

# Statistical Natural Language Processing

## Mathematical background: a refresher

Çağrı Çöltekin

University of Tübingen  
Seminar für Sprachwissenschaft

Summer Semester 2018

## Some practical remarks

(recap)

- Course web page:  
<http://sfs.uni-tuebingen.de/~ccoltekin/courses/snlp>
- Please complete Assignment 0
- Assignment 1 will be released this week
- Reminder: there are **Easter eggs** (in the version presented in the class)

## 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

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

- 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

$$\begin{array}{rcrcrcrcrcl} 2x_1 & + & x_2 & = & 6 & & & & \\ x_1 & + & 4x_2 & = & 17 & & \iff & & \end{array} \quad \begin{array}{c} \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 17 \end{bmatrix} \end{array}$$

## Why study linear algebra?

Consider an application counting words in multiple documents

	the	and	of	to	in	...
document <sub>1</sub>	121	106	91	83	43	...
document <sub>2</sub>	142	136	86	91	69	...
document <sub>3</sub>	107	94	41	47	33	...
...	...	...	...	...	...	...

You should already be seeing vectors and matrices here.

## Why study linear algebra?

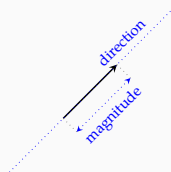
- Insights from linear algebra are helpful in understanding many NLP methods
- In machine learning, we typically represent input, output, parameters as vectors or matrices
- It makes notation concise and manageable
- In programming, many machine learning libraries make use of vectors and matrices explicitly
- 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  $\mathbf{v} = (v_1, v_2, \dots, v_n)$ ,
- The vector of  $n$  real numbers is said to be in *vector space*  $\mathbb{R}^n$  ( $\mathbf{v} \in \mathbb{R}^n$ )
- In this course we will only work with vectors in  $\mathbb{R}^n$
- Typical notation for vectors:

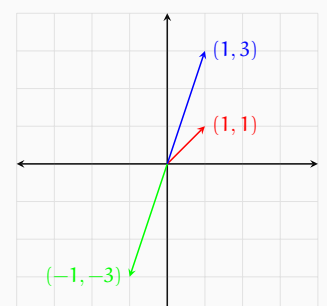
$$\mathbf{v} = \vec{v} = (v_1, v_2, v_3) = \langle v_1, v_2, v_3 \rangle = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

- 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  $\mathbf{v} = (v_1, v_2)$  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

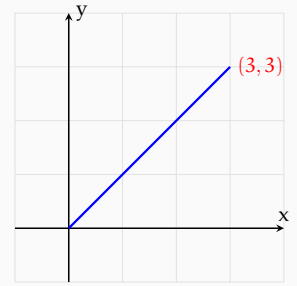
## L2 norm

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

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

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

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



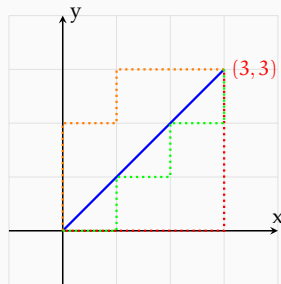
## L1 norm

- Another norm we will often encounter is the L1 norm

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

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

- L1 norm is related to Manhattan distance

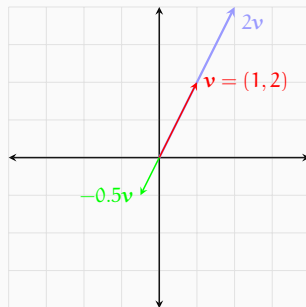


## Multiplying a vector with a scalar

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

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

- multiplying with a scalar 'scales' the vector



## Dot product

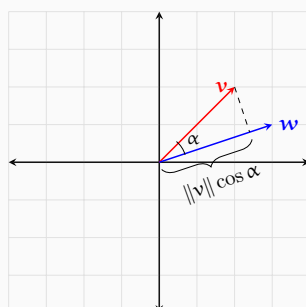
- For vectors  $\mathbf{w} = (w_1, w_2)$  and  $\mathbf{v} = (v_1, v_2)$ ,

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

or,

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

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



## $L_p$ norm

In general,  $L_p$  norm, is defined as

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

We will only work with than L1 and L2 norms, but  $L_0$  and  $L_\infty$  are also common

## Vector addition and subtraction

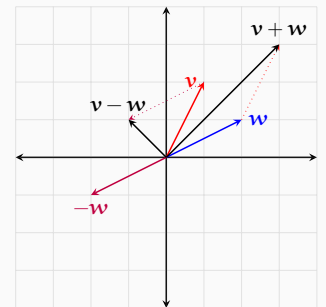
For vectors  $\mathbf{v} = (v_1, v_2)$  and  $\mathbf{w} = (w_1, w_2)$

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

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

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

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



## Cosine similarity

- The cosine of the angle between two vectors

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

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 matrices to multiple dimensions.

## 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 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  $\mathbf{C}$ ,  $c_{i,j}$ , 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  $i^{\text{th}}$  row of  $\mathbf{A}$  and the  $j^{\text{th}}$  column of  $\mathbf{B}$

## Dot product as matrix multiplication

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

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

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

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

\* This notation is somewhat sloppy, since the result of matrix multiplication is not a scalar.

## 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  $\mathbf{A}^T$ .

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

## 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 dimension

## Matrix multiplication

(demonstration)

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

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

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

## Outer product

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

$$\mathbf{w} \mathbf{w}^T$$

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

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 ones and all other elements are zeros, is called *identity matrix* and often denoted  $I$

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

- Multiplying a matrix with the identity matrix does not change the original matrix

$$IA = A$$

## Transformation examples

identity

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

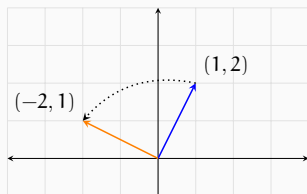
$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

## Transformation examples

rotation

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

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



## Inverse of a matrix

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

$$WW^{-1} = W^{-1}W = I$$

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

$$\begin{aligned} Wx &= b \\ W^{-1}Wx &= W^{-1}b \\ Ix &= W^{-1}b \\ x &= W^{-1}b \end{aligned}$$

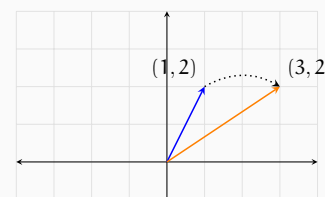
## Matrix multiplication as transformation

- 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

stretch along the x axis

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



## 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:

$$\underbrace{\begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}}_W \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_x = \underbrace{\begin{bmatrix} 6 \\ 17 \end{bmatrix}}_b$$

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

## 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

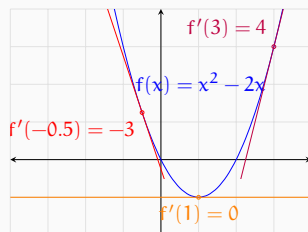
An *eigenvector*,  $\mathbf{v}$  and corresponding *eigenvalue*,  $\lambda$ , of a matrix  $\mathbf{A}$  are defined as

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

- Eigenvalues and 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 (and maybe more topics/concepts)

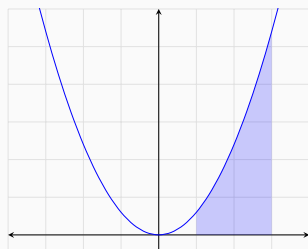
## Finding minima and maxima of a function

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



## 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



## 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

Wed Python tutorial (continued)

Fri We will do a similar excursion to probability theory

## 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

## Partial derivatives and gradient

- In ML, we are often interested in (error) functions of many variables
- A partial derivative is derivative of a multi-variate 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

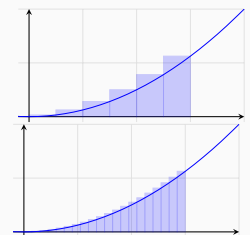
$$\nabla f(x_1, \dots, x_n) = \left( \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)$$

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

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

## 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'



## Further reading

- A classic reference book in the field is Strang (2009)
- Shifrin and Adams (2011) and Farin and Hansford (2014) are textbooks with a more practical/graphical orientation.
- Cherney, Denton, and Waldron (2013) and Beezer (2014) are two textbooks that are freely available.
- A well-known (also available online) textbook for calculus is Strang (1991)
- Form more alternatives, see <http://www.openculture.com/free-math-textbooks>



Beezer, Robert A. (2014). *A First Course in Linear Algebra*. version 3.40. Congruent Press. ISBN: 9780984417551. URL: <http://linear.ups.edu/>.



Cherney, David, Tom Denton, and Andrew Waldron (2013). *Linear algebra*. math.ucdavis.edu. URL: <https://www.math.ucdavis.edu/~linear/>.



Farin, Gerald E. and Dianne Hansford (2014). *Practical linear algebra: a geometry toolbox*. Third edition. CRC Press. ISBN: 978-1-4665-7958-3.

## Further reading (cont.)



Shifrin, Theodore and Malcolm R Adams (2011). *Linear Algebra. A Geometric Approach*. 2nd. W. H. Freeman. isbn: 978-1-4292-1521-3.



Strang, Gilbert (1991). "Calculus". In: *Wellesley-Cambridge press*. url: <https://ocw.mit.edu/resources/res-18-001-calculus-online-textbook-spring-2005/textbook/>.



Strang, Gilbert (2009). *Introduction to Linear Algebra, Fourth Edition*. 4th ed. Wellesley Cambridge Press. isbn: 9780980232714.