Group nuisances, invariance and equivariance
Draft of notes for CS 103

Alessandro Achille
Department of Computer Science
University of California, Los Angeles

1 Task and nuisances

Assume we observe input data $x \sim p(x)$, sampled from a (possibly unknown) distribution $p(x)$. To fix the ideas, in the following we will think of x as an image, but it can be data in any format and *sensory modality*, such as a feature vector, or an audio signal. Given this data, we want to infer the distribution of an hidden random variable $y \sim p(y|x)$, which we will refer as our **task**.

This setting is quite general, and includes several (if not most) common problems in machine learning. For example:

- **Image classification:** In this case x is an image, and y is the label associated to the image (*e.g.*, cat, dog, ...).
- **Object detection:** x is again an image, while y is a collection of bounding boxes (encoding size and position of the objects), and the class of the object.
- **3-D reconstruction:** x are multiple images, or a video sequence, and y is the 3-D geometry (descibed as a set of surfaces or a cloud of points) associated to the scene.

In general, the observed image x can depend on a number of factors, also called **factors of variation**. For example, the image of an object depends on: the 3D shape of the object, the texture of the object, the illumination of the scene, and the point of view from which the picture is taken.

Some of these factors carry information about the task: For example in an object classification task, changing the shape of an object is also likely to change its class and therefore change the answers to the task. On the other hand, other factors do not influence the task: for example, changing the illumination, or the point of view should not change the answer to an object classification task. We refer to these as "nuisance factors" for the task. More formally:

Definition 1. Assume without loss of generality that our input data x can be written as $x = I(e, n)$ for $e, n \sim p(e, n)$. We say that n is a **nuisance factor** for the task y if

$$p(y|I(e, n)) = p(y|I(e, n')),$$

for all $e \in \mathcal{E}$ and $n \in \mathcal{N}$. Note that this is the same as saying that $I(y; n) = 0$, that is, the nuisance factor n does not carry information about the task variable y.

Notice that the definition of nuisance depends on the task. For example, assume the task is to recognize whether two images are pictures of the same person or not. Then, the particular clothes worn by the person are a nuisance for the task. On the other hand, if the task is to classify the style of the objects in the picture, then the roles would be inverted: the clothes would not be a nuisance, and instead the identity of the person wearing them would now become a nuisance.

2 Nuisance invariance

Most of the information contained in the image is due to nuisance variability [2]. If we could find a representation $z = \phi(x)$ of x that is not affected by nuisance variability, we could reduce the effective dimension of the input space without losing information about y, therefore potentially simplifying the learning problem. This motivates the following definition:

Definition 2. We say that a representation $\phi(x) : \mathcal{X} \to \mathcal{Z}$ is invariant to a nuisance n if

$$\phi(I(e, n)) = \phi(I(e, n')),$$

for all $e \in \mathcal{E}$ and $n, n' \in \mathcal{N}$.

However, we now face two problems:

1. It is often very difficult construct an invariant representation for a given nuisance. Indeed, it can be as hard as solving the task itself;
2. Nuisances depends on the task, so we would need to build a different representation for each task.

The good news are that many tasks in computer vision share a common subset of nuisances, which have a relatively simpler structure. Examples of such common nuisances are: translations and rotations of the image plane, change of contrast, illumination, change of point of view. Some of these nuisances further have a *group structure*, which makes finding invariant representations particularly amenable, as we will now see.

3 Group nuisances

3.1 Groups and group actions

Recall that a **group** is a set G, together with a *group operation* $\cdot : G \times G \to G$ which satisfies the following axioms:

1. There is an element $e \in G$ called *group identity* such that $e \cdot g = g \cdot e = g$ for each $g \in G$;
2. For each $g \in G$ there is $g^{-1} \in G$, called the *inverse* of g, such that $g \cdot g^{-1} = g^{-1} \cdot g = e$;
3. The operation is *associative*: For each $g, h, k \in G$, we have $(g \cdot h) \cdot k = g \cdot (h \cdot k)$.

When clear from the context, the group operation is omitted and we write ab instead of $a \cdot b$. A useful property derived from the above axioms is the rule for the inverse of a product $(ab)^{-1} = b^{-1}a^{-1}$.

Examples. The following are some of the groups that are of interest to us:

1. The group $(\mathbb{Z}, +)$ of integers with addition. In this case, $0 \in G$ is the group identity and $-x$ is the inverse of x;
2. The group $\mathbb{Z}/\mathbb{Z}_k = \{0, 1, \ldots, k-1\}$ of integers with addition modulo k. This is also known as the *cyclic group of order* k, and is sometimes denoted with C_k or Z_k, or written in multiplicative form as $C_k = \{e, r, \ldots, r^{k-1}\}$, where $r^a \cdot r^b = r^{a+b}$ and $r^k = e$;
3. The group of rotations of the plane that are multiples of $2\pi/k$ degrees, or, equivalently, the group of rotational symmetries of a regular polyhedron with k vertexes. Notice that this group can be identified with the cyclic group $C_k = \{e, r, \ldots, r^{k-1}\}$, where r is the rotation of $2\pi/k$ degrees;
4. The group of translations in the plane. This can be identifies with $G = \mathbb{R} \times \mathbb{R}$, where the composition of two translation $(u, v), (x, y) \in \mathbb{R} \times \mathbb{R}$ is given by the sum $(u + x, v + y)$.
5. The group $\mathbb{GL}(n)$ of invertible $n \times n$ matrixes with multiplication (the group identity is of course the identity matrix I);
6. The group of 3-D rotation matrixes $SO(3)$;
7. The group S_n of permutations of n elements, which can equivalently be written as the group of bijective fucntions $f : \{1, \ldots, n\} \to \{1, \ldots, n\}$ with composition between functions.

Given a group G and a set X, we say that a function $\cdot : G \cdot X \to X$ is an **action** of G on X if:

1. $e \cdot x = x$ for each $x \in X$, where $e \in G$ is the group identity;
2. $(gh) \cdot x = g \cdot (h \cdot x)$.

Group actions are an important tool as they endow a set X with additional structure deriving from the group. In our case, X will often be a set of images, and G a group of nuisances acting on them.

Examples.

1. Action of translations on images: Let $\mathcal{X} = \{f : \mathbb{R}^2 \to \mathbb{R}\}$ be the set of all gray-scale (infinitely large) images. Define the translation $g = (u, v)$ on an image $f(x, y)$ as $(g \cdot f)(x, y) := f(x - u, y - v)$.

2. Action of rotations on images: consider again the continuous grey-scale images, and let $r \in SO(2)$ be a rotation of θ degrees. We can define the action $(g \cdot f)(x, y) = f(\cos(\theta)x + \sin(\theta), \sin(\theta)x - \cos(\theta)y)$

3.2 Group nuisances

Definition 3. We say that n is a group nuisance if we there is a group G and an action $\cdot : G \times \mathcal{I} \to G$, such that for any $n, n' \in N$ we can find $g_{n \to n'} \in G$ such that $I(e, n') = g_{n \to n'} \cdot I(e, n)$.

3.3 Invariance and equivariance

Given two spaces X and Y, and a group G which acts on X with action $\cdot_X$ and on Y with action $\cdot_Y$, we say that a representation $\phi : X \to Y$ is G-equivariant if

$$\phi(g \cdot_X x) = g \cdot_Y \phi(x),$$

for all $g \in G$ and $x \in X$. We say that ϕ is G-invariant if

$$\phi(g \cdot_X x) = \phi(x).$$

Notice that invariance is a particular case of equivariance when the action on Y is trivial, *i.e.*, when $g \cdot_Y y = y$ for all $y \in Y$.

3.4 Linear equivariant representations

Note: This section follows [1], but assuming a simpler setting to simplify the proof.

Given a set of indices X and a vector space V, let $L_V(X) := \{f : X \to V\}$ be the vector space of functions from X to V.[1] When $V = \mathbb{R}$, we simply write $L(X)$. Notice that if a group G acts on the set of indices X, this action can be extended to $L(X)$ naturally using $(g \cdot f)(x) := f(g^{-1}x)$.

This notation is useful to define several type of data in machine learning. For example:

1. Continuous black and white images are a function $f : \mathbb{R}^2 \to \mathbb{R}$ in $L(\mathbb{R}^2)$, while color images are a function $f : \mathbb{R}^2 \to \mathbb{R}^3$ in $L_{\mathbb{R}^3}(\mathbb{R}^2)$.

2. Discretized images can be thought as a function $f : \{1, \dots, n\} \times \{1, \dots, n\} \to \mathbb{R}$, and hence an element of $L(\{1, \dots, n\}^2)$;

3. Let (X, E) be a graph: if the input is a scalar value associated to each vertex or each edge, then it can be written as an element of $L(X)$ or $L(E)$ respectively.

Theorem 1. Let G be a countable group and let $X = G$ with the natural action given group composition. Given two functions $f, g \in L(X)$, define their G-convolution $f \star_G g$ as

$$(f \star_G g)(x) := \sum_{v \in G} f(xv^{-1})g(v).$$

Then, a linear representation $\phi : L(X) \to L(X)$ is G-equivariant if and only if $\phi(f) = f \star_G g$ for some $g \in L(X)$.

[1] $L_V(X)$ is a vector space with addition and multiplication by scalar defined by $(\alpha f + \beta g)(x) := \alpha f(x) + \beta g(x)$.

Proof. ($\Leftarrow$) We need to prove that $\phi(u \cdot f) = u \cdot \phi(f)$ for any $u \in G$. Expanding the last term we obtain:

$$\begin{aligned}
(u \cdot \phi(f))(x) &= (f \star_G g)(u^{-1}x) \\
&= \sum_{v \in G} f((u^{-1}x)v^{-1})g(v) \\
&= \sum_{v \in G} f(u^{-1}(xv^{-1}))g(v) \\
&= \sum_{v \in G} (u \cdot f)(xv^{-1})g(v) \\
&= ((u \cdot f) \star_G g)(x) \\
&= \phi(u \cdot f)(x).
\end{aligned}$$

($\Rightarrow$) Let $\delta_g \in L(X)$ be the function such that $\delta_v(x) = 1$ if $x = v$ and 0 otherwise. Notice that $v \cdot \delta_u = \delta_{vu}$. Notice that any $f \in L(x)$ can be written as $f(x) = \sum_{v \in G} f(v)\delta_v(x)$ (recall that we are assuming $X = G$). Then,

$$\begin{aligned}
\phi(f)(x) &= \phi(\sum_{v \in G} f(v)\delta_v)(x) \\
&= \sum_{v \in G} f(v)\phi(\delta_v)(x) \\
&= \sum_{v \in G} f(v)\phi(v \cdot \delta_e)(x) \\
&= \sum_{v \in G} f(v)(v \cdot \phi(\delta_e))(x) \\
&= \sum_{v \in G} f(v)g(v^{-1}x) \\
&= \sum_{w \in G} f(xw^{-1})g(w) \\
&= (f \star_G g)(x),
\end{aligned}$$

where we have defined $g = \phi(\delta_e)$, with $e \in G$ is the group identity, and we used the change of variables $w = v^{-1}x$. □

References

[1] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. *arXiv preprint arXiv:1802.03690*, 2018.

[2] Stefano Soatto. Steps towards a theory of visual information: Active perception, signal-to-symbol conversion and the interplay between sensing and control. *arXiv preprint arXiv:1110.2053*, 2011.