Mathematics Refresher

David Barber

Linear Algebra

Matrices

An $m \times n$ matrix $\mathbf{A}$ is a collection of scalar values arranged in a rectangle of m rows and n columns.

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

The i, j element of matrix $\mathbf{A}$ can be written A_{ij} or more conventionally a_{ij}. Where more clarity is required, one may write $[\mathbf{A}]_{ij}$ (for example $\left[\mathbf{A}^{-1}\right]_{ij}$).

Matrix addition

For two matrix $\mathbf{A}$ and $\mathbf{B}$ of the same size,

$$[\mathbf{A} + \mathbf{B}]_{ij} = [\mathbf{A}]_{ij} + [\mathbf{B}]_{ij}$$

Matrix multiplication

For an l by n matrix $\mathbf{A}$ and an n by m matrix B, the product $\mathbf{AB}$ is the l by m matrix with elements

$$[\mathbf{AB}]_{ik} = \sum_{j=1}^{n} [\mathbf{A}]_{ij} [\mathbf{B}]_{jk} \; ; \qquad i = 1, \ldots, l \quad k = 1, \ldots, m \, .$$

In general $\mathbf{BA} \neq \mathbf{AB}$. When $\mathbf{BA} = \mathbf{AB}$ we say they $\mathbf{A}$ and $\mathbf{B}$ commute.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{pmatrix}$$
$$= \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} & a_{11}b_{12} + a_{12}b_{22} + a_{13}b_{32} \\ a_{21}b_{11} + a_{22}b_{21} + a_{23}b_{31} & a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32} \\ a_{31}b_{11} + a_{32}b_{21} + a_{33}b_{31} & a_{31}b_{12} + a_{32}b_{22} + a_{33}b_{32} \end{pmatrix}$$

Identity

The matrix $\mathbf{I}$ is the identity matrix, necessarily square, with 1's on the diagonal and 0's everywhere else. For clarity we may also write $\mathbf{I}_m$ for a square $m \times m$ identity matrix. Then for an $m \times n$ matrix $\mathbf{A}$, $\mathbf{I}_m\mathbf{A} = \mathbf{A}\mathbf{I}_n = \mathbf{A}$. The identity matrix has elements $[\mathbf{I}]_{ij} = \delta_{ij}$ given by the Kronecker delta:

$$\delta_{ij} \equiv \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Transpose

The transpose $\mathbf{B}^\mathsf{T}$ of the n by m matrix $\mathbf{B}$ is the m by n matrix D with components

$$\left[\mathbf{B}^\mathsf{T}\right]_{kj} = \mathbf{B}_{jk}\,; \qquad k = 1, \ldots, m \quad j = 1, \ldots, n\,.$$

$\left(\mathbf{B}^\mathsf{T}\right)^\mathsf{T} = \mathbf{B}$ and $(\mathbf{A}\mathbf{B})^\mathsf{T} = \mathbf{B}^\mathsf{T}\mathbf{A}^\mathsf{T}$. If the shapes of the matrices $\mathbf{A}$,$\mathbf{B}$ and $\mathbf{C}$ are such that it makes sense to calculate the product $\mathbf{ABC}$, then

$$(\mathbf{ABC})^\mathsf{T} = \mathbf{C}^\mathsf{T}\mathbf{B}^\mathsf{T}\mathbf{A}^\mathsf{T}$$

Vector algebra

Vectors

Let $\mathbf{x}$ denote the n-dimensional column vector with components

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

A vector can be considered a $n \times 1$ matrix.

Addition

$$\mathbf{x} + \mathbf{y} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{pmatrix}$$

Vectors

Euclidean representation

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = x_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + x_3 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

We can write this as

$$\mathbf{x} = x_1 \mathbf{e}^1 + x_2 \mathbf{e}^2 + x_3 \mathbf{e}^3$$

Using a different basis

We can choose other basis vectors and then write the same vector

$$\mathbf{x} = y_1 \mathbf{b}^1 + y_2 \mathbf{b}^2 + y_3 \mathbf{b}^3$$

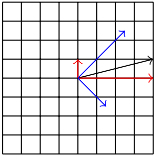
If these basis vectors are orthonormal

$$y_i = \mathbf{x}^{\mathsf{T}} \mathbf{b}^i$$

Vectors: 2D example

Euclidean representation

$$\mathbf{x} = \begin{pmatrix} 4 \\ 1 \end{pmatrix}$$



Using a different basis

We can choose other basis vectors and then write the same vector

$$\mathbf{x} = y_1 \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} + y_2 \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$$

Since these basis vectors are orthonormal

$$y_1 = \begin{pmatrix} 4 \\ 1 \end{pmatrix}^\mathsf{T} \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} = 5/\sqrt{2}, \quad y_2 = \begin{pmatrix} 4 \\ 1 \end{pmatrix}^\mathsf{T} \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} = -3/\sqrt{2}$$

Scalar product

$$\mathbf{w} \cdot \mathbf{x} = \sum_{i=1}^{n} w_i x_i = \mathbf{w}^\mathsf{T} \mathbf{x}$$

The length of a vector is denoted $|\mathbf{x}|$, the squared length is given by

$$|\mathbf{x}|^2 = \mathbf{x}^\mathsf{T} \mathbf{x} = \mathbf{x}^2 = x_1^2 + x_2^2 + \cdots + x_n^2$$

A unit vector $\mathbf{x}$ has $\mathbf{x}^\mathsf{T} \mathbf{x} = 1$. The scalar product has a natural geometric interpretation as:

$$\mathbf{w} \cdot \mathbf{x} = |\mathbf{w}| \, |\mathbf{x}| \cos(\theta)$$

where θ is the angle between the two vectors. Thus if the lengths of two vectors are fixed their inner product is largest when $\theta = 0$, whereupon one vector is a constant multiple of the other. If the scalar product $\mathbf{x}^\mathsf{T} \mathbf{y} = 0$, then $\mathbf{x}$ and $\mathbf{y}$ are orthogonal.

Linear dependence

- A set of vectors $\mathbf{x}^1, \ldots, \mathbf{x}^n$ is linearly dependent if there exists a vector $\mathbf{x}^j$ that can be expressed as a linear combination of the other vectors.
- If the only solution to

$$\sum_{i=1}^{n} \alpha_i \mathbf{x}^i = \mathbf{0}$$

is for all $\alpha_i = 0, i = 1, \ldots, n$, the vectors $\mathbf{x}^1, \ldots, \mathbf{x}^n$ are linearly independent.

Determinant

For a square matrix $\mathbf{A}$, the determinant is the volume of the transformation of the matrix $\mathbf{A}$ (up to a sign change). Writing $[\mathbf{A}]_{ij} = a_{ij}$,

$$\det \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = a_{11}a_{22} - a_{21}a_{12}$$

The determinant in the (3×3) case has the form

$$a_{11}\det \begin{pmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{pmatrix} - a_{12}\det \begin{pmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{pmatrix} + a_{13}\det \begin{pmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix}$$

More generally, the determinant can be computed recursively as an expansion along the top row of determinants of reduced matrices.

$$\det\left(\mathbf{A}^{\mathsf{T}}\right) = \det\left(\mathbf{A}\right)$$

$$\det\left(\mathbf{A}\mathbf{B}\right) = \det\left(\mathbf{A}\right)\det\left(\mathbf{B}\right), \qquad \det\left(\mathbf{I}\right) = 1 \Rightarrow \det\left(\mathbf{A}^{-1}\right) = 1/\det\left(\mathbf{A}\right)$$

Matrix inversion

For a square matrix $\mathbf{A}$, its inverse satisfies

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I} = \mathbf{A}\mathbf{A}^{-1}$$

It is not always possible to find a matrix $\mathbf{A}^{-1}$ such that $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$, in which case $\mathbf{A}$ singular. Geometrically, singular matrices correspond to projections. Provided the inverses exist

$$(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$$

Pseudo inverse

For a non-square matrix $\mathbf{A}$ such that $\mathbf{A}\mathbf{A}^{\mathsf{T}}$ is invertible,

$$\mathbf{A}^{\dagger} = \mathbf{A}^{\mathsf{T}}\left(\mathbf{A}\mathbf{A}^{\mathsf{T}}\right)^{-1}$$

satisfies $\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{I}$.

Solving Linear Systems

Problem

Given square $N \times N$ matrix $\mathbf{A}$ and vector $\mathbf{b}$, find the vector $\mathbf{x}$ that satisfies

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

Solution

Algebraically, we have the inverse:

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

In practice, we solve solve for $\mathbf{x}$ numerically using Gaussian Elimination – more stable numerically.

Complexity

Solving a linear system is $O\left(N^3\right)$ – can be very expensive for large N.
Approximate methods include conjugate gradient and related approaches.

Matrix rank

For an $m \times n$ matrix $\mathbf{X}$ with n columns, each written as an m-vector:

$$\mathbf{X} = \left[\mathbf{x}^1, \ldots, \mathbf{x}^n\right]$$

the rank of $\mathbf{X}$ is the maximum number of linearly independent columns (or equivalently rows).

Full rank

An $n \times n$ square matrix is full rank if the rank is n, in which case the matrix is must be non-singular. Otherwise the matrix is reduced rank and is singular.

Orthogonal matrix

A square matrix $\mathbf{A}$ is orthogonal if

$$\mathbf{A}\mathbf{A}^\mathsf{T} = \mathbf{I} = \mathbf{A}^\mathsf{T}\mathbf{A}$$

From the properties of the determinant, we see therefore that an orthogonal matrix has determinant ± 1 and hence corresponds to a volume preserving transformation.

$$\begin{aligned} \det\left(\mathbf{A}\mathbf{A}^\mathsf{T}\right) &= \det\left(\mathbf{I}\right) \\ \det\left(\mathbf{A}\right)\det\left(\mathbf{A}^\mathsf{T}\right) &= 1 \\ \det\left(\mathbf{A}\right)^2 &= 1 \end{aligned}$$

This means that the transformation that $\mathbf{A}$ represents is something like a rotation, reflection or shear.

Linear transformations

Cartesian coordinate system

Define $\mathbf{u}_i$ to be the vector with zeros everywhere expect for the i^{th} entry, then a vector can be expressed as $\mathbf{x} = \sum_i x_i \mathbf{u}_i$.

Linear transformation

- What does a matrix represent in terms of a transformation?

  $$\mathbf{A}\mathbf{u}_i = \mathbf{a}_i$$

  where $\mathbf{a}_i$ is the i^{th} column of $\mathbf{A}$.
- That is, the columns of the matrix $\mathbf{A}$ represent where the cartesian basis vectors get transformed to.
- More generally, a linear transformation of $\mathbf{x}$ is given by matrix multiplication by some matrix $\mathbf{A}$

  $$\mathbf{A}\mathbf{x} = \sum_i x_i \mathbf{A}\mathbf{u}_i = \sum_i x_i \mathbf{a}_i$$

Eigenvalues and eigenvectors

For an $n \times n$ square matrix $\mathbf{A}$, $\mathbf{e}$ is an eigenvector of $\mathbf{A}$ with eigenvalue λ if

$$\mathbf{A}\mathbf{e} = \lambda\mathbf{e}$$

For an $(n \times n)$ dimensional matrix, there are (including repetitions) n eigenvalues, each with a corresponding eigenvector. We can write

$$\underbrace{(\mathbf{A} - \lambda\mathbf{I})}_{\mathbf{B}}\mathbf{e} = \mathbf{0}$$

If $\mathbf{B}$ has an inverse, then the only solution is $\mathbf{e} = \mathbf{B}^{-1}\mathbf{0} = \mathbf{0}$, which trivially satisfies the eigen-equation. For any non-trivial solution we therefore need $\mathbf{B}$ to be non-invertible. Hence λ is an eigenvalue of $\mathbf{A}$ if

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$$

It may be that for an eigenvalue λ the eigenvector is not unique and there is a space of corresponding vectors.

Spectral decomposition

A real symmetric matrix $N \times N$ $\mathbf{A}$ has an eigen-decomposition

$$\mathbf{A} = \sum_{i=1}^{n} \lambda_i \mathbf{e}_i \mathbf{e}_i^{\mathsf{T}}$$

where λ_i is the eigenvalue of eigenvector $\mathbf{e}_i$ and the eigenvectors form an orthogonal set,

$$\left(\mathbf{e}^i\right)^{\mathsf{T}} \mathbf{e}^j = \delta_{ij} \left(\mathbf{e}^i\right)^{\mathsf{T}} \mathbf{e}^i$$

In matrix notation

$$\mathbf{A} = \mathbf{E} \boldsymbol{\Lambda} \mathbf{E}^{\mathsf{T}}$$

where $\mathbf{E}$ is the orthogonal matrix of eigenvectors and $\boldsymbol{\Lambda}$ the corresponding diagonal eigenvalue matrix.

Computational Complexity

It generally takes $O\left(N^3\right)$ time to compute the eigen-decomposition.

Singular Value Decomposition

The SVD decomposition of a $n \times p$ matrix $\mathbf{X}$ is

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathsf{T}}$$

where $\dim \mathbf{U} = n \times n$ with $\mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{I}_n$. Also $\dim \mathbf{V} = p \times p$ with $\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}_p$.

- The matrix $\mathbf{S}$ has $\dim \mathbf{S} = n \times p$ with zeros everywhere except on the diagonal entries.
- The singular values are the diagonal entries $[\mathbf{S}]_{ii}$ and are positive.
- The singular values are ordered so that the upper left diagonal element of $\mathbf{S}$ contains the largest singular value.

Computational Complexity

It takes $O\left(\max(n,p)\left(\min(n,p)\right)^2\right)$ time to compute the SVD-decomposition.

Positive definite matrix

- A symmetric matrix $\mathbf{A}$ with the property that $\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} \geq 0$ for any vector $\mathbf{x}$ is called positive semidefinite.
- A symmetric matrix $\mathbf{A}$, with the property that $\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} > 0$ for any vector $\mathbf{x} \neq 0$ is called positive definite.
- A positive definite matrix has full rank and is thus invertible.

Eigen-decomposition

Using the eigen-decomposition of $\mathbf{A}$,

$$\mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x} = \sum_i \lambda_i \mathbf{x}^\mathsf{T}\mathbf{e}^i(\mathbf{e}^i)^\mathsf{T}\mathbf{x} = \sum_i \lambda_i \left(\mathbf{x}^\mathsf{T}\mathbf{e}^i\right)^2$$

which is greater than zero if and only if all the eigenvalues are positive. Hence $\mathbf{A}$ is positive definite if and only if all its eigenvalues are positive.

Trace and Det

$$\operatorname{trace}(\mathbf{A}) = \sum_i A_{ii} = \sum_i \lambda_i$$

where λ_i are the eigenvalues of $\mathbf{A}$.

$$\det(\mathbf{A}) = \prod_{i=1}^{n} \lambda_i$$

A matrix is singular if it has a zero eigenvalue.

Trace-Log formula

For a positive definite matrix $\mathbf{A}$,

$$\operatorname{trace}(\log \mathbf{A}) \equiv \log\det(\mathbf{A})$$

The above logarithm of a matrix is not the element-wise logarithm. In general for an analytic function $f(x)$, $f(\mathbf{M})$ is defined via the Taylor series expansion of the function. On the right, since $\det(\mathbf{A})$ is a scalar, the logarithm is the standard logarithm of a scalar.

Calculus

Calculus

For a function $f(x)$, the derivative is defined to be

$$\frac{df}{dx} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{\delta}$$

This is also often written as $f'(x)$ for convenience.
The second derivative is defined to be the derivative of the derivative:

$$\frac{d^2 f}{dx^2} = \lim_{\delta \to 0} \frac{\frac{df}{dx}(x+\delta) - \frac{df}{dx}(x)}{\delta}$$

also written as $f''(x)$ for convenience.
Note that the funny notation is because one can think of the derivative as an operator $\frac{d}{dx}$ that we apply to the function $f(x)$. The second derivative is given by applying this operator twice: $(\frac{d}{dx})^2$ which is more conveniently written as $\frac{d^2}{dx^2}$.

Taylor Series

Any smooth function can be written as

$$\begin{aligned} f(x) &= f(0) + \sum_{i=1}^{\infty} \frac{x^i}{i!} \left(\frac{d}{dx} \right)^i f(x) \Big|_{x=0} \\ &= f(0) + x \frac{df}{dx} + \frac{x^2}{2} \frac{d^2 f}{dx^2} + \dots \end{aligned}$$

Some Calculus Rules

Chain Rule

For a function of a function $f(g(x))$ (*e.g.* $\sin(\cos(x))$)

$$\frac{d(f(g(x))}{dx} = \left.\frac{df(y)}{dy}\right|_{y=f(x)} \frac{dg}{dx}$$

which is usually more conveniently written as

$$\frac{d(f(g(x))}{dx} = \frac{df}{dg}\frac{dg}{dx}$$

Sum Rule

The differential operator is a linear operator and therefore

$$\frac{d}{dx}(f+g) = \frac{df}{dx} + \frac{dg}{dx}$$

Product Rule

$$\frac{d}{dx}(fg) = f\frac{dg}{dx} + g\frac{df}{dx}$$

Numerical Approximation

Take a finite (small value) for δ. Then

$$\frac{df}{dx} \approx \frac{f(x+\delta) - f(x)}{\delta} + O\left(\delta^2\right)$$

Central Difference

Using the Taylor series, we can write

$$f(x+\delta) \approx f(x) + \delta f'(x) + \frac{\delta^2}{2} f''(x) + O\left(\delta^3\right)$$
$$f(x-\delta) \approx f(x) - \delta f'(x) + \frac{\delta^2}{2} f''(x) + O\left(\delta^3\right)$$

Subtracting, we can rearrange to give

$$f'(x) \approx \frac{f(x+\delta) - f(x-\delta)}{2\delta} + O\left(\delta^3\right)$$

At the cost of an additional function evaluation, we therefore have a *much* more accurate approximation.

Partial and Total Derivative

For a function that depends on two (or more) variables $f(x, y)$, the partial derivative with respect to x is defined as

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

That is, the partial derivative with respect to x keeps the state of all other variables fixed.

- Consider a function $f(x)$ that depends directly on x in some manner, and indirectly through another function. We want to find the change in f as we change x, accounting also for indirect changes.
- Consider, for example

  $$f(x) = x^2 + xg, \qquad \text{where } g(x) = x^2$$

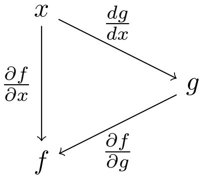
  Then df/dx (the total derivative) is given by

  $$\begin{aligned} \frac{df}{dx} &= \frac{\partial f}{\partial x} + \frac{\partial f}{\partial g}\frac{dg}{dx} \\ &= \underbrace{2x + g}_{\text{partial derivative}} + \underbrace{x}_{\text{p.d wrt } y.} \underbrace{2x}_{\text{t.d of } g} \end{aligned}$$

Partial and Total Derivative (Graphical Representation)

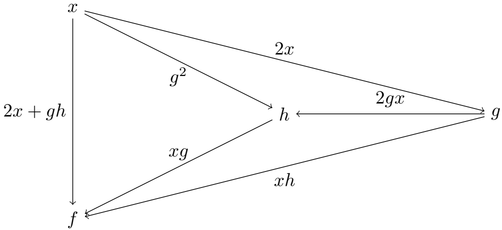
A useful graphical representation is that the total derivative of f with respect to x is given by the sum over all path values from x to f, where each path value is the product of the derivatives of the functions on the edges:

$$\frac{df}{dx} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial g}\frac{dg}{dx}$$



Example

For $f(x) = x^2 + xgh$, where $g = x^2$ and $h = xg^2$



$$f'(x) = (2x + gh) + (g^2 xg) + (2x2gxxg) + (2xxh) = 2x + 8x^7$$

Multivariate Calculus

Partial derivative

Consider a function of n variables, $f(x_1, x_2, \ldots, x_n)$ or $f(\mathbf{x})$. The partial derivative of f *w.r.t.* x_i is defined as the following limit (when it exists)

$$\frac{\partial f}{\partial x_i} = \lim_{h \to 0} \frac{f(x_1, \ldots, x_{i-1}, x_i + h, x_{i+1}, \ldots, x_n) - f(\mathbf{x})}{h}$$

Gradient vector

For function f the gradient is denoted ∇f or $\mathbf{g}$:

$$\nabla f(\mathbf{x}) \equiv \mathbf{g}(\mathbf{x}) \equiv \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

Interpreting the gradient vector

- Consider a function $f(\mathbf{x})$ that depends on a vector $\mathbf{x}$.
- We are interested in how the function changes when the vector $\mathbf{x}$ changes by a small amount : $\mathbf{x} \to \mathbf{x} + \boldsymbol{\delta}$, where $\boldsymbol{\delta}$ is a vector whose length is very small:

$$f\left(\mathbf{x} + \boldsymbol{\delta}\right) = f(\mathbf{x}) + \sum_i \delta_i \frac{\partial f}{\partial x_i} + O\left(\boldsymbol{\delta}^2\right)$$

- We can interpret the summation above as the scalar product between the vector ∇f with components $[\nabla f]_i = \frac{\partial f}{\partial x_i}$ and $\boldsymbol{\delta}$.

$$f\left(\mathbf{x} + \boldsymbol{\delta}\right) = f(\mathbf{x}) + (\nabla f)^{\mathsf{T}} \boldsymbol{\delta} + O\left(\boldsymbol{\delta}^2\right)$$

Interpreting the Gradient

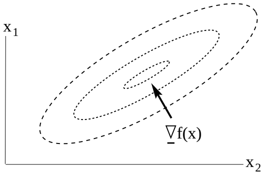


Figure : Interpreting the gradient. The ellipses are contours of constant function value, $f = \text{const}$. At any point $\mathbf{x}$, the gradient vector $\nabla f(\mathbf{x})$ points along the direction of maximal increase of the function.

Consider a direction $\hat{\mathbf{p}}$ (a unit length vector). Then a displacement, δ units along this direction changes the function value to

$$f(\mathbf{x} + \delta\hat{\mathbf{p}}) \approx f(\mathbf{x}) + \delta\nabla f(\mathbf{x}) \cdot \hat{\mathbf{p}}$$

The direction $\hat{\mathbf{p}}$ for which the function has the largest change is that which maximises the overlap

$$\nabla f(\mathbf{x}) \cdot \hat{\mathbf{p}} = |\nabla f(\mathbf{x})||\hat{\mathbf{p}}| \cos\theta = |\nabla f(\mathbf{x})| \cos\theta$$

where θ is the angle between $\hat{\mathbf{p}}$ and $\nabla f(\mathbf{x})$. The overlap is maximised when $\theta = 0$, giving $\hat{\mathbf{p}} = \nabla f(\mathbf{x})/|\nabla f(\mathbf{x})|$. Hence, the direction along which the function changes the most rapidly is along $\nabla f(\mathbf{x})$.

Higher derivatives

The 'second-derivative' of an n-variable function is defined by

$$\frac{\partial}{\partial x_i}\left(\frac{\partial f}{\partial x_j}\right) \qquad i = 1, \ldots, n; \;\; j = 1, \ldots, n$$

which is usually written

$$\frac{\partial^2 f}{\partial x_i \partial x_j}, \;\; i \neq j \qquad \frac{\partial^2 f}{\partial {x_i}^2}, \;\; i = j$$

If the partial derivatives $\partial f/\partial x_i$, $\partial f/\partial x_j$ and $\partial^2 f/\partial x_i \partial x_j$ are continuous, then $\partial^2 f/\partial x_i \partial x_j$ exists and

$$\partial^2 f/\partial x_i \partial x_j = \partial^2 f/\partial x_j \partial x_i \, .$$

This is also denoted by $\nabla\nabla f$. These n^2 second partial derivatives are represented by a square, symmetric matrix called the Hessian matrix of $f(\mathbf{x})$.

$$\mathbf{H}_f(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f}{\partial {x_1}^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n} & \cdots & \frac{\partial^2 f}{\partial {x_n}^2} \end{pmatrix}$$

Vector Taylor Series

For a scalar function of a vector argument, the first terms of the expansion are

$$f(\mathbf{x}+\boldsymbol{\delta}) \approx f(\mathbf{x}) + \boldsymbol{\delta}^{\mathsf{T}}\mathbf{g} + \frac{1}{2}\boldsymbol{\delta}^{\mathsf{T}}\mathbf{H}\boldsymbol{\delta}$$

where $\mathbf{g}$ is the gradient vector of f, evaluated at $\mathbf{x}$ and $\mathbf{H}$ is the Hessian of f, evaluated at $\mathbf{x}$.

- If $\mathbf{H}$ is positive definite, the function looks locally like a bowl $\cup$ around the point $\mathbf{x}$.
- If $\mathbf{H}$ is negative definite, the function looks locally like an upturned bowl $\cap$ around the point $\mathbf{x}$.
- If $\mathbf{H}$ is non-definite (neither positive nor negative), there are directions through $\mathbf{x}$ along which the function looks like $\cup$ and others along which is looks like $\cap$.

Matrix calculus

For matrices $\mathbf{A}$ and $\mathbf{B}$

$$\frac{\partial}{\partial \mathbf{A}} \mathrm{trace}\left(\mathbf{A}\mathbf{B}\right) \equiv \mathbf{B}^{\mathsf{T}}$$

$$\partial \log \det\left(\mathbf{A}\right) = \partial \mathrm{trace}\left(\log \mathbf{A}\right) = \mathrm{trace}\left(\mathbf{A}^{-1} \partial \mathbf{A}\right)$$

So that

$$\frac{\partial}{\partial \mathbf{A}} \log \det\left(\mathbf{A}\right) = \mathbf{A}^{-\mathsf{T}}$$

For an invertible matrix $\mathbf{A}$,

$$\partial \mathbf{A}^{-1} \equiv -\mathbf{A}^{-\mathsf{T}} \partial \mathbf{A} \mathbf{A}^{-1}$$

Convex Analysis

Convex Function

- A function $f(\mathbf{x})$ is convex if, for any two point $\mathbf{x}$ and $\mathbf{y}$ and $0 < \lambda < 1$

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y})$$

- If f is twice differentiable, $f(\mathbf{x})$ is convex if its Hessian $\mathbf{H}(\mathbf{x})$ is positive definite for all points $\mathbf{x}$.

Optimisation

- Geometrically, (strictly *i.e.* the above is $<$ not $\leq$) convex functions look like $\cup$ and have only one minimum.
- Convex functions are very important since there are typically very efficient algorithms that guarantee to find the global minimum of the function.
- A function $f(\mathbf{x})$ is concave if $-f(\mathbf{x})$ is convex.
- In much of machine learning, we need to learn parameters through some form of optimisation. If the objective function is convex, this will make parameter learning straightforward.

Properties of Convex functions

Norms are convex

All norms are convex, in particular the p-norm

$$||x||_p \equiv \left(\sum_i |x_i|^p \right)^{1/p}, \qquad p \geq 1$$

Compositions

If f and g are convex then:

- $f + g$ is convex (positive sums of convex functions are convex)
- $f(Ax + b)$ is convex ('affine transformation')
- $f(g(x))$ is convex provided f is an increasing function

Log convexity

- In machine learning we often encounter 'log convex' functions. This means a function g such that f, where $f(x) = \log g(x)$, is convex.
- For example $g(x) = \exp(x^2)$ is log convex.

Exercises: Show the following functions are convex

$f(x) = x^2$

$f(x) = -\log \sigma(x)$, where $\sigma(x) = 1/(1 + \exp(-x))$

Exercises: Show the following functions are convex

$f(\mathbf{x}) = \mathbf{x}^\mathsf{T}\mathbf{A}\mathbf{x}$ for positive definite $\mathbf{A}$

$f(\mathbf{x}) = -\log \sigma(\mathbf{x}^\mathsf{T}\mathbf{w})$, where $\sigma(x) = 1/(1 + \exp(-x))$

Numerical Issues

Numerical issues: rounding error

- Often in machine learning we have a large number of terms to sum, for example when computing the log likelihood for a large number of datapoints.
- It's good to be aware of potential numerical limitations and ways to improve accuracy, should this be a problem. Double floats have a relative error of around 1×10^{-16}.
- Operations that are mathematical identities may not remain so. For example

  $$\sum_n x_i^n x_j^n$$

  should give rise to a symmetric matrix. However, this symmetry can be lost due to roundoff.
- In general, it's worth checking key points in your code for such issues.

Numerical issues: rounding error

- Consider

$$S = \sum_{i=1}^{N} x_i$$

If x_i cannot be represented exactly by your machine, round-off error will potentially accumulate in computing S.

- Let y be an 'approximation' to each x_i, then

$$S = \sum_{i=1}^{N} (x_i - y + y) = Ny + \sum_{i=1}^{N} (x_i - y)$$

If each x_i is close to y, then the term $\sum_{i=1}^{N} (x_i - y)$ is small but not sensitive to round off error (since each term is small and has roughly the same value).

See `testacc.m` for an example.

`logsumexp`

- It's common in ML to come across expressions such as

$$S = \exp(a) + \exp(b)$$

for large (in absolute value) a or b. If $a = 1000$, Matlab will return ∞ (0 for $a = -1000$). It's not sufficient to simply compute the log:

$$\log S = \log\left(\exp(a) + \exp(b)\right)$$

since this requires the exponentiation still of each term.

- Let $m = \max(a, b)$.

$$\log S = m + \log\left(\exp(a-m) + \exp(b-m)\right)$$

Let's say that $m = a$, then

$$\log S = a + \log\left(1 + \exp(b-a)\right)$$

Since $a > b$ then $\exp(b-a) < 1$ and $\log\left(1 + \exp(b-a)\right) < \log 2$. We can compute $\log S$ more accurately this way.

- More generally, we define the `logsumexp` function

$$\texttt{logsumexp}(\mathbf{x}) = m + \log\left(\sum_{i=1}^{N} \exp(x_i - m)\right), \quad m = \max(x_1, \ldots, x_N)$$

`logsumexp`: example

In a classification problem of a 100 dimensional vector $\mathbf{x}$,

$$p(c = i|\mathbf{x}) = \frac{e^{-(\mathbf{x}-\mathbf{m}_i)^2}}{\sum_j e^{-(\mathbf{x}-\mathbf{m}_j)^2}}$$

A naive implementation of this is likely to lead to $\frac{0}{0}$ and a numerical error.

Using `logsumexp`

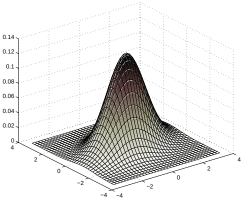
$$\log p(c = i|\mathbf{x}) = y_i - \texttt{logsumexp}(\mathbf{y})$$

where

$$y_j = -\left(\mathbf{x} - \mathbf{m}_j\right)^2$$

Distributions

Multivariate Gaussian



$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \equiv \frac{1}{\sqrt{\det\left(2\pi\boldsymbol{\Sigma}\right)}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

- $\boldsymbol{\mu}$ is the mean vector of the distribution:

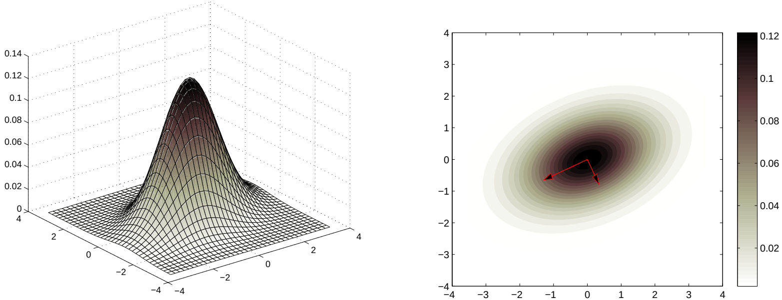
$$\boldsymbol{\mu} = \left\langle \mathbf{x} \right\rangle_{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})}$$

- $\boldsymbol{\Sigma}$ is the covariance matrix of the distribution.

$$\boldsymbol{\Sigma} = \left\langle \left(\mathbf{x} - \boldsymbol{\mu}\right)\left(\mathbf{x} - \boldsymbol{\mu}\right)^{\mathsf{T}} \right\rangle_{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})}$$

- $\int_{-\infty}^{\infty} p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) d\mathbf{x} = 1.$

Geometric Picture



(a) (b)

Figure : **(a)**: Bivariate Gaussian with mean $(0, 0)$ and covariance $[1, 0.5; 0.5, 1.75]$. Plotted on the vertical axis is the probability density value $p(x)$. **(b)**: Probability density contours for the same bivariate Gaussian. Plotted are the unit eigenvectors scaled by the square root of their eigenvalues, $\sqrt{\lambda_i}$.

Geometric Picture

Every real symmetric matrix $D \times D$ has an eigen-decomposition

$$\mathbf{\Sigma} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^{\mathsf{T}}$$

where $\mathbf{E}^{\mathsf{T}}\mathbf{E} = \mathbf{I}$ and $\mathbf{\Lambda} = \mathrm{diag}\left(\lambda_1, \ldots, \lambda_D\right)$. In the case of a covariance matrix, all the eigenvalues λ_i are positive. This means that one can use the transformation

$$\mathbf{y} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{E}^{\mathsf{T}}\left(\mathbf{x} - \boldsymbol{\mu}\right)$$

so that

$$\left(\mathbf{x} - \boldsymbol{\mu}\right)^{\mathsf{T}}\mathbf{\Sigma}^{-1}\left(\mathbf{x} - \boldsymbol{\mu}\right) = \left(\mathbf{x} - \boldsymbol{\mu}\right)^{\mathsf{T}}\mathbf{E}\mathbf{\Lambda}^{-1}\mathbf{E}^{\mathsf{T}}\left(\mathbf{x} - \boldsymbol{\mu}\right) = \mathbf{y}^{\mathsf{T}}\mathbf{y}$$

Under this transformation, the multivariate Gaussian reduces to a product of D univariate zero-mean unit variance Gaussians. This means that we can view a multivariate Gaussian as a shifted, scaled and rotated version of a 'standard' (zero mean, unit covariance) Gaussian in which the centre is given by the mean, the rotation by the eigenvectors, and the scaling by the square root of the eigenvalues.

Linear Transform of a Gaussian

- Let $\mathbf{y}$ be linearly related to $\mathbf{x}$ through

$$\mathbf{y} = \mathbf{M}\mathbf{x} + \boldsymbol{\eta}$$

where $\boldsymbol{\eta} \sim \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$, and $\mathbf{x} \sim \mathcal{N}\left(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x\right)$.

- Then the marginal $p(\mathbf{y}) = \int_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})p(\mathbf{x})$ is a Gaussian

$$p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y} \middle| \mathbf{M}\boldsymbol{\mu}_x + \boldsymbol{\mu}, \mathbf{M}\boldsymbol{\Sigma}_x\mathbf{M}^{\mathsf{T}} + \boldsymbol{\Sigma}\right)$$

Decorrelating (whitening)

If $\mathbf{x}$ has covariance matrix $\boldsymbol{\Sigma}_x$ and mean $\boldsymbol{\mu}_x$, then

$$\mathbf{y} = \boldsymbol{\Sigma}_x^{-1/2}\left(\mathbf{x} - \boldsymbol{\mu}_x\right)$$

has mean $\mathbf{0}$ and identity covariance matrix. A commonly used initial transformation on data.