

TODO: Rework 1.1

# 1 Disambiguations and definitions

## 1.1 Naming conventions

### 1.1.1 Basic notions

Let's recall the naming conventions of basic notions.

A *function*  $f : E \rightarrow F$  maps objects  $x \in E$  to objects  $y \in F$ , as  $y = f(x)$ .

Its *definition domain*  $\mathcal{D}_f = E$  is the set of objects onto which it is defined. We will often just use the term *domain*.

We also say that  $f$  is *taking values* in its *codomain*  $F$ .

The *image per  $f$*  of the subset  $U \subset E$ , denoted  $f(U)$ , is  $\{y \in F, \exists x \in U, y = f(x)\}$ .

The *image of  $f$*  is the image of its domain. We denote  $\mathcal{I}_f$ .

A vector space  $E$ , which we will always assume to be finite-dimensional in our context, is defined as  $\mathbb{R}^n$ , and is equipped with pointwise addition and scalar multiplication.

A *signal*  $s$  is a function taking values in a vector space. In other words, a signal can also be seen as a *vector* with an *underlying structure*, where the vector is composed from its image, and the underlying structure is defined by its *domain*.

For example, images are signals defined on a set of pixels. Typically, an image  $s$  in RGB representation is a mapping from pixels  $p$  to a 3d vector space, as  $s_p = (r, g, b)$ .

TODO?: figure

### 1.1.2 Graphs and graph signals

TODO: more defs on grid graphs and other graphs

A *graph*  $G = (V, E)$  is defined as a set of nodes  $V$ , and a set of edges  $E \subseteq \binom{V}{2}$ . The words *node* and *vertex* will be used equivalently, but we will rather use the first.

A *graph signal*, or *graph-structured signal* is a signal defined on the nodes of a graph, for which the underlying structure is the graph itself. A *node signal* is a signal defined on a node, in which case it is a *node embedding* in a vector space. Although this is rarely seen, a signal can also be defined on the edges of a graph, or on an edge. We then coin it respectively *dual graph signal*, or *edge signal / edge embedding*.

*Graph-structured data* can refer to any of these type of signals.

### 1.1.3 Data and datasets

A dataset of signals is said to be *static* if all its signals share the same underlying structure, it is said to be *non-static* otherwise.

For image datasets, being non-static would mean that the dataset contains images of different sizes or different scales. For graph signal datasets, it would mean that the underlying graph structures of the signals are different.

The point in specifying that objects of a dataset of a machine learning task are signals is that we can hope to leverage their underlying structure.

TODO: figure

## 1.2 Disambiguation of the subject

This thesis is entitled *Deep learning models for data without a regular structure*. So either the data of interest in this manuscript do not have any structure, or either their structure is not regular.

### 1.2.1 Irregularly structured data

By structured data, we mean that there exists an underlying structure over which the data is defined. This kind of data are usually modeled as signals defined over a domain. These domains are then composed of objects that are related together by some sort of structural properties. For example, pixels of images can be seen as located on a grid with integer spatial coordinates (a 2d cartesian grid graph).

It then comes in handy to define the notions of structure and regularity with the help of graph signals.

#### Definition 1.1. Structure

Let  $s : D \rightarrow F$  be a signal defined over a finite domain.

An *underlying structure* of the signal  $s$  is a graph  $G$  that has the domain of  $s$  for nodes.

A dataset is said to be *structured*, if its objects can be modeled as signals with an underlying structure.

It is said to be *static* if all its objects share the same underlying structure, and *non-static* otherwise.

In other words, we chose to define “structured data” as “graph-structured data” by some graph. Hence we need to specify for which graphs this structure would be said to be regular, and for which it would not.

#### Definition 1.2. Regularity

An underlying structure is said to be *regular*, if it is a regular grid graph. It is said to be *irregular* otherwise.

A dataset is said to be *regularly structured*, if the underlying structures of its objects are regular. It is said to be *irregularly structured* otherwise.

TODO: examples

### 1.2.2 Unstructured data

Data can also be unstructured. If the data is not yet embedded into a finite dimensional vector space, then we will be interested in embedding techniques used in representation learning. In the other case, it is often possible to fall back to the case of irregularly structured data. For example, vectors can be seen as signals defined over the canonical basis of the vector space, and the vectors of this basis can be related together by their covariances through the dataset. It is typical to use the graph structure that has the canonical basis for nodes, with edges obtained by covariance thresholding.

TODO: examples

What follows is a draft

## 1.3 Theoretical results on regularity and convolutions

## 2 Datasets

### 2.1 Tasks

## 3 Goals

### 3.1 Invariance

**Definition 3.1.**

In order to be observed, invariances must be defined relatively to an observation. Let's give a formal definition to support our discussion.

...

### 3.2 Methods