Mathematical background

Statistical Natural Language Processing

Çağrı Çöltekin

University of Tübingen
Seminar für Sprachwissenschaft

Summer Semester 2021

Some practical remarks

(recap)

- Course web page: `https://snlp2021.github.io` (public) `https://github.com/snlp2021/snlp/` (private)
- If you haven't already, please fill in the questionnaire on Moodle
- Assignment 1 will be released on Monday
  - Do not forget to add yourself to `assignments-match.txt` if you want to be assigned to a random team
- The first quiz will be released (on Moodle) after the class
- For those who need material on Python, see the private course repository for links to last semester's course
- If you prefer a book to study, the book by **bird2009** is a good option. For an update in progress see `https://www.nltk.org/book/`

Today's lecture

- Some concepts from linear algebra
- A (very) short refresher on
  - Derivatives: we are interested in maximizing/minimizing (objective) functions (mainly in machine learning)
  - Integrals: mainly for probability theory

This is only a high-level, informal introduction/refresher.

Linear algebra

Linear algebra is the field of mathematics that studies *vectors* and *matrices*.

- A vector is an ordered sequence of numbers

$$\boldsymbol{v} = (6, 17)$$

- A matrix is a rectangular arrangement of numbers

$$\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}$$

- A well-known application of linear algebra is solving a set of linear equations

$$\begin{aligned} 2x_1 + x_2 &= 6 \\ x_1 + 4x_2 &= 17 \end{aligned} \iff \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 17 \end{bmatrix}$$

Why study linear algebra?

Consider an application counting words in multiple documents

	the	and	of	to	in	...
document_1	121	106	91	83	43	...
document_2	142	136	86	91	69	...
document_3	107	94	41	47	33	...
...	...	...	...	...	...	...

You should already be seeing vectors and matrices here.

Why study linear algebra?

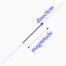
- Insights from linear algebra are helpful in understanding many NLP methods
- In machine learning, we typically represent input, output, parameters as vectors or matrices (or tensors)
- It makes notation concise and manageable
- In programming, many machine learning libraries make use of vectors and matrices explicitly
- In programming, vector-matrix operations correspond to loops
- 'Vectorized' operations may run much faster on GPUs, and on modern CPUs

Vectors

- A vector is an ordered list of numbers $\boldsymbol{v} = (v_1, v_2, \ldots v_n)$,
- The vector of n real numbers is said to be in *vector space* $\mathbb{R}^n$ ($\boldsymbol{v} \in \mathbb{R}^n$)
- In this course we will only work with vectors in $\mathbb{R}^n$
- Typical notation for vectors:

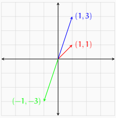
$$\boldsymbol{v} = \vec{v} = (v_1, v_2, v_3) = \langle v_1, v_2, v_3 \rangle = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

- Vectors are (geometric) objects with a magnitude and a direction



Geometric interpretation of vectors

- Vectors (in a linear space) are represented with arrows from the origin
- The endpoint of the vector $\boldsymbol{v} = (v_1, v_2)$ correspond to the Cartesian coordinates defined by v_1, v_2
- The intuitions often (!) generalize to higher dimensional spaces



Vector norms

- The *norm* of a vector is an indication of its size (magnitude)
- The norm of a vector is the distance from its tail to its tip
- Norms are related to distance measures
- Vector norms are particularly important for understanding some machine learning techniques

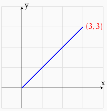
L2 norm

- Euclidean norm, or L2 (or L_2) norm is the most commonly used norm
- For $\boldsymbol{v} = (v_1, v_2)$,

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

$$\|(3,3)\|_2 = \sqrt{3^2 + 3^2} = \sqrt{18}$$

- L2 norm is often written without a subscript: $\|\boldsymbol{v}\|$



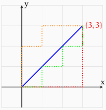
L1 norm

- Another norm we will often encounter is the L1 norm

$$\|\boldsymbol{v}\|_1 = |v_1| + |v_2|$$

$$\|(3,3)\|_1 = |3| + |3| = 6$$

- L1 norm is related to Manhattan distance



L_p norm

In general, L_p norm, is defined as

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

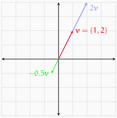
We will only work with than L1 and L2 norms, but you may also see L_0 and L_∞ norms in related literature

Multiplying a vector with a scalar

- For a vector $\boldsymbol{v} = (v_1, v_2)$ and a scalar a,

$$a\boldsymbol{v} = (av_1, av_2)$$

- multiplying with a scalar 'scales' the vector



Vector addition and subtraction

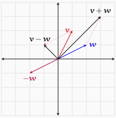
For vectors $\boldsymbol{v} = (v_1, v_2)$ and $\boldsymbol{w} = (w_1, w_2)$

- $\boldsymbol{v} + \boldsymbol{w} = (v_1 + w_1, v_2 + w_2)$

$$(1, 2) + (2, 1) = (3, 3)$$

- $\boldsymbol{v} - \boldsymbol{w} = \boldsymbol{v} + (-\boldsymbol{w})$

$$(1, 2) - (2, 1) = (-1, 1)$$



Dot (inner) product

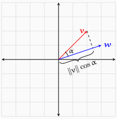
- For vectors $\boldsymbol{w} = (w_1, w_2)$ and $\boldsymbol{v} = (v_1, v_2)$,

$$\boldsymbol{w}\boldsymbol{v} = w_1v_1 + w_2v_2$$

or,

$$\boldsymbol{w}\boldsymbol{v} = \|w\|\|v\| \cos \alpha$$

- The *dot product* of two orthogonal vectors is 0
- $\boldsymbol{w}\boldsymbol{w} = \|\boldsymbol{w}\|^2$
- Dot product may be used as a similarity measure between two vectors



Cosine similarity

- The cosine of the angle between two vectors

$$\cos \alpha = \frac{\boldsymbol{v}\boldsymbol{w}}{\|\boldsymbol{v}\|\|\boldsymbol{w}\|}$$

is often used as another similarity metric, called *cosine similarity*
- The cosine similarity is related to the dot product, but ignores the magnitudes of the vectors
- For unit vectors (vectors of length 1) cosine similarity is equal to the dot product
- The cosine similarity is bounded in range $[-1, +1]$

Matrices

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,m} \\ a_{2,1} & a_{2,2} & a_{2,3} & \dots & a_{2,m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \dots & a_{n,m} \end{bmatrix}$$

- We can think of matrices as collection of row or column vectors
- A matrix with n rows and m columns is in $\mathbb{R}^{n \times m}$
- Most operations in linear algebra also generalize to more than 2-D objects
- A *tensor* can be thought of a generalization of vectors and matrices to multiple dimensions

Transpose of a matrix

Transpose of a $n \times m$ matrix is an $m \times n$ matrix whose rows are the columns of the original matrix.
Transpose of a matrix $\mathbf{A}$ is denoted with $\mathbf{A}^T$.

$$\text{If } \mathbf{A} = \begin{bmatrix} a & b \\ c & d \\ e & f \end{bmatrix}, \mathbf{A}^T = \begin{bmatrix} a & c & e \\ b & d & f \end{bmatrix}.$$

Multiplying a matrix with a scalar

Similar to vectors, each element is multiplied by the scalar.

$$2\begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} = \begin{bmatrix} 2 \times 2 & 2 \times 1 \\ 2 \times 1 & 2 \times 4 \end{bmatrix} = \begin{bmatrix} 4 & 2 \\ 2 & 8 \end{bmatrix}$$

Matrix addition and subtraction

Each element is added to (or subtracted from) the corresponding element

$$\begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 4 \end{bmatrix}$$

Note:

- Matrix addition and subtraction are defined on matrices of the same dimensions

Matrix multiplication

- if $\mathbf{A}$ is a $n \times k$ matrix, and $\mathbf{B}$ is a $k \times m$ matrix, their product $\mathbf{C}$ is a $n \times m$ matrix
- Elements of C, $c_{i,j}$, are defined as

$$c_{ij} = \sum_{\ell=0}^{k} a_{i\ell}b_{\ell j}$$

- Note: $c_{i,j}$ is the dot product of the i^{th} row of $\mathbf{A}$ and the j^{th} column of $\mathbf{B}$

Matrix multiplication

(demonstration)

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ a_{21} & a_{22} & \dots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nk} \end{pmatrix} \times \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1m} \\ b_{21} & b_{22} & \dots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{k1} & b_{k2} & \dots & b_{km} \end{pmatrix}$$

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots a_{ik}b_{kj}$$

$$= \begin{pmatrix} c_{11} & c_{12} & \dots & c_{1m} \\ c_{21} & c_{22} & \dots & c_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \dots & c_{nm} \end{pmatrix}$$

Dot product as matrix multiplication

In machine learning literature, the *dot product* of two vectors is often written as

$$\boldsymbol{w}^T\boldsymbol{v}$$

For example, $\boldsymbol{w} = (2, 2)$ and $\boldsymbol{v} = (2, -2)$,

$$\begin{bmatrix} 2 & 2 \end{bmatrix} \times \begin{bmatrix} 2 \\ -2 \end{bmatrix} = 2 \times 2 + 2 \times -2 = 4 - 4 = 0$$

- This is a 1×1 matrix, but matrices and vectors with single entries are often treated as scalars

Outer product

The *outer product* of two column vectors is defined as

$$\boldsymbol{v}\boldsymbol{w}^T$$

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{bmatrix}$$

Note:

- The result is a matrix
- The vectors do not have to be the same length

Identity matrix

- A square matrix in which all the elements of the principal diagonal are one and all other elements are zero is called *identity matrix* ($\mathbf{I}$)

$$\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$$

- Multiplying a matrix with the identity matrix has no affect

$$\mathbf{IA} = \mathbf{A}$$

Matrix multiplication as transformation

- Multiplying a vector with a matrix transforms the vector
- Result is another vector (possibly in a different vector space)
- Many operations on vectors can be expressed with multiplying with a matrix (linear transformations)

Transformation examples

identity

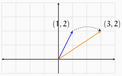
- Identity transformation maps a vector to itself
- In two dimensions:

$$\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix} \times \begin{bmatrix}x\\y\end{bmatrix} = \begin{bmatrix}x\\y\end{bmatrix}$$

Transformation examples

stretch along the x axis

$$\begin{bmatrix}3 & 0\\0 & 1\end{bmatrix} \times \begin{bmatrix}1\\2\end{bmatrix} = \begin{bmatrix}3\\2\end{bmatrix}$$

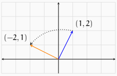


Transformation examples

rotation

$$\begin{bmatrix}\cos\theta & -\sin\theta\\\sin\theta & \cos\theta\end{bmatrix}$$

$$\begin{bmatrix}0 & -1\\1 & 0\end{bmatrix} \times \begin{bmatrix}1\\2\end{bmatrix} = \begin{bmatrix}-2\\1\end{bmatrix}$$



Linear maps or linear functions

- A linear function has the properties:
  - $f(\mathbf{x} + \mathbf{y}) = f(\mathbf{x}) + f(\mathbf{y})$ (additivity)
  - $f(a\mathbf{x}) = af(\mathbf{x})$ (homogeneity)

  or more generally,
  - $f(a\mathbf{x} + b\mathbf{y}) = af(\mathbf{x}) + bf(\mathbf{y})$
- A linear function can be expressed by matrix multiplication

Q: Is $f(x) = 2x + 1$ a linear function?

Matrix-vector representation of a set of linear equations

Our earlier example set of linear equations

$$\begin{aligned}2x_1 + x_2 &= 6\\x_1 + 4x_2 &= 17\end{aligned}$$

can be written as:

$$\underbrace{\begin{bmatrix}2 & 1\\1 & 4\end{bmatrix}}_{\boldsymbol{W}} \underbrace{\begin{bmatrix}x_1\\x_2\end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix}6\\17\end{bmatrix}}_{\mathbf{b}}$$

One can solve the above equation using *Gaussian elimination* (we will not cover it today).

Inverse of a matrix

Inverse of a square matrix $\boldsymbol{W}$ is denoted $\boldsymbol{W}^{-1}$, and defined as

$$\boldsymbol{W}\boldsymbol{W}^{-1} = \boldsymbol{W}^{-1}\boldsymbol{W} = \mathbf{I}$$

The inverse can be used to solve equation in our previous example:

$$\begin{aligned}\boldsymbol{W}\mathbf{x} &= \mathbf{b}\\\boldsymbol{W}^{-1}\boldsymbol{W}\mathbf{x} &= \boldsymbol{W}^{-1}\mathbf{b}\\\mathbf{I}\mathbf{x} &= \boldsymbol{W}^{-1}\mathbf{b}\\\mathbf{x} &= \boldsymbol{W}^{-1}\mathbf{b}\end{aligned}$$

Determinant of a matrix

$$\begin{vmatrix}a & b\\c & d\end{vmatrix} = ad - bc$$

The above formula generalizes to higher dimensional matrices through a recursive definition, but you are unlikely to calculate it by hand. Some properties:

- A matrix is invertible if it has a non-zero determinant
- A system of linear equations has a unique solution if the coefficient matrix has a non-zero determinant
- Geometric interpretation of determinant is the (signed) change in the volume of a unit (hyper)cube caused by the transformation defined by the matrix

Eigenvalues and eigenvectors of a matrix

An *eigenvector*, $\boldsymbol{v}$ and corresponding *eigenvalue*, λ, of a matrix $\mathbf{A}$ are defined as

$$\mathbf{A}\boldsymbol{v} = \lambda\boldsymbol{v}$$

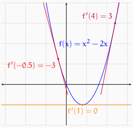
- Eigenvalues an eigenvectors have many applications from communication theory to quantum mechanics
- A better known example (and close to home) is Google's PageRank algorithm
- We will return to them while discussing PCA and SVD

Derivatives

- Derivative of a function $f(x)$ is another function $f'(x)$ indicating the rate of change in $f(x)$
- Alternatively: $\frac{df}{dx}(x)$, $\frac{df(x)}{dx}$
- Example from physics: velocity is the derivative of the position
- Our main interest:
  - the points where the derivative is 0 are the stationary points (maxima, minima, saddle points)
  - the derivative evaluated at other points indicate the direction and steepness of the curve defined by the function

Finding minima and maxima of a function

- Many machine learning problems are set up as optimization problems:
  - Define an error function
  - Finding the parameters minimizing the error
- We search for $f'(x) = 0$
- The value of $f'(x)$ on other points tell us which direction to go (and how fast)



Partial derivatives and gradient

- In ML, we are often interested in (error) functions of many variables
- A partial derivative is derivative of a multivariate function with respect to a single variable, noted $\frac{\partial f}{\partial x}$
- A very useful quantity, called *gradient*, is the vector of partial derivatives with respect to each variable

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

- Gradient points to the direction of the steepest change
- Example: if $f(x, y) = x^3 + yx$

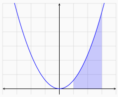
$$\nabla f(x, y) = \left(3x^2 + y, x\right)$$

Integrals

- Integral is the reverse of the derivative (anti-derivative)
- The indefinite integral of $f(x)$ is noted $F(x) = \int f(x)dx$
- We are often interested in definite integrals

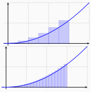
$$\int_a^b f(x)dx = F(b) - F(a).$$

- Integral gives the area under the curve



Numeric integrals & infinite sums

- When integration is not possible with analytic methods, we resort to numeric integration
- This also shows that integration is 'infinite summation'



Summary & next week

- Some understanding of linear algebra and calculus is important for understanding many methods in NLP (and ML)
- See bibliography at the end of the slides if you need a 'more complete' refresher/introduction
- Do not forget the weekly quiz!

Mon Probability theory
Wed Information theory

Further reading

- A classic reference book in the field is **strang2009**
- **shifrin2011** and **farin2014** are textbooks with a more practical/graphical orientation.
- **cherney2013**; **beezer2014** are two textbooks that are freely available.
- A well-known (also available online) textbook for calculus is **strang1991**
- Form more alternatives, see `http://www.openculture.com/free-math-textbooks`