose $T : V \to V$ is a symmetric linear transformation with matrix representation A. Then V has an orthogonal basis consisting of eigenvectors of A.

Linear Algebra on YouTube

Watch the 3Blue1Brown linear algebra video series. The series includes most of what has been covered here as well as other great material (https://www.youtube.com/playlist?list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE_ab).



Multivariable Calculus

Partial Derivatives

Definition

Suppose we have $f : \mathbb{R}^n \to \mathbb{R}$. Then the **partial derivative** of f with respect to x_i is

$$\frac{\partial f}{\partial x_i} = \lim_{h \to 0} \frac{f(x_1, x_2, \ldots, x_i + h, \ldots, x_n) - f(x_1, x_2, \ldots, x_n)}{h}.$$

The most common case is $f : \mathbb{R}^2 \to \mathbb{R}$ such that $z = f(x, y)$. Then the two partials are

$$f_x(x, y) = \frac{\partial z}{\partial x} = \lim_{h \to 0} \frac{f(x + h, y) - f(x, y)}{h}$$

and

$$f_y(x, y) = \frac{\partial z}{\partial y} = \lim_{h \to 0} \frac{f(x, y + h) - f(x, y)}{h}.$$

Clairaut's Theorem

Theorem (Clairaut)

Suppose f is defined on a disk D that contains the point (a, b). If the functions f_{xy} and f_{yx} are both continuous on D, then

$$f_{xy}(a, b) = f_{yx}(a, b).$$

Difference Approximation

Theorem

Suppose f_x and f_y exist on a rectangular region R with sides parallel to the axes and containing the points (a, b) and $(a + \Delta x, b + \Delta y)$. Suppose f_x and f_y are continuous at the point (a, b), and let

$$\Delta z = f(a + \Delta x, b + \Delta y) - f(a, b).$$

Then

$$\Delta z = f_x(a, b)\Delta x + f_y(a, b)\Delta y + \varepsilon_1 \Delta x + \varepsilon_2 \Delta y$$

where $\varepsilon_1, \varepsilon_2 \to 0$ as $\Delta x, \Delta y \to 0$.

Example

Example

Suppose $f(x, y, z) = \sqrt{xyz}$. Approximate $f(3.9, 4.2, 3.9)$.

The Chain Rule

Suppose u is a differentiable function of n variables $x_1, x_2, \ldots, x_n$ and each x_j is a function of the m variables $t_1, t_2, \ldots, t_m$ such that the partial derivate $\frac{\partial x_j}{\partial t_i}$ exists each for all i and j. Then u is a function of $t_1, t_2, \ldots, t_m$ and

$$\frac{\partial u}{\partial t_i} = \frac{\partial u}{\partial x_1}\frac{\partial x_1}{\partial t_i} + \frac{\partial u}{\partial x_2}\frac{\partial x_2}{\partial t_i} + \ldots + \frac{\partial u}{\partial x_n}\frac{\partial x_n}{\partial t_i}$$

for any i.

Chain Rule Example

Example

Consider the Black-Scholes differential equation

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} = rV.$$

Rewrite the partial differential equation using the substitution $z = \ln S$.

Implicit Differentiation

Suppose that $F(x, y) = 0$ and $y = f(x)$. Then

$$\frac{\partial F}{\partial x} + \frac{\partial F}{\partial y}\frac{dy}{dx} = 0.$$

So,

$$\frac{dy}{dx} = -\frac{\frac{\partial F}{\partial x}}{\frac{\partial F}{\partial y}} = -\frac{F_x}{F_y}$$

as long as $\frac{\partial F}{\partial x}$ is continuous and $\frac{\partial F}{\partial y}$ is both continuous and nonzero.

Implicit Differentiation Example

Example

Find $\frac{dy}{dx}$ if $x^3 + y^3 = 6xy$.

Gradient Vector

Let's introduce new notation. Suppose $x \in \mathbb{R}^n$. Define

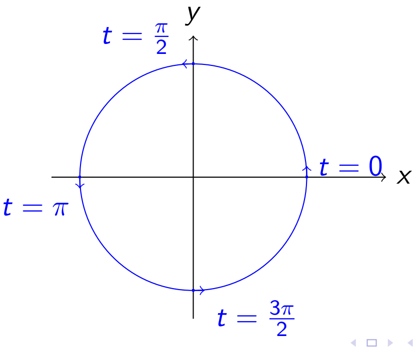
$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right).$$

Space Curve

A curve in $\mathbb{R}^n$ can be defined in terms of a vector valued function $\boldsymbol{r}$, where $\boldsymbol{r} : I \to \mathbb{R}^n$ and I is some subset of $\mathbb{R}$.

For example, consider $\boldsymbol{r} : [0, 2\pi) \to \mathbb{R}^2$ such that $\boldsymbol{r} : t \mapsto (\cos t, \sin t)$.

This defines a circle oriented counterclockwise in $\mathbb{R}^2$.



Derivatives of Space Curves

The derivative of $\boldsymbol{r} : I \to \mathbb{R}^n$ is defined to be

$$\boldsymbol{r}'(t) = \lim_{h \to 0} \frac{\boldsymbol{r}(t+h) - \boldsymbol{r}(t)}{h}.$$

The derivative is tangent to the curve described by $\boldsymbol{r}$.

Gradient Vector Orthogonal to Surface

Suppose that we have a surface defined by $F(\boldsymbol{x}) = k$ where $\boldsymbol{x}$ is in $\mathbb{R}^n$. Consider any curve defined by $\boldsymbol{r} : I \to \mathbb{R}^n$. Then we have the value of the function at $\boldsymbol{r}(t)$ is

$$F(\boldsymbol{r}(t)) = k.$$

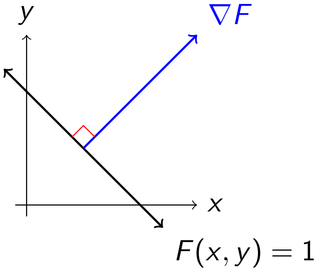
The chain rule implies

$$\nabla F(\boldsymbol{r}(t)) \bullet \boldsymbol{r}'(t) = 0.$$

The vector $\boldsymbol{r}'(t)$ is parallel to the surface. Hence, $\nabla F(\boldsymbol{r}(t))$ is orthogonal. Since $\boldsymbol{r}$ was arbitrary, we must have $\nabla F(x)$ is orthogonal to $F(x) = k$.

Gradient Vector Example

Consider $F(x, y) = x + y = 1$. Then $\nabla F = (1, 1)$. We can see this vector is orthogonal to our line (or "surface" to use the more general term).



Local Maximum and Minimum

Definition

A function of two variables has a **local maximum** at (a, b) if $f(x, y) \leq f(a, b)$ for all points (x, y) in some disk with center (a, b). The number $f(a, b)$ is called a **local maximum value**, if $f(x, y) \geq f(a, b)$ for all (x, y) in such a disk, $f(a, b)$ is a **local minimum value**.

Theorem

If f has a local extremum at (a, b) and the first-order partial derivatives of f exist there, then $f_x(a, b) = f_y(a, b) = 0$.

Second Derivatives Test

Suppose the second partial derivatives of f are continuous in a disk with center (a, b), and suppose that $f_x(a, b) = f_y(a, b) = 0$. Let

$$D = \begin{vmatrix} f_{xx}(a, b) & f_{xy}(a, b) \\ f_{yx}(a, b) & f_{yy}(a, b) \end{vmatrix}.$$

(a) If $D > 0$ and $f_{xx}(a, b) > 0$, then $f(a, b)$ is a local minimum.

(b) If $D > 0$ and $f_{xx}(a, b) < 0$, then $f(a, b)$ is a local maximum.

(c) If $D < 0$, then $f(a, b)$ is not a local extremum.

(d) If $D = 0$, then the test fails.

Second Derivatives Test Example

Example

Find the shortest distance from the point $(1, 0, -2)$ to the plane

$x + 2y + z = 4.$

Minimization Python Example

Example

Use Python to verify the previous example.

```
import numpy as np
from scipy.optimize import minimize

# Define function
def distance(pt):

    # Get the x- and y-values
    x, y = pt[0], pt[1]

    # Define z
    z = 4 - x - 2 * y

    return np.sqrt((x - 1)**2 + y**2 + (z + 2)**2)

# Get the result
minimize(distance, x0 = [0, 0])
```

Another option is to use the constraint $x + 2y + z - 4 = 0$ and optimize with three variables .

Minimization Python Result

Note that

$$\frac{11}{6} \approx 1.83, \qquad \frac{5}{3} \approx 1.67 \qquad \text{and} \qquad \frac{5\sqrt{6}}{6} \approx 2.04$$

```
      fun: 2.0412414523198583
 hess_inv: array([[ 1.71207714, -0.67961111],
       [-0.67961111,  0.68098713]])
      jac: array([9.23871994e-07, 1.51991844e-06])
  message: 'Optimization terminated successfully.'
     nfev: 32
      nit: 7
     njev: 8
   status: 0
  success: True
        x: array([1.83333386, 1.66666707])
```

Extreme Value Theorem for Functions of Two Variables

Theorem

If f is continuous on a closed and bounded set D in $\mathbb{R}^2$, then f attains an absolute maximum value $f(x_1, y_1)$ and an absolute minimum value $f(x_2, y_2)$ at some points (x_1, y_1) and (x_2, y_2) in D.

Global Optimization

Example

Find the absolute maximum and minimum values of $f(x, y) = 5 - 3x + 4y$ on the triangular region with vertices $(0, 0)$, $(4, 0)$, $(4, 5)$.

Global Optimization in Python

There are many global optimization techniques in Python. However, they tend to be slow and not particularly effective on functions that aren't convex or concave. See the SciPy documentation for more details.

Global optimization

`basinhopping`(func, x0[, niter, T, stepsize, ...])	Find the global minimum of a function using the basin-hopping algorithm.
`brute`(func, ranges[, args, Ns, full_output, ...])	Minimize a function over a given range by brute force.
`differential_evolution`(func, bounds[, args, ...])	Finds the global minimum of a multivariate function.
`shgo`(func, bounds[, args, constraints, n, ...])	Finds the global minimum of a function using SHG optimization.
`dual_annealing`(func, bounds[, args, ...])	Find the global minimum of a function using Dual Annealing.
`direct`(func, bounds, *[, args, eps, maxfun, ...])	Finds the global minimum of a function using the DIRECT algorithm.

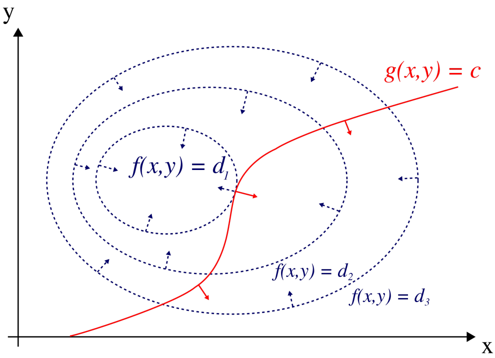
Lagrange Multipliers

For $\boldsymbol{x} \in \mathbb{R}^n$, consider $f(\boldsymbol{x})$ subject to $g(\boldsymbol{x}) = k$. If extrema exist, they satisfy the equation

$$\nabla f(\boldsymbol{x}) = \lambda \nabla g(\boldsymbol{x}).$$

for some λ in $\mathbb{R}$.

Lagrange Multipliers Picture



Lagrange Multipliers Example

Example

Find the shortest distance from the point $(1, 0, -2)$ to the plane

$x + 2y + z = 4$.

Construction of Double Integral

Consider $f : \mathbb{R}^2 \to \mathbb{R}$ and define the closed region

$$R = [a, b] \times [c, d] = \{(x, y) \in \mathbb{R}^2 : a \leq x \leq b,\ c \leq y \leq d\}.$$

Consider partition P of R into subrectangles $R_{ij} = [x_{i-1}, x_i] \times [y_{i-1}, y_i]$, where

$$a = x_0 \leq x_1 < \ldots < x_m = b$$

$$c = y_0 \leq y_1 < \ldots < y_n = d$$

Select (s_{ij}, t_{ij}) from R_{ij}. The area of R_{ij} is $\Delta A_{ij} = \Delta x_i \Delta y_j$. Define the mesh $\|P\| = \max_{i,j}\{\Delta A_{ij}\}$. Then

$$\iint_R f(x, y)\, dA = \lim_{\|P\| \to 0} \sum_{i=1}^{m} \sum_{j=1}^{n} f(s_{ij}, t_{ij})\, \Delta A_{ij}$$

whenever the limit exists.

Double Integral Example

Example

Calculate $\iint_{[0,1]\times[0,1]} xy^2 \, dA$. Use a uniform partition with n^2 subrectangles R_{ij}, and the upper right point of R_{ij} for (s_{ij}, t_{ij}).

Double Integral Python Example

Example

Use `scipy.integrate.dblquad` to verify the previous result for

$$\iint_{[0,1]\times[0,1]} xy^2 \; dA.$$

```python
import numpy as np
from scipy.integrate import dblquad

# Define f
f = lambda y, x: x * y**2

# Integrate
dblquad(f, 0, 1, 0, 1)[0]
```

The output is `0.16666666666666669` which agrees with our previous result.

Fubini's Theorem

Theorem (Fubini)

Consider $R = [a,b] \times [c,d]$ *and define*

$$A_1(x) = \int_c^d f(x,y)\ dy \qquad A_2(y) = \int_a^b f(x,y)\ dx.$$

Then

$$\iint_R f(x,y)\ da = \int_a^b A_1(x)\ dx = \int_c^d A_2(y)\ dy.$$

Fubini's Theorem Example

Example

Evaluate $\displaystyle\iint_R y\sin(xy)\ dA$, where $R = [1,2] \times [0,\pi]$.

Integral over General Region

Theoretically speaking, to calculate

$$\iint_D f(x, y)\ dA$$

where D isn't a rectangle, simply choose rectangle R which contains D and define

$$F(x, y) = \begin{cases} f(x, y), & (x, y) \in D \\ 0, & (x, y) \in R \setminus D. \end{cases}$$

The next slide will help show how to handle these integrals in practice.

Integral of General Region

Example

Compute $\iint_D ye^x \, dA$, where D is the triangular region with vertices $(0,0)$, $(2,4)$, and $(6,0)$.

Jacobian

Definition

The **Jacobian** of the transformation given by $x = g(u, v)$ and $y = h(u, v)$ is

$$J = \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{pmatrix}.$$

Change of Variables

Suppose that we have a one-to-one transformation with continuous partial derivatives that maps the uv-plane to the xy-plane, and in particular the region S to D. Then

$$\iint_D f(x, y)\; dxdy = \iint_S f\left(x(u, v), y(u, v)\right) \left|\det(J)\right| \; dudv.$$

Change of Variables Example

Example

$$\int_{-\infty}^{\infty} e^{-x^2/2}\, dx =$$

Jacobian on YouTube

Watch the Mathemaniac video about the Jacobian on YouTube (`https://youtu.be/wCZ1VEmVjVo`).