# Deep learning notes

Guillem Tobías Larrauri

January 2025

## 1 Theory: Deep learning goodfellow 2016

This first section is structured into math basics, then modern practical deep networks and finally some research topics(which are now maybe also working in practice)

### 1.1 Chap 5: ML basics

Many tasks you can do in Machine learning, they are cooler that just regression and classification.

- classification
- classification with missing inputs, biomed sciences!
- regression
- transcription and language related stuff: from image to text
- structrued output, output is a vector
- anomaly detection
- synthesis and sampling
- imputation of missing values
- denoising
- density estimation: read math book.

### 1.2 Chapter 6: Deep Feedforward Networks

No info going backwards in the layers of the model. The question is then how to choose the mapping $\psi$ .

The strategy of deep learning is to learn $\psi$ which is often in a parametrized form, this way there need not be manual feature engineering, just find the right class of general functions. It is truly like regular statistics.

The XOR problem shows how a linear classifier would fail and so we need to find some transformation of the underlying space that allows for better classification. If we think we are choosing amongst a large class of functions we can view the loss as a functional, mapping from a set of functions to a real number. And we find the best function.

Finding optimums of functionals is from calculus of variations, which is complicated. However, we have the typical result of the *conditional expectation* being optimal with squared error loss. Suposedly mse and mae are pretty bad for classification because some saturation stuff.

#### 1.2.1 Output functions

**LENGTHY DISCUSSION on sigmoid... recheck** Gaussian mixture outputs

#### 1.2.2 Hidden units

- relu
- sigmoid
- maxout, check in more depth

Unless indicated otherwise, most hidden units can be described as accepting a vector of inputs $\mathbf{x}$, computing an affine transformation $\mathbf{z} = \mathbf{W}'\mathbf{x} + \mathbf{b}$ and then applying an element wise nonlinear function $g(\mathbf{z})$ most hidden units only different is the choice of this activation function g(*)

#### 1.2.3 Architecture design

Some theorems but mainly trial and error.

#### 1.2.4 Backpropagation to compute gradients

Check out my math book it has very good example

### 1.3 Regularization for deep learning

A first step is to add a ridge penalty to the overall loss function.

#### 1.3.1 Dataset augmentation

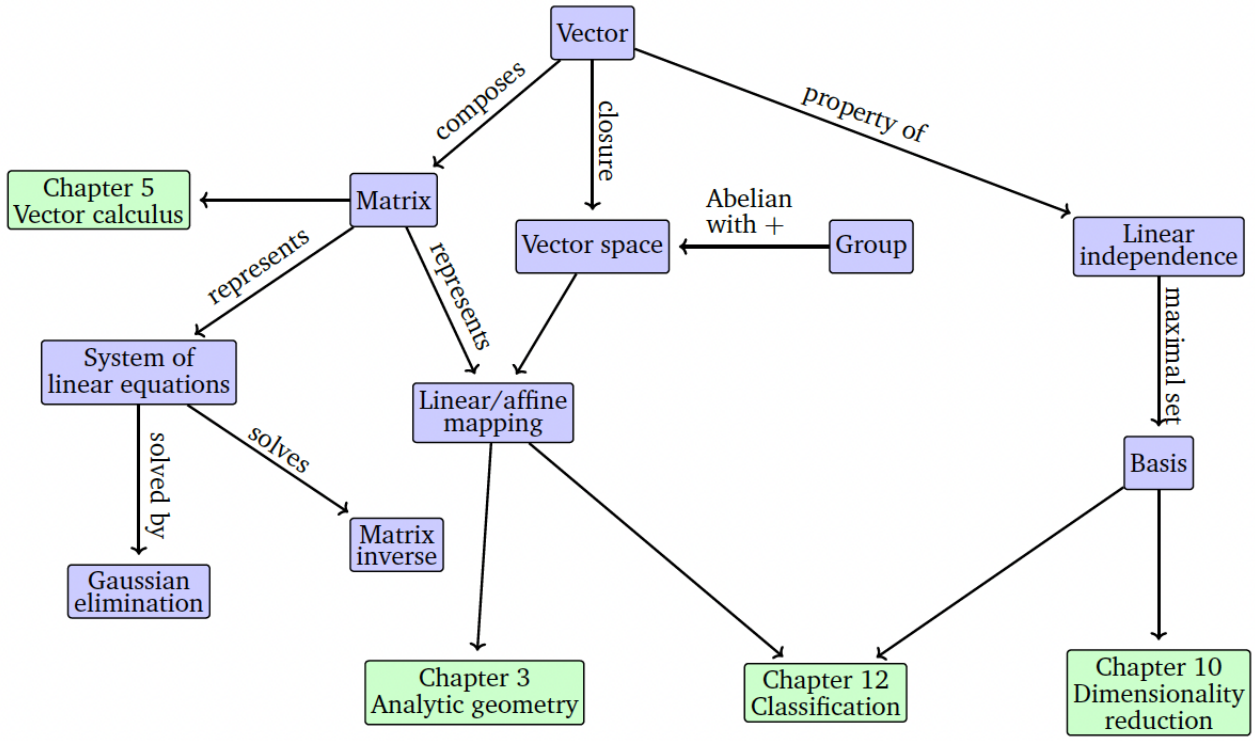
Create fake data, in terms of images you can try to rescale it or rotate it.

## 2 Math: From Mathematics for Machine learning

This will be copied over and is needed for ML and also Deep and Graph learning, I will be adding notes on which chapters matter most for where.

### 2.1 Linear Algebra

An algebra is a set of objects and a set of rules to manipulate these objects. Vectors are objects such that they can be added together and multiplied by a constant and the result is still a vector.




### 2.1.1 Systems of linear equations

We will mostly ignore for now.

### 2.1.2 Matrices

A matrix as an m * n tuple of elements $a_{ij}$, we define it as $\mathbf{A}$ We have some important properties of all matrices TO DO! $\forall \mathbf{A} \in \mathbb{R}^{m \times n} : \mathbf{I}_m \mathbf{A} = \mathbf{A}\mathbf{I}_n = \mathbf{A}$

The transpose and the inverse (which not always exists, but when it exists it is unique.) Some important properties of transpose and inverse:

- $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}$
- $(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$
- $(\mathbf{A} + \mathbf{B})^{-1} \neq \mathbf{A}^{-1} + \mathbf{B}^{-1}$
- $(\mathbf{A}^T)^T = \mathbf{A}$
- $(\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T$
- $\mathbf{A}\mathbf{B}^T = \mathbf{B}^T\mathbf{A}^T$

A matrix is symmetric if it is equal to its transpose and we also have $(\mathbf{A}^{-1})^T = (\mathbf{A}^T)^{-1} =: \mathbf{A}^{-T}$ For multiplication by a scalar all properties are super intuitive.

### 2.1.3 Solving systems of linear equations, we skip for now

We can represent compactly via $\mathbf{A}\mathbf{x} = \mathbf{b}$

If the matrix $\mathbf{A}$ is square we can directly compute the solution (if it exists) as $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ If the columns of a non square matrix are linearly independent we can compute the following: $\mathbf{A}\mathbf{x} = \mathbf{b} \iff \mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b} \iff \mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$

### 2.1.4 Vector spaces

Add the math if needed. *Groups:* We consider a set $\mathcal{G}$ and an operation $\otimes : \mathcal{G} \times \mathcal{G} \longrightarrow \mathcal{G}$

- Closure: For any x and y in the group, if we apply the operation to x and y the result will still be in the group
- Associativity For any 3 in the group the parentheses can be changed meaning we do one operation first or second and we get the same result.
- Neutral element: There exists in the group a neutral element such that if we apply the operation to anything in the group the result will be the original thing
- Inverse element: There exists for every element in the group another element such that if we apply the operation we will get the neutral element.

If we have commutative property (x * y = y * x) we have **Abelian group.** We then have some examples of what is and what is not a group.

**Vector spaces** A vector space will now have two operations, an inner operation and an outer operation.

- $+ : \mathcal{V} \times \mathcal{V} \longrightarrow \mathcal{V}$
- $* : \mathbb{R} \times \mathcal{V} \longrightarrow \mathcal{V}$

And:

- $(\mathcal{V}, +)$ is an Abelian group.
- We have distributivity with respect to the scalar/outer operation
- We have Associativity with respect to outer operation
- There exists a neutral element with respect to outer operation

Our inner operation will therefore be vector addition and the outer will be multiplication by scalars Vector multiplication is not defined for all as the dimensions may not be the same.

**Examples:** I don't understand why $\mathbb{R}^{m \times n}$ is a vector space if the inverse may not exist. AH, inverse with respect to inner operation, meaning find another matrix such that we can get the inverse. And that is always defined by Abelian, meaning commutative under the sum.

**Vector subspaces:** Subsets of the original space such that when we perform the operations (sum and multiplication) we will never leave the subspace.

We will naturally inherit many of the properties from the larger vector space. However we need to show that U is not empty, that the 0 element is in there and that we have closure with respect to both the outer and inner operations.

*Every subspace of Rn isthe solution space of a system of homogeneous linear equations.* I don't fully understand the implications of this.

### 2.1.5 Linear independence

Consider a vector space V and a finite number of vectors (k). Remember matrices are also vectors in the R nm vector space.

Every $\mathbf{v} \in V$ of the form $\mathbf{v} = \sum_{i=1}^{k} \lambda_i \mathbf{x}_i \in V$ where the lambdas are real numbers are a linear combination of the initial vectors x. Notice we apply both operations, the inner operation in the sum and the outer operation. $\mathbf{0}$ can always be written as a linear combination.

The notion of **linear independence** is the following: If from the original vectors we can build a non trivial linear combination that gives us the $\mathbf{0}$ vector the original vectors are linearly dependent. If only setting all lambdas to 0 gives us the solution the vectors are linearly independent. *There are a set of practical rules in the book such that we show if they are independent or not, you can also solve the*

*whole system (pain)*

The intuition is that if we drop any of the original vectors we are losing informaiton! tblueReview page 32 prett cool!

### 2.1.6 Basis and rank

In a vector space we are interested in a subset of it such that any vector in the larger space can be generated as a linear combination of the vectors in the smaller set. The set of all linear combinations is called the span of a set. If a subset A spans the whole vector space V we say V = span(A). Generating sets are sets are sets that span the vector space.

A **Basis** is: The minimal generating set. Can also be defined in terms of linear independence, it is the maximal linearly independent set of vectors in V Also uniqueness things, yesyes.

- Every vector space has a basis, however there is not unique basis. All basis have the same number of vectors.
- To find a basis we can see the original vectors, then solve $\sum_{i=1}^{k} \lambda_i \mathbf{x}_i = \mathbf{0}$ which can be put in matrix form Then find pivot columns and those will be the span.
- Example 2.14 is beautiful Solving linear systems is useful!!

The **Rank** of a matrix is closely defined: With it we can see the span of the columns and the rows. Columns are m dimensional and so on. Important remarks for solvable system and existance of an inverse, gp back adn write down!

### 2.1.7 Linear mappings

I get lost with this!!! Do a review in the future for now move on. Non linear mapping is if the two properties hold individually but not necessarily together. A linear mapping $\mathbf{v}$ from V to W vector if for any two vectors in the original space and any two real numbers we get $\Phi(\lambda\mathbf{x} + \psi\mathbf{y}) = \lambda\Phi(\mathbf{x}) + \psi\Phi(\mathbf{y})$ Therefore the result is still a vector space, added and multiplied is still preserved. It is going to be a different vector space though! We can represent linear mappings as **matrices** and we will see more about this.

Special linear mappings:

- Injective, if we have two vectors equal after transform they were equal in the original space.
- Surjective, we can reach any point in the second vector space from the first vector space using the mapping.
- Bijective if 1 and 2 hold

Bijective mappings can be undone with the inverse mapping, so this guarantees existance or whatever. Isomorphic is if linear and bijective from V to W.

We have a theorem that says that V and W are isomophic iff they have the same dimension. This justifies treating $\mathbb{R}^{m\times n} = \mathbb{R}^{mn}$ we can unstack the matrix and not lose any information. Because there exists a bijective linear mapping that goes from one to the other.

There are some extra properties that I can write down in p 38.

**Matrix representation of linear mappings:** Bottom line is that linear mappings can be represented as matrices.

Any n dimensional vector space is isomophic to $\mathbb{R}^n$ We will then define $B = (\mathbf{b}_1, ..., \mathbf{b}_n)$ as an ordered basis.

Then for any vector space V with its ordered basis B, we can define any vector inside the space as $\sum_{i=1}^{n} \alpha_i \mathbf{b}_i$ and this representation is unique, and the alphas are the coordinate representation of the vector x with basis B. Let's say we are in R2 and we have the canonical basis ((0,1),(1,0)) to represent (2,2) we will pick alphas to be 2 and 2. But if we have a different basis (something that equivalently defines the space in R2) our alphas will be very dfifferent. Say our basis is now (-1,1) and (0,1), to represent (2,2) we would need aplhas to be (4,2)

**2.19** We now have 2 vector spaces, V, W with the following bases B = $(\mathbf{b}_1, ..., \mathbf{b}_n) and C(\mathbf{c}_1, ..., \mathbf{c}_m)$ and we consider a linear mapping from V to W: $\Phi$ Then for all j: $\Phi(\mathbf{b}_j) = \sum_{i=1}^{m} \alpha_{ij}\mathbf{c}_i$ is the unique representation of $\Phi(\mathbf{b}_j)$ with respect to c.

By the definitons we get to $\hat{\mathbf{y}} = A_\Phi \hat{\mathbf{x}}$ where x are coordinates of a vector in the original basis and we use the matrix to transform to the new basis **I don't understand example 2.21!!!** Use chatgpt, vector in old space maps to the new one. But the transf matrix is about basis, each basis element in old space can be represented a certain way in the new one. And because all vectors in a space can be represented as a linear combination of the basis we can apply this to all vectors in the space. **Example 2.22** We use a linear transform to get new coordinates and expand the square applying the transformation matrix.

**Basis change:** What happens to transformation matrix if we change the basis in the old or new space we can exploit this to find easy transformation matrices. For now I skip it.

**Image and kernel:** For linear mappings:

The kernel is the set of vectors in the original space such that applying the linear mapping gives you the 0 vector in the new space. The image is the set of vectors in the new space by applying the linear mapping into any vector of the old space. Contain many interesting properties but they seem very inutitive, left for another time.

### 2.1.8 Affine spaces

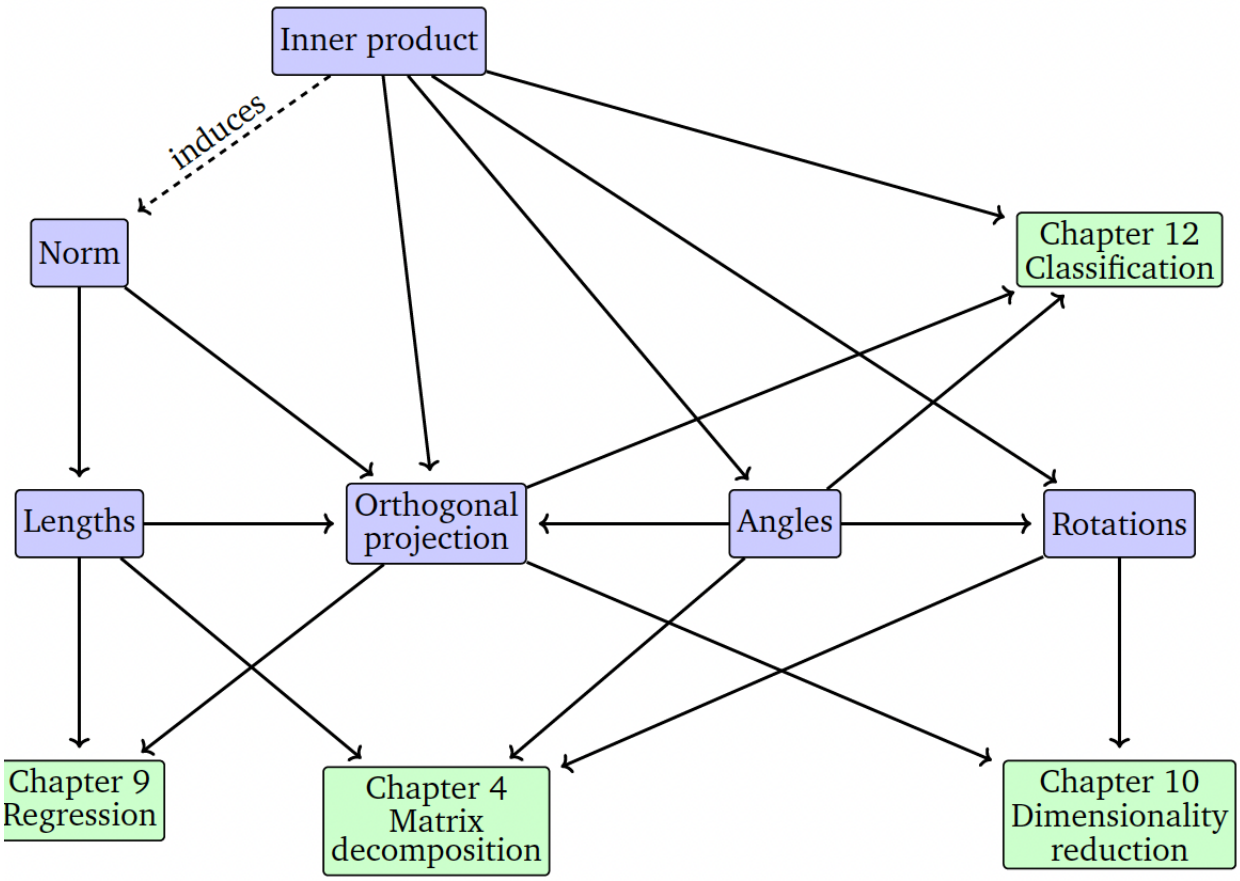
Skipped this for now

### 2.1.9 Exercises and complete above

**2,17,...**

## 2.2 Analytic geometry

Goes into geometric interpretation for linear algebra, lengths and distances between two vectors inner products and norms and finally orthogonal projections.




### 2.2.1 Norms

A norm is a defined on a vector space and goes from the vector space to a real number. It defines the length of a vector. To be a proper norm the following need to hold for any two vectors in the space and any real number lambda:

- $\|\lambda \mathbf{x}\| = abs(\lambda)\|\mathbf{x}\|$
- $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$
- Positive definite meaning larger or equal than 0 and 0 for the 0 vector.

We usually use the Euclidean norm $\sqrt{\mathbf{x}^T\mathbf{x}}$ which computes the eucldiean distance from the origin.

### 2.2.2 Inner products

Allow introduction of geometrical concepts like length angle and distance between two vectors. The simplest is the dot product.

The general definition of a inner product is as a positive definite (no neg distances), symmetric(distance a to b is same as b to a) and bilinear (why not) such that it maps from $V \times V \to \mathbb{R}$

Bilinear property: For any x y z in vector space, and two real numbers: The inner product will satisfy the following:

$$\Omega(\lambda x + \psi y, z) = \lambda\Omega(x, z) + \psi\Omega(y, z) \quad \Omega(x, \lambda y + \psi z) = \text{the equivalent}$$

**Symmetric positive definite matrices** The inner product is determined thorugh a positive definite matrix (all these matrices are symmetric, so symmetric positive definite) Because all vectors in the

space can be represented through a linear combination of basis functions we can end up representing the inner product as

$$\langle x, y\rangle = \text{properties fo the norm, bilinear} = \sum_{i=1}^{n}\sum_{j=1}^{n} \psi_i \langle b_i, b_j\rangle \lambda_j = \hat{x}' A \hat{y}$$

Where the elements of A are norms amongst basis vectors..0

### 2.2.3 Lengths and distances

Any inner product induces a norm: $\|x\| := \sqrt{(\langle x, x\rangle)}$ This induced norm will always satisfy the Cauchy-Schwartz inequality: $|\langle x, y\rangle| \leq \|x\|\|y\|$

We can define distances with these elements:

$$d(x, y) := ||x - y|| = \sqrt{\langle x - y, x - y\rangle}$$

the second step is only if the norm is induced by the inner product which may not always be true.

**Inner product and distance behave in opposite directions, if large similarity we have large inner product, and low distance, and viceversa.**

### 2.2.4 Angles and orthogonality

By Cauchy Schwartz inequality we can define an angle between two vectors, which can be ignored without too much issue.

*orthogonality:* When inner product between two vectors is 0, if the vectors have norm 1 we call and are orthogonal we call them orthonormal.

Matrices are orthogonal if all columns are orthonormal, we define the inner product as dot product here I believe. $AA' = I = A'A$ the inverse is the transpose. These matrices preserve both angles and distances.

### 2.2.5 Orthonormal basis

Each basis vector is orthogonal to all the others and it has length one. They can be built via the Gram Schmidt process and they are used for PCA.

### 2.2.6 Orthogonal complement

We have a V (D dimensional) and a subset U (m dimensional), the orthogonal component has dimensions (D-M) and contains all vectors in V that are orthogonal to every vector in U. Morover we can represent any vector in V through a linear combination of the basis of U and the basis of its orthogonal complement. The example is pretty nice

### 2.2.7 Inner prodcuts of functions

THey are integrals

### 2.2.8 Orthogonal projections

Let V be a vector space and $u \subseteq V$. A linear mapping $\pi : V \to U$ is called a projection if $\pi^2 = \pi \circ \pi = \pi$ Because a linear mapping can be represented via a transformation matrix, we will call projection matrices those that have this given property.

**Projection onto one dimensional subspaces** We are really trying to solve a minimization problem. Read and mostly understood, write down another time. proof of orthogonality by contradiction in notebook

Do it with ols

## 2.3 Matrix decomposition

Determinants and eigenvalues, Cholesky and Diagonalization and SVD

### 2.3.1 Determinant and trace

The determinant is only defined for square matrices. It determines invertibility, that is if the determinant is 0 the matrix is not invertible.

The determinant has some properties. Not that used anymore but important for eigen stuff.

The trace is the sum of diagonal elements.

Add properties if they come up again in the book.

### 2.3.2 Eigenvalues and eigenvectors:

Defined from the equation $\mathbf{Ax} = \lambda\mathbf{x}$ where lambda is the igenvalue for x the eigenvector. The 0 vector is always a solution but we ignore it.

Eigenvectors are not unique they can always be scaled and the result will not change. Eigenvectors are roots of the characteristic polynomial, that is why $det(\mathbf{A} - \lambda\mathbf{I}_n) = 0$

All eigenvectors of A span a subspace called the eigenspace.

Some properties:

- A matrix and its transpose have the same eigenvalues but not necessarily the same eigenvectors.
- The eigenspace is the null space of $\mathbf{A} - \lambda\mathbf{I}$
- Something about similar matrices Review
- Symmetric positive definite matrices have all positive real eigenvalues.

N eigenvectors from a Matrix with distinct eigenvalues are linearly independent, that means they form a basis of Rn.

We can obtain a symmetric positive definite matrix from any $\mathbf{A} \in \mathbb{R}^{m \times n}$ by defining the following: $S := \mathbf{A}'\mathbf{A}$

If A is symmetric there exists an orthonormal basis consisting of eigenvectors of A and all eigenvalues are real. We can thus decompose a matrix A = PDP' where the P contain the eigenvecrtors are columns.

finally, the determinant of a matrix is the product of eigenvalues and the trace is the sum of eigenvalues.

### 2.3.3 Cholesky decomposition

A symmetric positive definite matrix can be decomposed into a lower triangular matrix and its transpose. $\mathbf{A} = \mathbf{LL}'$ This lower triangular matrix is unique. Used in practice

### 2.3.4 Eigendecomposition:

With diagonal matrices and blablabla diagonal is eigenvalues.

### 2.3.5 Singular value decomposition

$$A = U\Sigma V'$$

Where A is m * n, U is m*m, sigma is m * n and V' is n n Both U and V are orthonormal and sigma is a diagonal matrix.

Cover in much more depth as well as matrix apporximation and the movie example!!!

# 3 Tensorflow